frontiers | Research Topics

Robotic telescopes

Edited by
Alberto J. Castro-Tirado, Frederic Victor Hessman, Chenzhou Cui and Bringfried Stecklum

Published in
Frontiers in Astronomy and Space Sciences

ISSN 1664-8714
ISBN 978-2-8325-2235-6
DOI 10.3389/978-2-8325-2235-6

About Frontiers

Frontiers is more than just an open access publisher of scholarly articles: it is a pioneering approach to the world of academia, radically improving the way scholarly research is managed. The grand vision of Frontiers is a world where all people have an equal opportunity to seek, share and generate knowledge. Frontiers provides immediate and permanent online open access to all its publications, but this alone is not enough to realize our grand goals.

Frontiers journal series

The Frontiers journal series is a multi-tier and interdisciplinary set of open-access, online journals, promising a paradigm shift from the current review, selection and dissemination processes in academic publishing. All Frontiers journals are driven by researchers for researchers; therefore, they constitute a service to the scholarly community. At the same time, the *Frontiers journal series* operates on a revolutionary invention, the tiered publishing system, initially addressing specific communities of scholars, and gradually climbing up to broader public understanding, thus serving the interests of the lay society, too.

Dedication to quality

Each Frontiers article is a landmark of the highest quality, thanks to genuinely collaborative interactions between authors and review editors, who include some of the world's best academicians. Research must be certified by peers before entering a stream of knowledge that may eventually reach the public - and shape society; therefore, Frontiers only applies the most rigorous and unbiased reviews. Frontiers revolutionizes research publishing by freely delivering the most outstanding research, evaluated with no bias from both the academic and social point of view. By applying the most advanced information technologies, Frontiers is catapulting scholarly publishing into a new generation.

What are Frontiers Research Topics?

Frontiers Research Topics are very popular trademarks of the *Frontiers journals series*: they are collections of at least ten articles, all centered on a particular subject. With their unique mix of varied contributions from Original Research to Review Articles, Frontiers Research Topics unify the most influential researchers, the latest key findings and historical advances in a hot research area.

Find out more on how to host your own Frontiers Research Topic or contribute to one as an author by contacting the Frontiers editorial office: frontiersin.org/about/contact

Robotic telescopes

Topic editors

Alberto J. Castro-Tirado — Institute of Astrophysics of Andalusia, Spanish National Research Council (CSIC), Spain

Frederic Victor Hessman — Georg-August-Universität Göttingen, Germany

Chenzhou Cui — National Astronomical Observatories, Chinese Academy of Sciences (CAS), China

Bringfried Stecklum — Thüringer Landessternwarte Tautenburg, Germany

Citation

Castro-Tirado, A. J., Hessman, F. V., Cui, C., Stecklum, B., eds. (2023). *Robotic telescopes*. Lausanne: Frontiers Media SA. doi: 10.3389/978-2-8325-2235-6

Table of contents

frontiers | Frontiers in Astronomy and Space Sciences

ORIGINAL RESEARCH
published: 22 June 2022
doi: 10.3389/fspas.2022.903429

Exoplanet Two-Square Degree Survey With SAO RAS Robotic Facilities

Oleg Ya. Yakovlev[1,2]*, *Azamat F. Valeev*[1,3,4], *Gennady G. Valyavin*[1], *Alexander V. Tavrov*[2,5], *Vitaly N. Aitov*[1], *Guram Sh. Mitiani*[1], *Oleg I. Korablev*[2], *Gazinur A. Galazutdinov*[1,4], *Grigory M. Beskin*[1], *Eduard V. Emelianov*[1], *Timur A. Fatkhullin*[1], *Valery V. Vlasyuk*[1], *Vyacheslav V. Sasyuk*[6], *Alexei V. Perkov*[7], *Sergei Bondar*[7†], *Tatyana E. Burlakova*[1,4], *Sergei N. Fabrika*[1] *and Iosif I. Romanyuk*[1]

[1]Special Astrophysical Observatory, Russian Academy of Sciences, Nizhnij Arkhyz, Russia, [2]Space Research Institute, Russian Academy of Sciences, Moscow, Russia, [3]Mathematics and Mechanics Faculty, Saint Petersburg State University, Saint Petersburg, Russia, [4]Federal State Budget Scientific Institution Crimean Astrophysical Observatory, Russian Academy of Sciences, Bakhchisaray, Russia, [5]Moscow Institute of Physics and Technology, Dolgoprudnii, Russia, [6]Institute of Physics, Kazan Federal University, Kazan, Russia, [7]Research Corporation "Precision Systems and Instruments", Moscow, Russia

OPEN ACCESS

Edited by:
Alberto J. Castro-Tirado, Institute of Astrophysics of Andalusia (CSIC), Spain

Reviewed by:
Javier Pascual Granado, Institute of Astrophysics of Andalusia (CSIC), Spain
Bringfried Stecklum, Thüringer Landessternwarte Tautenburg, Germany

***Correspondence:**
Oleg Ya. Yakovlev
yko-v@ya.ru

†*Deceased*

Specialty section:
This article was submitted to Astronomical Instrumentation, a section of the journal Frontiers in Astronomy and Space Sciences

Received: *24 March 2022*
Accepted: *13 May 2022*
Published: *22 June 2022*

Citation:
Yakovlev OY, Valeev AF, Valyavin GG, Tavrov AV, Aitov VN, Mitiani GS, Korablev OI, Galazutdinov GA, Beskin GM, Emelianov EV, Fatkhullin TA, Vlasyuk VV, Sasyuk VV, Perkov AV, Bondar S, Burlakova TE, Fabrika SN and Romanyuk II (2022) Exoplanet Two-Square Degree Survey With SAO RAS Robotic Facilities. Front. Astron. Space Sci. 9:903429. doi: 10.3389/fspas.2022.903429

We used the 0.5-m robotic telescope located at the Special Astrophysical Observatory of the Russian Academy of Sciences for monitoring two square degrees of the sky with the aim of detecting new exoplanets. A dimming of the visible brightness is expected due to the exoplanets transiting their host stars. We analyzed about 25,000 raw images of stars taken in the period between August 2020 and January 2021 and plotted the light curves for about 30,000 stars on a half-year timescale. Five newly discovered exoplanet candidates are being investigated to determine their transit event parameters. We also present the light curves for dozens of binary stars.

Keywords: exoplanets, photometry, transit method, robotic telescope, variable stars

1 INTRODUCTION

At present, exoplanets are detected, and their parameters are determined mainly by two indirect observation methods: photometric, where the light curve is analyzed when an exoplanet passes across the disk of the host star (transit method) (Deeg and Alonso, 2018), and spectral measurements, used for analyzing the radial velocity curve of the host star influenced by the gravitational pull of the exoplanet (RV method) (Wright, 2017). In ground-based observations, most exoplanets (1,296 out of 1,494) (NASA, 2022) were detected using the RV method (910) or the transit method (386). In space observations, the most used technique is the transit method (3,387 out of 3,426), while the RV method is not used [hereinafter, exoplanets and their host-star parameters and counts are taken from (NASA, 2022) relevant for 2022.02.18].

The region in the exoplanet and host star parameter space in which exoplanets are detected is determined by the technical capabilities of observational instruments, the features of exoplanet detection methods, and the presence of an atmosphere in ground-based observations. Space telescopes [CoRoT (Auvergne et al., 2009), Kepler (Borucki, 2016), and TESS (Ricker et al., 2015)] are more focused (**Figure 1A**) on detecting small and light exoplanets (Earth-like planets and mini-Neptune), while ground-based telescopes [Super-WASP (Pollacco et al., 2006) and HATNet (Bakos et al., 2004)] detect mostly large and heavy ones (mainly gas giants). Most exoplanets with short orbital periods were detected by the transit method: those with periods of less than 50 days were observed by using space telescopes, and exoplanets whose periods are less than 10 days were observed by using ground-based telescopes. Exoplanets with long periods up to 40,000 days (106 years) (Rosenthal et al., 2021) were detected by using ground-based telescopes by the RV method.

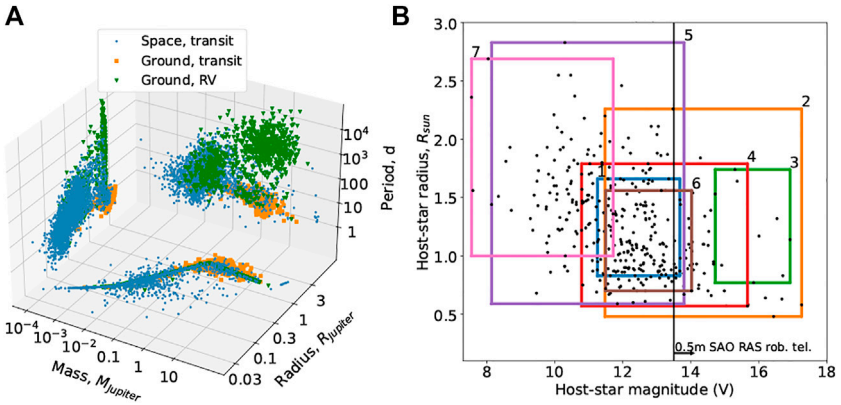


FIGURE 1 | (A) Confirmed exoplanets (NASA, 2022) in the mass–radius–period planes depending on the discovery locale (space and ground-based) and on the detection method (transit, RV). For exoplanets with unknown mass or radius, they were calculated using the mass-radius model dependence. **(B)** Working areas in the host stars' magnitude–radius plane for seven ground-based telescopes that detect exoplanets using the transit method (shown by dots, 347 out of 386): 1.0.1 m Schmidt telescope, 2.0.18 m Takahashi Epsilon Astrograph, 3.1.3 m Warsaw University telescope, 4.0.2 m telescope, 5.0.2 m Canon, 6.0.4 m Canon, and 7.0.08 m Mamiya 645. Also shown is the minimum limiting magnitude of the considered stars (13.5^m) for data that were obtained with the 0.5-m SAO RAS robotic telescope and are discussed in this article; the maximum limit (19.5^m) is not shown.

Ground-based observations are weighed down by the Earth's atmosphere which produces star scintillation and variation of seeing and the duty cycle of observations. The wavefront perturbations of stellar light passing through the atmosphere reduce the signal-to-noise ratio, which limits photometric precision and the minimum brightness dimming that can be registered. The brighter the star, the greater is the signal-to-noise ratio, but the decrease in brightness during a transit will be less significant. Therefore, exoplanets near the stars with magnitudes $m = 10 - -14^m$ are mainly detected (300 out of 386) by the transit method in ground-based surveys (**Figure 1B**). The daily rotation and annual motion of the Earth determine the possibility of observing a given section of the sky from a given observation point. Therefore, the transit of an exoplanet may occur at those moments of time when it cannot be observed, several times in a row, although the technical capabilities make it possible to register the observed decrease in brightness. For reliable detection of exoplanets, one must observe at least several transit events that occur with a period equal to the exoplanet orbital period (for example, it takes several years or more to detect planets at a distance of more than 1 AU from a Sun-like star).

Thus, for the detection of exoplanets by the transit method in ground-based surveys, it is specifically important to increase the number of observed stars and the observation time. For these purposes, it is relevant to use robotic telescopes that regularly perform long-term routine observations of a certain part of the sky (to increase the observation time) or different parts of the sky (to increase the number of stars). The use of a group of such telescopes makes it possible to achieve both the goals.

A group of robotic telescopes is currently being developed at SAO RAS with the aim of detecting exoplanets by the transit method and conducting additional observations of the already known exoplanets. Currently, one such telescope is under operation, automatically observing two regions of the sky since the summer of 2020, each for 6 months. We have developed a piece of software to perform photometric analysis and to search for transit events in the light curves of the stars. In this article, we present the analysis and primary results of processing the first data set obtained in the last third of 2020.

The aim of this work was to examine the obtained data for suitability for the search of exoplanets. To that end, it was

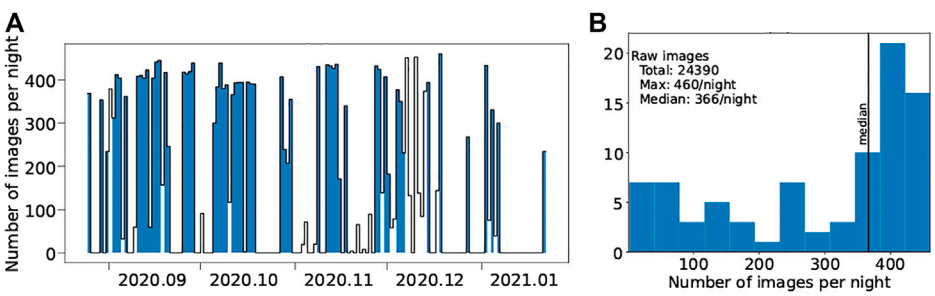


FIGURE 2 | Statistics on the number of images per night: **(A)** distribution by time with the colored columns marking the nights with successfully performed photometry and **(B)** distribution by number.

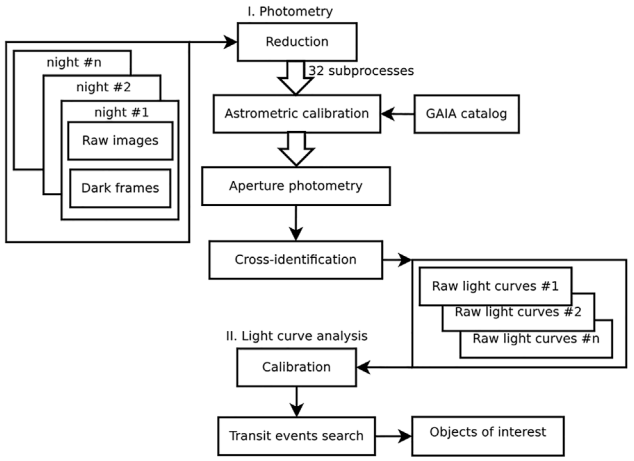


FIGURE 3 | Flowchart of the data processing algorithm for detecting exoplanets using the transit method.

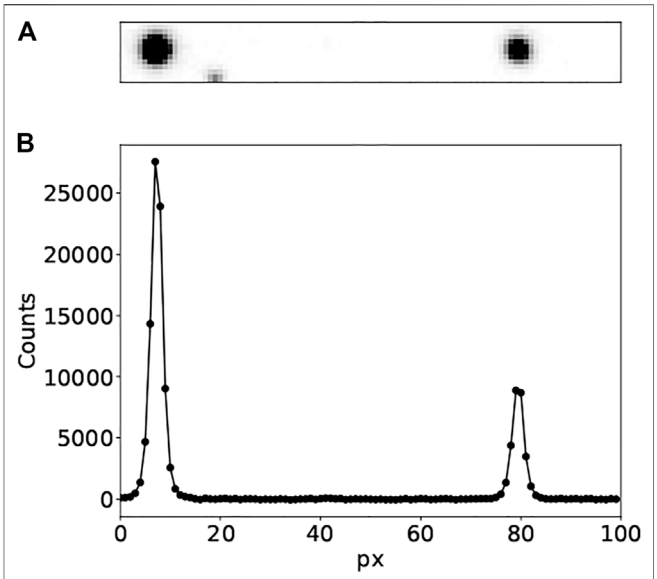


FIGURE 4 | Example with two stars in the image after reduction **(A)** and their projection onto one axis, where each point corresponds to the counts in a pixel **(B)**.

necessary to go through all the stages of image processing, determine the problems that arise in the process of data processing, and evaluate the possibility of using algorithms to search for transit events in the light curves. **Section 2.2** is devoted to the description of the observations and the quality of the obtained data. **Section 2** briefly describes the stages of the developed pipeline and also discusses the problems that arise when working with the data. The primary results of the search for exoplanet candidates are presented in **Section 3**, which also presents the light curves for some of the observed variable stars. A discussion of the five detected candidates and the prospects of our exoplanet survey are presented in **Section 4**.

2 DATA AND METHOD

2.1 Robotic Telescope Array

The "Astrosib" (Novosibirsk, Russia) RC-500 telescopes with a 0.5-m hyperbolic main mirror were installed on the fast tracking "10 Micron GM 4000" high precision equatorial mounts. In the primary focus, each unit was equipped with an FLI Proline PL16801 front illuminated CCD camera with a 9 μm pixel size. The Baader Planetarium AllSky 4.5-m dome with a weather forecast meteo station and a 4.5-m Astrosib AllSky dome were used as shelters. The FLI CCD camera, FLI Atlas focuser, and FLI five-position motorized 50-mm filter wheel were operated by an industrial PC which collects the raw data and provides remote access.

The cloud and humidity sensors automatically send the telescope into its parking position and signal for the dome to close. The twilight sky flat-field correction frames have been obtained since February 2021. Bias and dark calibration frames are collected daily.

2.2 Observations

From 25 August 2020 to 21 January 2021, surveys were carried out for 84 nights (**Figure 2A**). Of these, 56 turned out to be suitable for light curve analysis; they are colored in **Figure 2A**. The white dwarf WD0009 + 501 with a well-known 8-h period and 5 mmag amplitude variations was chosen as the central object in the frame (Valeev et al., 2015). The 2.45 × 1.56° (RA x DEC) field of view around this object has good visibility conditions in the autumn.

Observations were carried out with an exposure of 60 s and with a period of 80–100 s in the V filter of the Johnson system. The image scale was $1.34''/pixel$. Each night, 2 to 460 images were taken; the median value was 366. The total number of images is 24,390 (**Figure 2B**). In addition, 10 dark and 10 bias frames were made for each night. No flat-field frames were taken.

2.3 Data Processing

The image processing is implemented in Python scripts combined into one bash-script. It can be divided into two independent parts (**Figure 3**): photometry–automatically obtaining light curves from images and light curve analysis–searching for dimming in the light curves.

A catalog of stars in the considered field was prepared in advance, according to the data on the GAIA space telescope [Prusti et al. (2016); Brown et al. (2021)]: ID, coordinates (ra, dec), magnitude. This catalog contains 39,978 stars with magnitudes $m \in [13.5,19.5]^m$.

2.3.1 Photometry

The entire field was divided into nine overlapping parts: central (2500 × 2500 pix), two sides (820 × 2500 pix), top and bottom (2500 × 820 pix), and four corners (1000 × 1000 pix). For each of the nine parts of the frame, the processing is carried out sequentially for each night, which takes up to 5 min of machine time (CPU frequency 2.2 GHz). Obtaining light

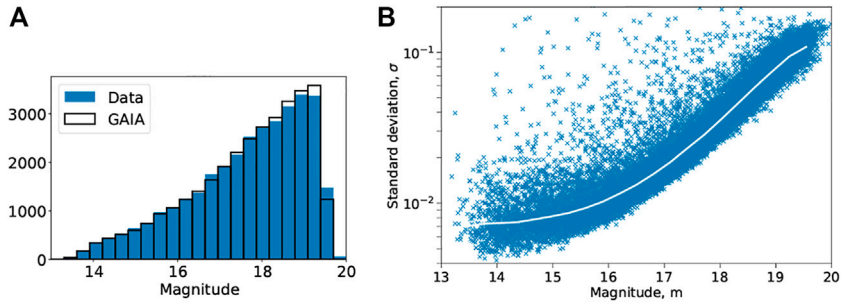


FIGURE 5 | Photometry statistics for one night: **(A)** magnitude distribution compared to the GAIA catalog (Prusti et al. (2016); Brown et al. (2021)) and **(B)** standard deviation versus magnitude, with the median dependence shown by the white curve.

curves for one part of the frame for all nights takes up to 2 to 4 h. Reduction, astrometry calibration, and aperture photometry are carried out in a multiprocessing mode, which makes it possible to process 32 images simultaneously at these stages.

The first stage of the pipeline is to create a configuration file based on the parameters entered by the user. After that, a calibration file is prepared. A total of 10 dark frames are combined, and the resulting dark frame is subtracted from the images taken at night. Further, using the astrometry.net package (Lang et al., 2009), the images are calibrated to the celestial coordinate system, and the coefficients of the transition matrix from the rectangular coordinates of the image x, y to astronomical equatorial coordinates ra, dec are determined. Reduction and astrometry calibration are carried out using the CCDpack package (Warren-Smith et al., 2014).

The next step is aperture photometry using the SExtractor package (Bertin and Arnouts, 1999) in eight apertures with a diameter of 4–12 pixels. The PSF of stars falls within this range under various weather conditions (an example is shown in **Figure 4**). For the identification of stars, the previously prepared GAIA catalog with an association radius of 5 pixels is used. In the resulting table, each star from the catalog identified by the source in the image is assigned an instrumental magnitude for each aperture.

Furthermore, the obtained tables for each image taken at night are combined into a single catalog, in which instrumental magnitudes in eight apertures for the entire night are assigned to each identified star. Thus, as a result of performing the described steps, the light curves of the stars from the input catalog are obtained from raw images for each night. **Figure 5A** shows a magnitude distribution example for the stars, according to the catalog derived from photometry. For some stars with magnitudes of $m = 14^m$, the standard deviation reaches minimum values up to 0.005^m (**Figure 5B**).

To select the operating range of magnitudes, we first analyzed the dependence of magnitude, according to the GAIA catalog, on the number of counts on the CCD array. Then, 13.5^m was taken as the lower limit, and 19.5^m was taken as the upper limit. In this range, there are no stars with saturated pixels or those whose signal levels are comparable with the noise levels.

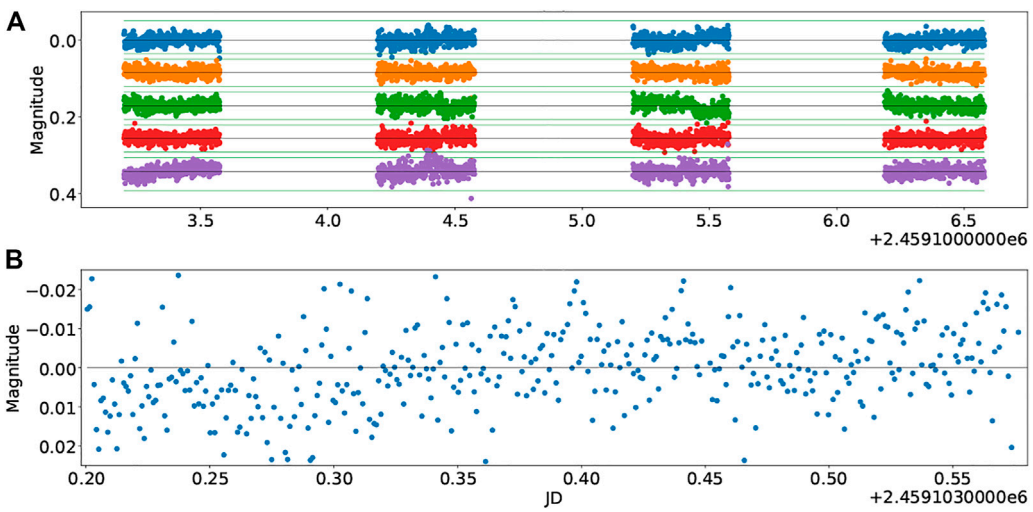


FIGURE 6 | Light curve examples for four stars with $m \approx 15.5^m$: **(A)** for 4 nights, with a $\pm 0.05^m$ zone shown for each star, and **(B)** for the first star on the first night.

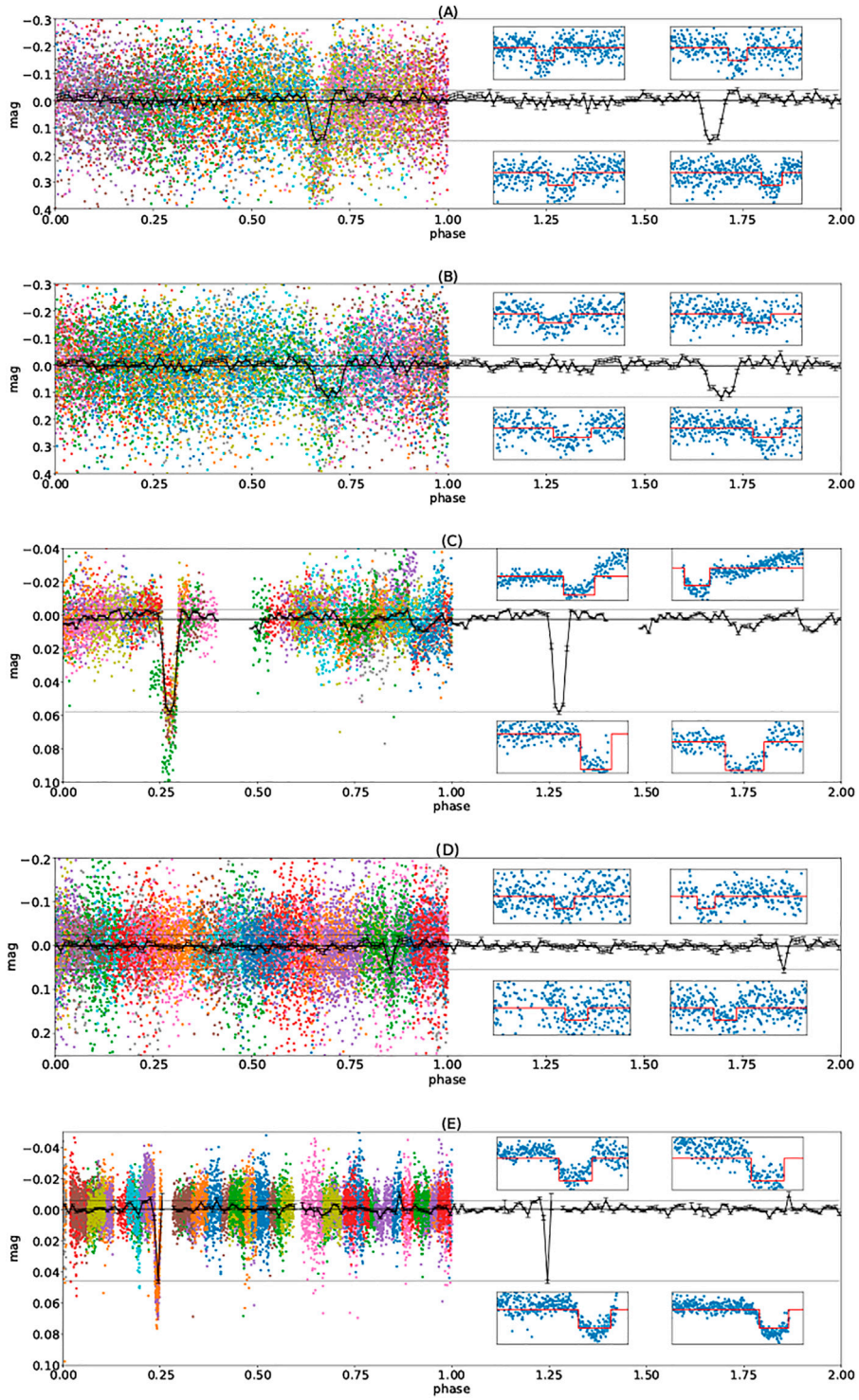


FIGURE 7 | Folded light curves of five objects of interest and binned averaged curves (main panels). Points from different periods are shown in different colors. The horizontal lines show the minimum, maximum, and median values. The four small panels show examples of individual transit events with the model found by the BLS method (see their parameters in **Table 1**).

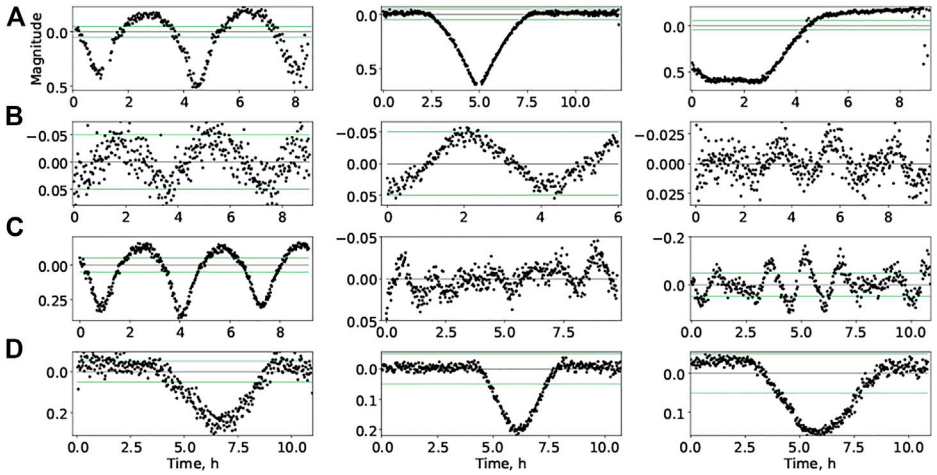


FIGURE 8 | Light curves of variable stars: **(A)** with large amplitudes ($\Delta m > 0.5^m$), **(B)** with small amplitudes ($\Delta m <= 0.1^m$), **(C)** with short periods ($P < 4^h$), and **(D)** with a shape similar to a transit event. The horizontal lines show the $\pm 0.05^m$ zone.

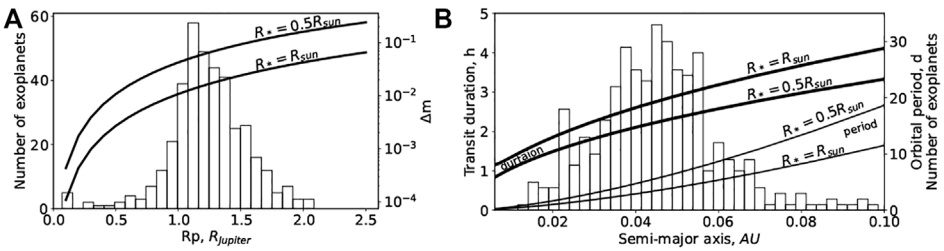


FIGURE 9 | Maximum brightness dimming of the host star during a transit event Δm versus the exoplanet radius (**(A)**, right axis) and **(B)** transit duration (left axis and bold lines) and orbital period (right axis and thin lines) versus the semi-major axis of the exoplanet orbit for the stars of solar radius and those two times smaller. Also shown are the histograms of the distribution of confirmed transiting exoplanets (NASA, 2022) detected by ground-based facilities by the exoplanet radius (**(A)**, left axis) and by the semi-major axis of their orbits (**(B)**, right axis).

TABLE 1 | Objects of interest and transit event parameters.

Object	SOI-1	SOI-2	SOI-3	SOI-4	SOI-5
Magnitude, m	18.8	18.9	14.3	17.8	14.9
Period, h	26.1	25.2	46.0	63.4	198.3
Depth, m	0.1	0.07	0.05	0.04	0.04
Duration, h	1.4	2.0	1.7	1.6	2.4

2.3.2 Light Curve Analysis

The light curves obtained for each night are combined into one for all nights. After that, they are calibrated. Stars are selected as standard stars if they satisfy the following conditions: 1) they are identified on all nights, 2) their magnitudes are less than 15.5, and 3) they are in the same part of the frame as the target star. The light curves for several standard stars are shown in **Figure 6**.

The last step is to search for transit events in the light curves using the BLS (box least squares) method (Kovacs et al., 2002). In this method, the phase folded light curve is approximated by a two-level model: one level (high) before and after the transit and another level (low) during the transit. The function changes instantly at the beginning and at the end of the transit and is similar in shape to a box. The idea of the method is to calculate statistics depending on the differences between the model and the available data. The larger the statistic, the smaller is the residual. For each trial period, the best combination of transit duration and transit start time is determined, which corresponds to the maximum value of the statistics. Thus, a periodogram is constructed, and the dependence of this statistic is on the trial period. On the periodogram, a star with a light curve that exhibits a transit event has a global maximum, which differs significantly from the rest of the local maxima. For each star, a periodogram is plotted for the light curves constructed using five standard stars. The threshold value at which the flag is triggered is set. For most interesting objects, the flag is triggered twice or more. After that, the light curves of the selected objects are viewed individually to decide whether to consider the star as an object of interest.

3 RESULTS

As a result of applying the algorithm to the obtained data (**Section 2**), five interesting objects were found (named SOI–SAO RAS objects of interest). Their light curves and parameters are shown in **Figure 7** and **Table 1**, respectively.

As a result of the deviation-magnitude relation analysis (**Figure 5B**), as well as in the process of searching for exoplanets, more than 90 variable stars were found; the light curves for 12 of them are shown in **Figure 8**. A detailed description of the selection criteria for stars in multiple systems or stars with exoplanet candidates will be updated in the next study. The validation of their binary nature (multiplicity) using additional spectral data from the 6-m BTA telescope has just started.

4 DISCUSSION

We have shown in the practice that exoplanet candidates can be found from the obtained data. The dimming in the objects of interest is in the range of $\delta m \approx 0.04 \ldots 0.1^m$, which corresponds to a transit of a 1–2 Jupiter radius exoplanet across the disk of a star of solar radius or less (**Figure 9A**). They have an orbital period of 1–8 days, corresponding to a planet orbiting close to the host-star, with the semi-major axis less than 0.08 AU (**Figure 9B**). These estimates indicate that hot Jupiter or, possibly, hot Neptune are the targets of our search for exoplanet candidates. After processing the data in this sky area for the second half of 2021, we have planned to search for exoplanets in wider orbits with periods of up to several months. The transit durations of all the SOI with determined periods are less than the corresponding maximum value; they passed this initial test successfully. SOI-2 remains mostly unclear. The odd and even minima in the light curve of SOI-2 differ. Furthermore, based on the detected objects, we can improve the algorithm by refining the criterion for selecting transit events.

The detected objects must undergo further validation. To that end, we have planned to carry out additional photometric and spectral measurements. The light curves in different filters and the radial velocity (RV) amplitudes of the stars will allow us to reject the double star hypothesis. The shape of the residual between the light curves in different filters during the transit of an exoplanet differs from the shape for an eclipsing binary star with different spectral type components. The RV amplitude determines the minimum mass of the component that produces dimming. On one hand, most ground-based surveys observe brighter stars, so our observations in the range of $m \in [13.5,19.5]^m$ can be productive. However, the necessary validation of faint objects by the RV method is more difficult. There is also less information about faint stars, so modeling transit events for validation purposes is more uncertain. Therefore, it is possible that in the future, this range will be shifted toward brighter stars.

The disadvantage of the existing pipeline is the different orientation of raw images, which complicates the last step of creating the light curve, when the star's light curves converge into one for all nights. Therefore, in the future, we have planned to transform each image at the first stage, bringing all the images to the same orientation. In addition, it is possible to speed up the process by forming packages of consecutive images to reduce the time of astrometric calibration. At the light curve analysis stage, detrending needs to be added.

Currently, the data have been processed only for the last third of 2020, but the same sky field was observed in 2021; 2021 data are also available for another field. After processing these data and improving the algorithm, the pipeline will be introduced into the observing process. Processing will be carried out immediately after the observation.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusion of this article will be made available by the authors, without undue reservation.

AUTHOR CONTRIBUTIONS

OY and AV wrote the manuscript and performed data processing. GV, VS, and VV managed the process of facility design and construction. AT and OK stated the problem of exoplanetary research. VA, GM, EE, and TF performed the observation. EE and TF domed and mounted the software development. AP and SB maintained instrumental and technical solutions. TB performed software testing and database engineering. GG, GB, SF, and IR proposed the grant for AstroM funding. SB designed the solution at the first stage; SB died in 2019.

FUNDING

The study was supported by the Government and the Ministry of Education and Science of Russia (Grant No.075-15-2020-780 (N13.1902.21.0039)). AV acknowledges the star multiplicity advocation algorithm development, which was supported by the Russian Science Foundation G rant No 19-72-10023.

ACKNOWLEDGMENTS

In this study, we used the NASA Exoplanet Archive, which is operated by the California Institute of Technology, under contract with the National Aeronautics and Space Administration under the Exoplanet Exploration Program. This work has made use of data from the European Space Agency (ESA), mission *Gaia* (https://www.cosmos.esa.int/gaia), processed by the *Gaia* Data Processing and Analysis Consortium (DPAC, https://www.cosmos.esa.int/web/gaia/dpac/consortium). Funding for the DPAC has been provided by national institutions, in particular the institutions participating in the *Gaia* Multilateral Agreement.

REFERENCES

Auvergne, M., Bodin, P., Boisnard, L., Buey, J.-T., Chaintreuil, S., and Team, C. (2009). The CoRoT Satellite in Flight : Description and Performance. *Astronomy Astrophysics* 506, 411. doi:10.1051/0004-6361/200810860

Bakos, G., Noyes, R. W., Kovács, G., Stanek, K. Z., Sasselov, D. D., and Domsa, I. (2004). Wide-Field Millimagnitude Photometry with the HAT: A Tool for Extrasolar Planet Detection. *Publ. Astron Soc. Pac* 116, 266–000277. doi:10.1086/382735

Bertin, E., and Arnouts, S. (1996). SExtractor: Software for Source Extraction. *Astron. Astrophys. Suppl. Ser.* 117, 393–404. doi:10.1051/aas:1996164

Borucki, W. J. (2016). KEPLERMission: Development and Overview. *Rep. Prog. Phys.* 79, 036901. doi:10.1088/0034-4885/79/3/036901

Brown, A., Vallenari, A., Prusti, T., and de Bruijne, J. (2021). Gaia Early Data Release 3. Summary of the Contents and Survey Properties. *Astronomy Astrophysics* 650, 7. doi:10.1051/0004-6361/202039657e

[Dataset] NASA (2022). *NASA Exoplanet Archive*. Washington, D.C., USA: NASA. doi:10.26133/NEA12

Deeg, H. J., and Alonso, R. (2018). Transit Photometry as an Exoplanet Discovery Method. *Handb. Exopl.* 2018, 633–657. doi:10.1007/978-3-319-55333-7_117

Kovács, G., Zucker, S., and Mazeh, T. (2002). A Box-Fitting Algorithm in the Search for Periodic Transits. *Astronomy Astrophysics* 391, 369–377. doi:10.1051/0004-6361:20020802

Lang, D., Hogg, D. W., Mierle, K., Blanton, M., and Roweis, S. (2010). Astrometry.net: Blind Astrometric Calibration of Arbitrary Astronomical Images. *Astronomical J.* 139, 1782–1800. doi:10.1088/0004-6256/139/5/1782

Pollacco, D., Skillen, I., Cameron, A., Christian, D., Irwin, J., Lister, T., et al. (2006). The WASP Project and SuperWASP Camera. *Astrophys. Space Sci.* 304, 253–255. doi:10.1007/s10509-006-9124-x

Prusti, T., de Bruijne, J., Brown, A., Vallenari, A., Babusiaux, C., Bailer-Jones, C., et al. (2016). The Gaia Mission. *Astronomy Astrophysics* 595, 36. doi:10.1051/0004-6361/201629272

Ricker, G., Winn, J., Vanderspek, R., Berta-Thompson, Z., Levine, A., Seager, S., et al. (2015). Transiting Exoplanet Survey Satellite (TESS). *J. Astronomical Telesc. Instrum. Syst.* 1 (1), 014003. doi:10.1117/1.JATIS.1.1.014003

Rosenthal, L. J., Fulton, B. J., Hirsch, L. A., Isaacson, H. T., Howard, A. W., Dedrick, C. M., et al. (2021). The California Legacy Survey. I. A Catalog of 178 Planets from Precision Radial Velocity Monitoring of 719 Nearby Stars over Three Decades. *ApJS* 255, 8. doi:10.3847/1538-4365/abe23c

Valeev, A. F., Antonyuk, K. A., Pit, N. V., Solovyev, V. Y., Burlakova, T. E., Moskvitin, A. S., et al. (2015). Detection of Regular Low-Amplitude Photometric Variability of the Magnetic Dwarf WD0009+501. On the Possibility of Photometric Investigation of Exoplanets on the Basis of 1-meter Class Telescopes of the Special and Crimean Astrophysical Observatories. *Astrophys. Bull.* 70, 318–327. doi:10.1134/S1990341315030104

Warren-Smith, R., Draper, P., Taylor, M., and Allan, A. (2014). CCDPACK: CCD Data Reduction Package. *Astrophys. Source Code Libr.* 2018.

Wright, J. T. (2018). Radial Velocities as an Exoplanet Discovery Method. *Handb. Exopl.* 2017, 619–631. doi:10.1007/978-3-319-55333-7_4

Conflict of Interest: AP and SB were employed by the Research Corporation "Precision Systems and Instruments".

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

frontiers | Frontiers in Astronomy and Space Sciences

ORIGINAL RESEARCH
published: 24 June 2022
doi: 10.3389/fspas.2022.912546

Eight Years of TIGRE Robotic Spectroscopy: Operational Experience and Selected Scientific Results

José Nicolás González-Pérez[1], Marco Mittag[1], Jürgen H. M. M. Schmitt[1], Klaus-Peter Schröder[2], Dennis Jack[2], Gregor Rauw[3] and Yaël Nazé[3]*

[1]Hamburger Sternwarte, Universität Hamburg, Hamburg, Germany, [2]Departamento de Astronomía, Universidad de Guanajuato, Guanajuato, Mexico, [3]Groupe d'Astrophysique des Hautes Energies, STAR, Université de Liège, Liège, Belgium

TIGRE (Telescopio Internacional de Guanajuato Robótico Espectroscópico) has been operating in fully robotic mode in the La Luz Observatory (Guanajuato, Mexico) since the end of 2013. With its sole instrument, HEROS, an échelle spectrograph with a spectral resolution R ~20,000, TIGRE has collected more than 48,000 spectra of 1,151 different sources with a total exposure time of more than 11,000 h in these 8 years. Here we briefly describe the system and the upgrades performed during the last years. We present the statistics of the weather conditions at the La Luz Observatory, emphasizing the characteristics that affect the astronomical observations. We evaluate the performance and efficiency of TIGRE, both optical and operational, and describe the improvements of the system implemented to optimize the telescope's performance and meet the requirements of the astronomer in terms of timing constraints for the observations and the quality of the spectra. We describe the actions taken to slow down the optical efficiency loss due to the aging of the optical surfaces as well as the upgrades of the scheduler and the observing procedures to minimize the time lost due to interrupted observations or observations that do not reach the required quality. Finally, we highlight a few of the main scientific results obtained with TIGRE data.

Keywords: automated telescopes, instrumentation: spectrographs, techniques: spectroscopic, atmospheric effects, stars: activity, stars: massive, novae, supernovae

OPEN ACCESS

Edited by:
Chenzhou Cui,
National Astronomical Observatories (CAS), China

Reviewed by:
Yi Hu,
National Astronomical Observatories (CAS), China
Jianyan Wei,
National Astronomical Observatories (CAS), China

****Correspondence:***
José Nicolás González-Pérez
st1h317@hs.uni-hamburg.de

Specialty section:
This article was submitted to Astronomical Instrumentation, a section of the journal Frontiers in Astronomy and Space Sciences

Received: *04 April 2022*
Accepted: *06 May 2022*
Published: *24 June 2022*

Citation:
González-Pérez JN, Mittag M, Schmitt JHMM, Schröder K-P, Jack D, Rauw G and Nazé Y (2022) Eight Years of TIGRE Robotic Spectroscopy: Operational Experience and Selected Scientific Results. Front. Astron. Space Sci. 9:912546. doi: 10.3389/fspas.2022.912546

1 INTRODUCTION

Robotic telescopes have gained considerable importance over the last 2 decades (Castro-Tirado, 2010), and so the advent of robotic astronomy has transformed observational astronomy due to the increased economic and operational efficiency. In addition, robotic telescopes allow different modes of observation that are not possible or are only very difficult to achieve with "normal" telescopes. These include rapid alerts, long-term monitoring, phase-constrained observations, or simultaneous observations with other observatories.

In the list of robotic telescopes maintained by F. V. Hessmann[1], one finds more than eighty robotic telescopes in operation and more than forty in commissioning, under construction, or

[1]http://www.astro.physik.uni-goettingen.de/~hessman/MONET/links.html; last actualization from 2016.

FIGURE 1 | The TIGRE facility at La Luz Observatory.

planning. While most of these telescopes are small, a few tens have a diameter larger than 1 m. Even a 4 m diameter telescope, the NRT, will be starting operating in La Palma in the next few years (Gutiérrez et al., 2019). The scientific goals of the robotic telescopes cover the detection of near-Earth asteroids (such as LINEAR, Stokes et al., 2000), the study of exoplanets (e.g., SuperWASP, Pollacco et al., 2006), to the detection of rapid transients such as γ-ray Bursts (e.g., BOOTES, Castro-Tirado et al., 2012). Only a few of these telescopes are exclusively dedicated to high-resolution spectroscopy, such as STELLA-1 (Strassmeier et al., 2004), SONG (Andersen et al., 2019), or TIGRE (Schmitt et al., 2014), the topic of this paper.

The TIGRE (Telescopio Internacional de Guanajuato Robótico Espectroscópico) project is a collaboration between Hamburg Observatory (Germany), and the universities of Guanajuato (Mexico) and Liège (Belgium). The telescope was installed in 2013 in its final location, the La Luz Observatory in central Mexico (see **Figure 1**, which gives an impression of the telescope building and its surroundings), but it was first delivered to Hamburg, where it was used to test and develop the final system (Mittag, 2010). Initially, TIGRE was designed to study the stellar activity of cool stars using the CaII H&K line cores, but over the years it has widened its scientific goals to other fields of stellar astronomy, like hot stars, novae, or binaries.

In this paper, we describe our experience with the first 8 years of fully-automatic operation of TIGRE. We summarize the characteristics of the telescope, instrument, and software in Sect. 2. We also describe in Sect. 3 the La Luz Observatory and report the weather, seeing, and photometric statistics of the observatory. In Sect. 4, we discuss the different aspects of the efficiency of TIGRE, from the optical to the operational point of view. Finally, in Sect. 5, we outline some of the main scientific results obtained with TIGRE.

2 SYSTEM OVERVIEW

The TIGRE telescope was manufactured by Halfmann Teleskoptechnik GmbH. The Cassegrain-Nasmyth optics of the telescope was built by Carl Zeiss Jena and made of Zerodur. The primary mirror (M1) has a diameter of 1.2 m with a focal ratio of f/3. With the secondary mirror, the focal distance of the system is 9.6 m with a focal ratio of f/8. The telescope has two Cassegrain-Nasmyth foci that can be accessed by changing the position of the flat M3 mirror, although currently only one of the foci is in use. Since M1 is relatively thin, it has actuating cell support with a total of 30 static levers, 18 at the back of the mirror arranged in two rings, and 12 at the edge of the mirror.

The compact Alt/Az mount has an hydraulic bearing in the azimuth axis and a mechanical bearing in the elevation axis. The high precision encoders allow a high slewing velocity of $5^{\circ}s^{-1}$ and high precision pointing and tracking. Using an extended pointing model, with 17 parameters instead of the standard 11 for a Nasmyth telescope, we could improve the pointing accuracy to $< 5''$ (Mittag et al., 2008).

The only focal-plane instrument of TIGRE is the Heidelberg Extended Range Optical Spectrograph (HEROS), which is located in a thermally controlled room. Its spectral resolution is $\lambda/\Delta\lambda \sim$ 20,000, and a beam-splitter divides the spectrum into two channels with wavelength ranges of 3,740–5,740 Å and 5,770–8,830 Å. The CCD cameras of each channel are Andor iKonL CCDs with E2V 2K×2K chips and a pixel size of 13.5 μm. These CCDs have very low dark noise and high quantum efficiency and are optimized for the wavelength range of their respective channels and reach quantum efficiencies of 80% at 3,900 Å, 93% at 6,000 Å, and 50% at 8,500 Å. The spectrograph is fed by a 15 m fused silica multimode fiber with a core diameter of 50 μm. Microlenses are attached to both ends of the fiber to adapt the focal ratios of the spectrograph and telescope.

The adapter, located in one of the Cassegrain-Nasmyth foci, contains the acquisition and guiding unit, the calibration unit, and feeds the light of the stars or the calibration lamps into the optical fiber. The diameter of the microlens at the fiber entrance is 150 μm which translates to $\sim 3''$ on the sky. A pellicle beam-splitter in the adapter deviates 8% of the starlight to the guiding camera, and the remaining 92% enter the fiber. Although losing a small fraction of the light, this procedure allows us to accurately monitor the seeing and the transparency changes of the atmosphere in real-time, which is very useful to estimate the necessary exposure times (see below).

The guiding camera is an Atik 420 m CCD with a field of view of $2.6' \times 1.9'$. A set of neutral density filters helps to fix the exposure time of the guiding to 10 s regardless of the star's brightness. Also, the adapter incorporates a tungsten lamp and a hollow cathode ThAr lamp for flat-field and wavelength calibrations, respectively.

The design of the building is very compact (cf. **Figure 1**); the clamshell dome, with a diameter of only 6 m, thanks to the small size of the mount, was manufactured by Astrohaven Enterprises. The telescope is mounted over a pier 3.5 m above ground to avoid most of the dust raised by wind during the dry season. The building has two extensions, one of which is the air-conditioned spectrograph room, and the second contains the electronic cabinet and the hydraulic unit of the telescope.

The monitoring of the weather parameters is absolutely essential for the system's health. We monitor the weather conditions at La Luz Observatory using three different weather stations and an independent rain sensor. Two of the weather stations are placed on a ~6 m high pole, together with a GPS antenna that provides the accurate time necessary for the telescope pointing and tracking. Additionally, we use an all-sky camera to assess the cloudiness above the observatory independently. The most critical parameters are humidity, rain, wind, and sky temperature. Each of these four parameters are provided by more than one weather station, so the system can continue monitoring these parameter in case of failure of one of the devices.

The sky temperature device contains an IR photometer pointed to the sky and an air thermometer. It measures the difference between the ambient temperature and the sky temperature. In a cloudy sky, the sky temperature is the temperature at the bottom of the clouds. This temperature difference gives us a rough estimate of the level of cloudiness, so a threshold in this temperature difference can be used to decide when to close due to a cloudy sky. Although this relationship is far from being perfect–it depends on other factors, e.g., humidity–it is very effective to evaluate the rain risk, thus protecting the telescope's health.

The software of TIGRE, written in Java under Linux, was designed simple and modular, to make it very easy to substitute or add a device, such as a weather station. Each subsystem of TIGRE is operated by its program that communicates with the main program (CCS) through TCP/IP ASCII messages. Two essential parts of the CCS are the robotic operator and the error handler. The robotic operation of the telescope is fully automatic and is divided into procedures: initialization, calibrations, observation, open, close, and shutdown. Each procedure starts when several conditions are met. These conditions depend on the weather, time of the day, Sun elevation, and whether other procedures have already been performed. Also, we implemented a comprehensive error detection and handling. We are continuously improving the error handler once we gain experience with any upcoming issues. In the event of severe problems, where the system is at risk, the error handler closes and shuts the telescope down and sends an e-mail to the technical staff reporting the issue.

The scheduler, one of the subsystems, selects the star to be observed from a pool of stars considering a series of multiplicative weights. These weights account for the targets' scientific priority, the star's position in the sky (the scheduler favors stars close to the meridian), the timing requirements of the astronomer, and whether the available time in the night is enough to finish the observation. Also, the scheduler considers Targets of Opportunities (ToOs) that have the highest priority. TIGRE will observe the active ToOs when the weather conditions are favorable and all other requirements are fulfilled.

At the end of the night, once all calibrations are finished, the data are copied to Hamburg, where the automatic reduction pipeline immediately starts. The pipeline is written in IDL and adapted from the REDUCE package (Piskunov and Valenti, 2002) to the TIGRE-HEROS data. It corrects the raw images from bias, automatically calibrates the wavelength, calculates the order positions, and makes an optimal extraction of the spectra. With the flat-field frames taken each night, the pipeline performs the flat field and the blaze function corrections (Mittag et al., 2010). Also, it automatically calculates the radial velocity (RV) of the stars (Mittag et al., 2018) and several stellar activity indices from the Ca II H&K and infrared triple lines (Hempelmann et al., 2016; Mittag et al., 2016, 2017). Besides, the pipeline produces a series of plots to monitor the CCDs' performance and the reduction quality. TIGRE observes each night one photometric standard star and one RV standard star. These observations are used to remove the residuals from the blaze correction and monitor the instrument's RV stability, respectively.

TIGar, the TIGRE archive, is the interface between the astronomer and the telescope. It comprises a MySQL database and has access to the FITS files (reduced and raw spectra) obtained by TIGRE. The human interface is a webpage written in PHP. Using this webpage, the astronomer can create or edit a proposal and add the stars to be observed with their requirements. Also, when the observations are finished and reduced, the astronomer can retrieve the data through the same webpage.

3 LA LUZ OBSERVATORY

La Luz Observatory is located about 20 km from Guanajuato, about 300 km NW of Mexico City in the high plateau of central Mexico at coordinates 101.32478W and 21.053139N and at an elevation of 2,435 m above sea level.

Since the observatory is relatively isolated, an internet microwave beam antenna connects to a server on the Cerro de Cubilete around 10 km away. The electricity supply, unfortunately, shows frequent power failures, overvoltages, and surges, often related to thunderstorms and strong winds. A UPS and a power plant operating in the observatory guarantee a regulated power supply.

The climate in the observatory has two very different seasons: a very dry winter and a humid-rainy summer. The conditions in winter are excellent for the observations (see below), but the dust production is relatively high due to the dryness and the type of soil. After the road to the observatory was paved in 2014, the dust contamination has decreased significantly. In the summer, on the contrary, thunderstorms cause many power failures, and

lightning strikes can cause severe damage to the system. Although we have a robust lightning protection, with surge protection in the power line and lightning rods, we already had to replace the internet and GPS antennas twice.

In the following subsections, we provide a detailed analysis of the weather conditions and the seeing in the observatory.

3.1 Weather Conditions

We have collected the weather data of 2015–2021 from our weather stations and present their statistics in this section. January is the coldest month with an average temperature of 12.5°C (the average nightly temperature is 11°C). Afterwards, the monthly average temperatures rise continuously until May, when the average temperature reaches 19°C (17 °C at night). Then, the temperatures drop (< 17°C), even in the summer, because the cloudy sky prevents the ground to be excessively heated.

The temperature variations during the night are relatively small, typically below five degrees between June and December and less than seven degrees during the rest of the year. The maximum daily temperature contrast is usually below 10°C, and between July and January below 8.5°C. A small temperature drop in the night helps to match the telescope and air temperatures and thus reduce the influence of the local air layer in the seeing.

The daily-averaged relative humidity shows seasonal variations, with a minimum in April (around 30%) and a maximum in September (70%). From June to October, the humidity is generally high (> 60%). The rainiest month is June, followed by July, August, and September in this order.

There are no substantial variations of the monthly-average wind velocities, with values between 2 ms^{-1} and 4 ms^{-1}. However, the averages of the maximal-minimal wind daily differences have a maximum of 9.5 ms^{-1} in June. It is higher than 8 ms^{-1} between March and October, indicating that the probability of having adverse winds preventing any observations is much higher during these months. Furthermore, the windy nights are also the nights with larger seeing (see below), which has led us to analyze the frequency of the strong winds. The fraction of the time with wind velocities > 7.5 ms^{-1} grows from 0.02 in December and January to 0.11 in June. After June, this fraction remains higher than 5% until October. Furthermore, windy conditions in summer are ten times more probable when the sky is clear than cloudy.

To finalize this section, we study the periods of time with good weather conditions and how this changes through the year. Good weather conditions are defined by thresholds in some weather parameters: wind, humidity, sky temperature, and rain. We principally set these limits to avoid damages in the system and evaluate when the observations are possible with a high probability. Observations are allowed only for wind velocities lower than 10 ms^{-1}. With stronger winds, the telescope starts oscillating, and the dome may be damaged. The maximum permitted humidity is 90% to avoid condensation on the mirror surfaces. Any rain detection also triggers the procedure to abort the observations and close the dome.

The last threshold is for the difference between the sky and ambient temperature: we consider the sky as cloudy if this difference is higher than -28°C. Note that this limit cannot be extrapolated to other observatories because it depends on the location, particularly on the elevation above sea level and the device itself. For the setting of this limit, we have carefully compared the sky temperature values with the all-sky camera images. However, it should be emphasized that the correlation between the sky temperature and the cloudiness is not very tight. The sky temperature is not very sensitive to high clouds; thus, the sky may be covered by high clouds, and the sky temperature is below the threshold by two or three degrees. The robotic operator handles this possibility by estimating the necessary exposure time using the guiding images (see below), or aborting the observation if the star is not detected.

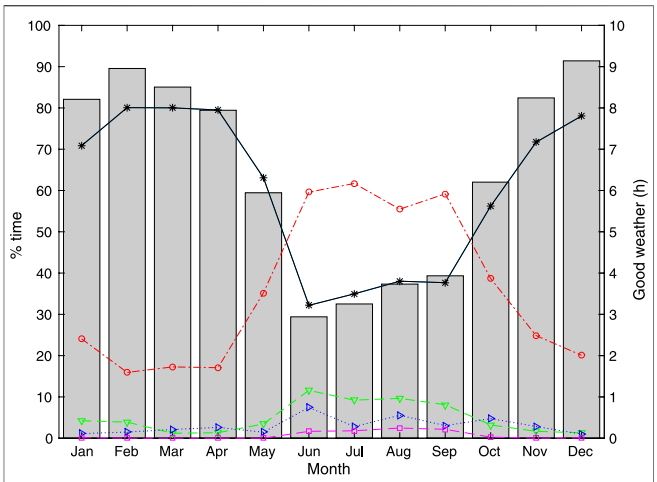


FIGURE 2 | Grey bars (right axis) show the average daily number of hours of good weather in each month. The color symbols (left axis) display the percentage of time of good weather (black asterisks) and observations not allowed by clouds (red circles), rain (green downward-pointing triangles), wind (blue right-pointing triangles), and humidity (magenta squares).

If any of these limits are exceeded, the observations are aborted or not allowed, and the dome remains closed. After the weather parameters return below the limits, we wait for additional 15 min before the observations start again. In this fashion, we ensure stable conditions and avoid opening and closing the dome too often. We link this waiting time to the last bad-weather event in the following analysis.

Between 2015 and 2021, 61% of the available time had good weather, 35% of the time was cloudy, and it was rainy 5%, humid 1%, and windy 3% of the time. **Figure 2** shows the monthly percentage of the time with good weather. Also, it displays the frequency of the different conditions that prevent the observations. We also provide the average total number of hours per night available for astronomical observations; as obvious from **Figure 2**, we have typically 8 h in the winter (November-April) and 3–4 h in the Summer (June-September)

In contrast to other northern observatories with a worse winter and better summer conditions, we have excellent winter conditions at La Luz Observatory, with more than 70% of the time with observable conditions between November and April, and reaching 80% of the time in several months. The wet season starts very fast, with June being the worst month for the observations, with a short transition period in May. Between June and September, only 30–40% of the time can be used for the observations.

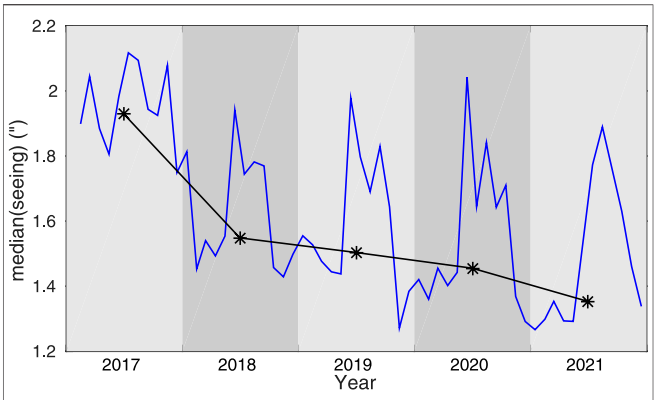


FIGURE 3 | Monthly median of the seeing observed by TIGRE between 2017 and 2021 (blue line). The black asterisks and lines show the yearly medians of the seeing.

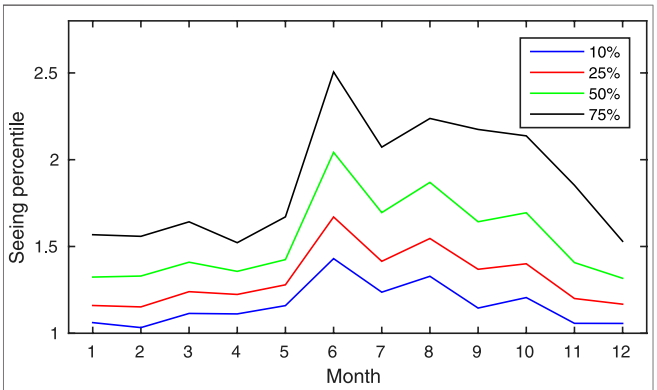


FIGURE 4 | Statistics of the seeing FWHM observed by TIGRE. The 10%, 25%, 50%, and 75% percentiles of the seeing distribution in each months are displayed.

The main reasons for the bad weather are clouds, around 20% in the winter and 60% in the wet seasons. On the other hand, rain is present around 10% of the time in summer. The wind is usually less critical, but as described above, it can be an issue in summer when it is clear.

3.2 Seeing Statistics

Even for a spectroscopic telescope like TIGRE, good seeing is essential. The diameter of the microlens in front of the fiber entrance is 3″, implying that a large amount of photons is lost if the seeing becomes too large. To compute how many photos are lost by such an aperture, we assume a seeing profile modeled as a Moffat function as follows:

$$F(r) \propto \left[1 + \left(\frac{r}{\alpha}\right)^2\right]^{-\beta}, \quad (1)$$

with α and β are seeing dependent parameters (FWHM= $2\alpha\sqrt{2^{1/\beta}-1}$). Assuming $\beta = 3$, only 25% of the photons fall inside a centered circle of diameter 3″ with a seeing of FWHM = 4″. This fraction of photons increases to

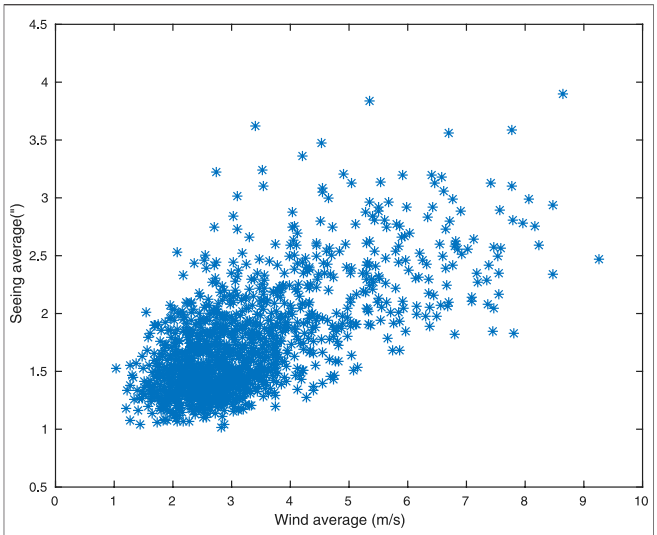


FIGURE 5 | Relation between the nightly averages of seeing and wind velocity.

38%, 60%, and 91% respectively, when the seeing is 3″, 2″, and 1″. Although poor seeing reduces the number of detected photons significantly, it is not necessary to have excellent seeing to collect most of the photons.

We have collected the seeing values from 2017 to 2021 measured by the guiding camera during the acquisition and guiding processes. The scheduler favors the observations close to the meridian, but many observations are nonetheless taken at lower elevations. Seeing does depend on elevation; following the Kolmogorov turbulence model, one expects the seeing to be proportional to the airmass (X) as seeing $\propto X^{(2/3)}$ (see e.g., Schroeder, 2000). Using this correction, we show in **Figure 3** the evolution of the monthly median seeing at zenith between 2017 and 2021. There is an apparent variation inside a year and also an unexplained and intriguing decline of the yearly-averaged seeing in the last years.

As is the case with many other weather parameters, the seeing also depends strongly on the season. The median values range from 1.4″ in January to 2″ in June (**Figure 4**). The monthly median is below 1.5″ between November and May. Furthermore, the seeing is better than 1.5″ for more than 60% of the time and even below 1.2″ for 30% between December and May. On the other hand, in June, only 18% of the time the seeing is below 1.5″. The season with bad seeing starts in June and continues until October when the median seeing is 1.7″ and only 37% of the time it is below 1.5″.

However, these measurements have some limitations. First, the seeing also depends on the wavelength, being smaller for redder wavelengths. Since the guiding camera has no color filter, the measurements also change with the star's color. Second, there is a residual aberration in the optics of the telescope. This aberration is apparent when the seeing is below 1.5″, so the lower values of the seeing measured by TIGRE ought to be considered as upper limits of the atmospheric seeing. Furthermore, the seeing of an observatory is usually measured

with a differential image motion monitor (DIMM), which operates at a particular wavelength (commonly 5,000 Å) and is independent of tracking errors and instrumental effects. For these reasons, we have not compared our results with other observatories.

As mentioned above, we have noticed that the nights with strong wind always have poor seeing. **Figure 5** shows the nightly-averaged seeing against the nightly-averaged wind velocity. Both parameters present a clear correlation, although this does not imply that the wind is directly causing the turbulence. In fact, usually at some point during windy nights, the wind velocity decreases to lower values, but the seeing remains high for several additional hours. It seems that it is the weather situation that produces strong winds also causes the atmospheric turbulence to be stronger and therefore causes more prominent seeing.

3.3 Photometric Nights

Finally, we analyze here the photometric quality of the La Luz Observatory. Such an analysis is usually performed by estimating the photometry in a particular filter with a large aperture for many standard stars at different elevations. If the relation between the magnitude with the airmass is linear with small dispersion, the night can be considered photometric.

Since TIGRE is a spectroscopic telescope, it is very challenging to evaluate whether a night is photometric or not. The reasons are, first, the inherent difficulty of photometric calibration of echelle spectra, which is aggravated by the diameter of the fiber entrance: the fraction of photons feeding the spectrograph depends on the seeing as described above. Second, the guiding camera has no color filter, so the dependence of the instrumental magnitude on the star's color is strong and complex. This makes the guiding images in principle useless to estimate the photometric quality of the night. Third, we do not know the accurate values of the magnitudes of the stars; the great majority of them are not photometric standards, and many are actually variables.

As a consequence another method must be used. Here, we calculate the statistic of the guiding camera photometry for each star observed in the night. The guiding cycle takes about 14 sec, from which 10 s is the exposure time and the rest the time needed to readout and process the image and offset the telescope to the new position. Also, TIGRE spends most of the time in observations that last between 10 and 75 min. Thus, there are between 40 and 300 flux estimates with the guiding camera for each star observed.

To evaluate the photometric quality of the night, we first calculate the standard deviation of the instrumental magnitude taken with the guiding camera for each star observed. For very long HEROS exposures, for which the airmass of the telescope has changed by more than 0.1, we corrected the magnitudes from extinction, linearly fitting the instrumental magnitude against the airmass and first removing the linear fit before the standard deviation is calculated.

We consider the night as photometric if the standard deviation of the instrumental magnitudes of the different stars is < 0.01mag for at least 90% of the time, and we have data for a minimum of 6 h for each night. This method ensures that no clouds are obscuring the stars during the observations. However, this technique does not account for slow changes in the atmospheric extinction or variations depending on the direction of the observation.

During the years 2018–2021, we have obtained an average of a total of 40 photometric nights per year using this technique. The best months are December until February, with around 7 photometric nights per month. On the contrary, conditions are much worse in summer, when we have an average of less than two photometric nights per month between June and October. Since TIGRE's instrument is a spectrograph, the photometric quality of the observatory, despite not being excellent, does not change the capabilities of TIGRE significantly. Echelle spectra are complicated to be photometrically calibrated. The only possible consequence of a non-photometric night is thus to extend the exposures to reach the desired signal-to-noise ratio (see below).

4 PERFORMANCE AND EFFICIENCY OF TIGRE

From a technical point of view, our goal is to expose as many good-quality spectra as possible. Here we describe how we have quantified the performance of TIGRE in these 8 years and how we have pursued this goal. There are different aspects to be considered when talking about the performance:

- *Quality of the spectra:* The spectra obtained should meet the requirements of the astronomer.
- *Optical efficiency of the system:* From the incoming photons at the telescope's aperture, as many as possible should be detected in the spectrograph.
- *Available time:* The astronomical (Sun below an altitude of -10°) and meteorological conditions are constraints out of our control, and we have already discussed the weather above.
- *Technical issues:* Problems in the hardware are a major impediment for the observations, particularly in robotic telescopes, because usually no human intervention is directly possible when the issue appears. Some problems can be solved by the robotic operator and cause only a moderate amount of time loss. However, others are severe and may stop the activity of the facility possibly for months.
- *Operational efficiency:* This measures the fraction of the available time that is actually spent on exposures.

In the following subsections, we will describe in detail all of these subjects.

4.1 Quality of the Spectra

We have a signal-to-noise calculator, yet it only gives reliable values when the atmospheric conditions are relatively good, and the actual conditions, under which the observations are carried out, are obviously unknown when the exposure is calculated. To avoid taking too many underexposed spectra in case of poor conditions, it is of utmost importance to extend the exposure time

to reach the desired signal-to-noise ratio (SNR). Hence, to assure a well-exposed observation, the astronomer can request a minimum SNR at a particular wavelength and suggest the number of exposures in which the observation is divided and the exposure time of the individual exposures. Then, if necessary, TIGRE exposes longer to reach the required SNR, and so it adapts the exposure to the changing atmospheric conditions of seeing and transparency.

To estimate the SNR, the number of analog-to-digital units (ADUs) in a particular order in each channel (*CT*) is calculated once an exposure is taken[2]. TIGRE then estimates the SNR at the requested wavelength using these number of ADUs and the B-V color of the star.

The procedure that ensures that the requested SNR is reached, starts with the star acquisition. The instrumental magnitude of the star estimated by the guiding camera gives information about the transparency of the atmosphere. With this magnitude, the B-V color of the star, and the seeing, TIGRE can predict the exposure time necessary to reach the SNR. If this predicted time is longer than the one suggested by the astronomer, TIGRE calculates the exposure time for the individual exposures. If necessary, the robotic operator increments the number of exposures. After each exposure, *CT* is calculated and compared with the expectation. After the predicted number of exposures are taken, the sum of *CT* for all exposures is compared with the corresponding value of the requested SNR. Then, another exposure is taken if the target value for the *CT* is not reached. The calibration of the relation between the SNR and both *CT* and the instrumental magnitude of the guiding camera has been fitted using many stars of different spectral types.

We re-emphasize that it is very easy for a robotic telescope to take data that do not reach the desired quality because no human evaluates the recently-acquired data and decides the necessary exposure time. With the method described here, we assure that TIGRE acquires good-quality spectra, regardless of the atmospheric conditions.

4.2 Optical Efficiency

Without specific instruments, the determination of the system's absolute optical efficiency or only the reflectivity of the mirrors is impossible due to the temporal variation of the atmospheric conditions. Despite this impediment, we attempted to monitor the changes in the optical efficiency of TIGRE using alternative methods. For this, we considered the relative variations of the measurements of the guiding camera or the acquired spectra. The guiding camera photometry has the advantage that it does not depend on the seeing since one can calculate the magnitude with a large aperture. However, these data with large aperture were logged only over the last years. Also, we installed a new (and different) guiding camera in May 2019, and there is no easy method to compare the data taken with the two guiding CCDs. The use of the spectra, on the contrary, depends strongly on the seeing, but has the advantage that it is possible to compare the data taken now with data taken years ago. Also, the spectral measurements keep track of variations not only in the reflectivity of the mirrors of the telescope but also in the efficiency of the whole optical system, including the optical fiber, collimator, echelle grating, and cross-disperser of the spectrograph.

The main disadvantage of the spectral measurements, i.e., the dependence on the seeing, can be overcome by calculating how much light is lost outside the fiber entrance. We have calculated the flux losses using the difference of guiding magnitudes obtained with a large aperture (with a diameter of $6''$) and an aperture with a diameter of $3''$. This difference depends on the seeing and has been used to correct the data of the spectra.

Another problem of this method is that we do not know the precise magnitude of the stars. Many stars are variable, and for others, the magnitudes obtained from SIMBAD are not accurate enough. To solve this problem, we have used our sample of flux-calibration stars and their GAIA photometry (Casagrande and VandenBerg, 2018; Gaia Collaboration et al., 2018)[3]. This is a very small sample of stars, and one of them is observed every night whenever possible. Another advantage of this sample is that they all have similar colors, $-0.2 < \text{B-V} < 0.00$, which eases the analysis since the color dependence is negligible.

Thus, we calculated the magnitude M_{spec}, as a seeing corrected magnitude of the counts of the spectrum (as estimated above, *CT*) compared with the GAIA magnitude of all stars in the sample and presented their temporal variations in **Figure 6**. There are significant variations from day to day due to the different atmospheric conditions. Still, when the conditions are excellent, the trends in the upper envelope are easy to notice. The big jumps in May 2019 and July 2021 resulted from the washing of the mirrors, and the red lines mark the time when a CO_2-snow cleaning was done.

In many telescopes, such as, WHT or VLT, the mirrors are cleaned with CO_2 snow (Zito, 1990). The cleaning process occurs when the snow quickly sublimates on the mirror surfaces and the particles of the mirror are blown away. This process is not abrasive and free of residues. We try to apply this cleaning once per month.

We wash the mirror appoximately once per year with distilled water and a drop of dishwashing detergent. After washing the mirror, it is gently dried with absorbent paper.

The mirrors become dirtier with time. Also, the coating of the mirrors ages and deteriorates, so it is expected that the optical efficiency decreases with time, as is apparent in **Figure 6**. Apart from this, washing is an excellent method to recover a large part of the reflectivity losses. Also, CO_2-snow cleaning reduces the losses in reflectivity. **Figure 6** shows that the loss of optical efficiency is much faster when we could not clean the mirrors due to technical problems with the cleaning device. Curiously, the CO_2-snow cleaning is not always equally effective; in summer and autumn it is less effective than in winter and spring. However, if one looks carefully, the reflectivity losses are also higher in the winter and spring. This is because the dome stays open longer in

[2]The wavelength ranges of the measured bands are $\sim 5315\mathring{A} - 5355\mathring{A}$ in the blue channel and $\sim 7250\mathring{A} - 7300\mathring{A}$ in the red channel.

[3]Our flux calibration stars are HR153, HR1544, HR3454, HR4468, HR7001, and HR8634.

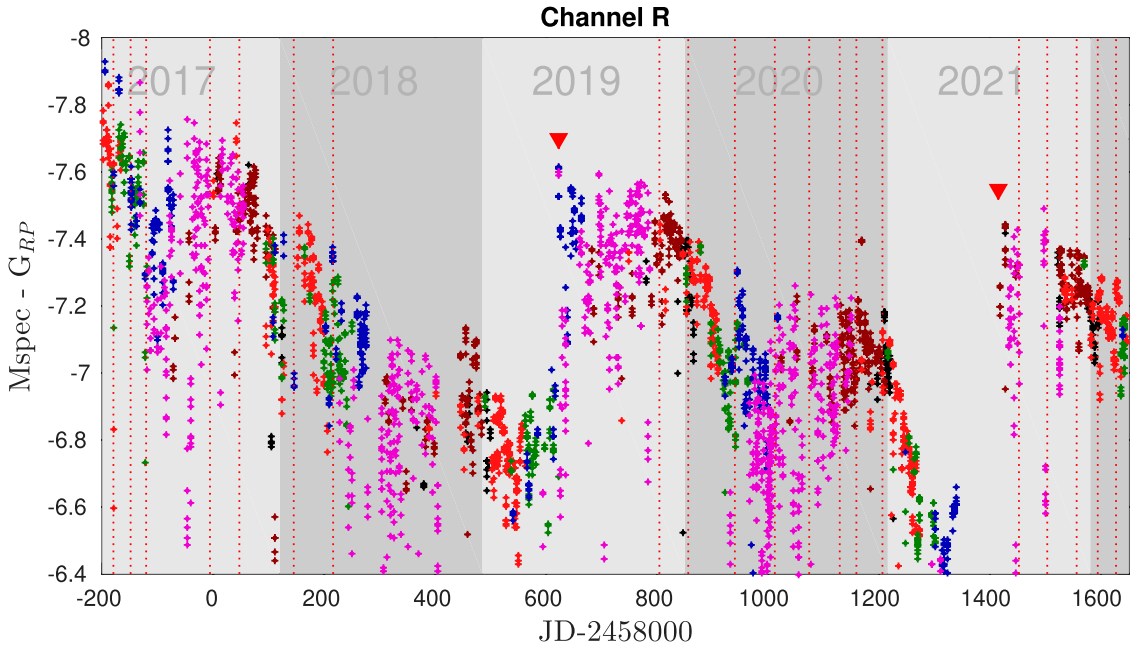


FIGURE 6 | Relative variation of the optical efficiency of the TIGRE-HEROS system. We show the difference of the GAIA magnitude and a seeing-corrected magnitude calculated with the HEROS spectral counts in a ca. 100 Å-wide region around 7,300 Å. The symbols with different colors represent the measurements of the six flux-calibration stars. The red triangles display the times when the mirrors were washed, and the red vertical dotted lines depict the times when the mirrors were CO_2-cleaned.

the first months of a year, while this is not the case in summer. This, combined with the larger amount of dust in winter and spring due to less rain and dry soil, explains the more significant deposition of dust on the mirror of the telescope in these months; this new deposition of dust is what the CO_2-cleaning can remove.

4.3 Technical Issues

TIGRE has suffered from several issues that limited or hindered its functionalities in these 8 years. Here we can classify the problems regarding their severity:

- Minor issues that can be detected and solved by the error handler while the robotic operator continues working imply only a time loss of seconds or minutes. However, they may add to hours of time lost per month if they appear too frequently.
- Other issues cause the robotic operations to stop completely, with time losses from hours to days. Examples of these are internet cuts and power failures. For some of these faults, human intervention is necessary to solve the problem.
- Finally, severe issues can stop the observations for days to months. These are usually hardware failures that need the purchase of spare parts, and even the travel of a team from Hamburg to Mexico to repair the broken components.

Although some issues are not avoidable, prevention helps to minimize the time lost. Some efforts have been made in this direction. First, we have continuously developed the error detection procedures and error handler. New issues appear occasionally, and our

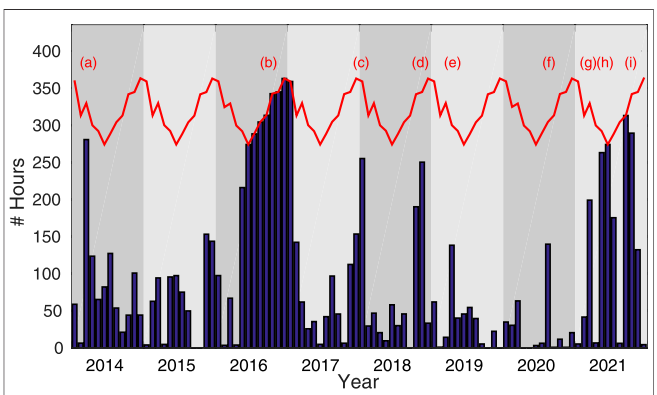


FIGURE 7 | Number of hour per month lost due to technical issues (bars). The red line display the total number of night hours. Specific technical problems caused the events labeled (a) to (i) as detailed in the text.

experience in the interactive solution is used to design a detection procedure and a solution that the error handler can perform. This is particularly important for the minor issues described above. Also, we have installed switchable sockets and remote-controlled USB hubs. With them, we can switch off/on any computer or device that has hung up. We also reboot the computers regularly, once per week, to avoid them from getting hung up when the different programs continuously run for a very long time.

Apart from the continuous improvement in the software, we have purchased several spare parts, in particular for the electronic cabinet. These include some relays, fuses, and other spare parts

that the local staff can replace, and also a spare computer with its cards, as well as Profibus cards for the azimuth and elevation axes of the telescope. As a result, some of the severe technical faults we encountered were solved relatively quickly because we already had the spare parts at the telescope. This is particularly important for some essential components of the telescope electronics because they are no longer manufactured.

One of the long-standing problems for TIGRE is the power supply. Power outages do occur frequently, up to several times per month, and even more often in the rainy season. Since the power plant does not work very reliably, TIGRE closes and stops the robotic operation, and the system remains in idle status after an extended power failure (longer than 5 minutes), to avoid that the telescope stays open when the batteries of the UPS are out. In this last case, when the failure lasts several hours, the UPS does not restart automatically when the power returns, and the UPS needs human intervention to power up the system. However, the external power supply has improved in the last years: while we had ~25 power cuts per year at night time between 2015 and 2019, the frequency reduced to ~15 power cuts per year afterward.

Figure 7 shows the number of night hours lost by technical issues in these 8 years. To collect this information, we have extracted from the log files of TIGRE the amount of time when the robotic operator should be running, but it was not. In this statistic, the weather is not considered as long the system was running. The availability, as defined by Sybilski et al. (2014), is the fraction of the night time when the system runs without any issues. The availability of TIGRE has increased in the last years, thanks to the measures described above, except for the severe issues labeled in the figure listed below:

(a) Hydraulic oil spillage.
(b) Mirror cell refurbishment and mirror aluminization.
(c) Failure in the calibration unit (flat field and ThAr).
(d) GPS antenna failure during a thunderstorm.
(e) Profibus card failure.
(f) GPS antenna failure during a thunderstorm.
(g) Failure in the hydraulic pump.
(h) Fault in the focus-controlling card
(i) UPS batteries in poor conditions.

4.4 Operational efficiency

We define operational efficiency as the fraction of the available time with good weather that is used in exposures. With this parameter we measure how well the available time of TIGRE is used. This parameter depends, of course, on the number of stars observed in the night. More stars with shorter average exposure times imply a lower operational efficiency because the overhead time, related to the number of stars observed, is larger. This overhead time includes the telescope's movement, the acquisition of the star, the beginning of the guiding, and the readout time. Technical issues as described above reduce also the operational efficiency.

This effect of the overhead time in the operational efficiency is unavoidable and comes from the requirements of the astronomer and the atmospheric conditions. However, other factors reduce

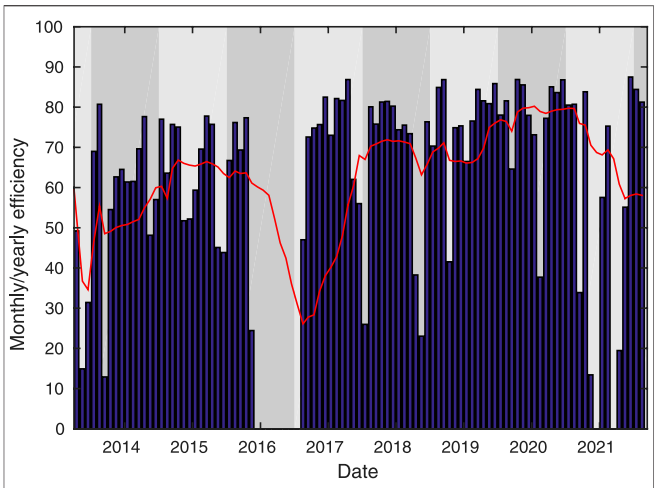


FIGURE 8 | Operational efficiency of TIGRE (bars). This is defined as the ratio between the total exposure time and the good-weather time. The red line shows the running 12-months average of the operational efficiency.

the operational efficiency and are not desirable. Some of these factors are minor issues. After they are detected, the error handler often interrupts the robotic operation while solving them. Sometimes this involves restarting a device or even pointing the telescope and acquiring the star again.

Another factor is related to the weather conditions. Worsening of the weather often implies that the current observation is aborted. Unfortunately, the spectrograph CCDs do not allow to abort a running exposure and readout the collected electrons by then, which means that an aborted exposure is lost. To minimize this effect, we abort the exposures due to poor weather when the exposure is already performed by less than 70% of the time. Besides, after a bad-weather period, when the conditions allow the observation, some time is needed to open the dome and initialize the telescope. which also reduces the efficiency.

To reduce the time lost due to the weather, we try to avoid observations with low chances to be performed successfully. This is accomplished by both the robotic operator and the scheduler.

- We avoid long observations in case of unstable atmospheric conditions. A straightforward approach is to select only targets whose observation takes not longer than the good-weather time. The latter is defined as the time since the last bad-weather event.
- The second possibility is to avoid the observations that require a minimum SNR and the calculated exposure time extends too much. Thus, observations that do not need much time are preferred, and hence they have a lower probability of being aborted because of poor weather or the end of the available time. The time necessary to reach the required minimum SNR is calculated to fulfill this condition. If the predicted time is much longer than the exposure time suggested by the astronomer, the observation is not performed. Therefore, the predicted exposure time should be:

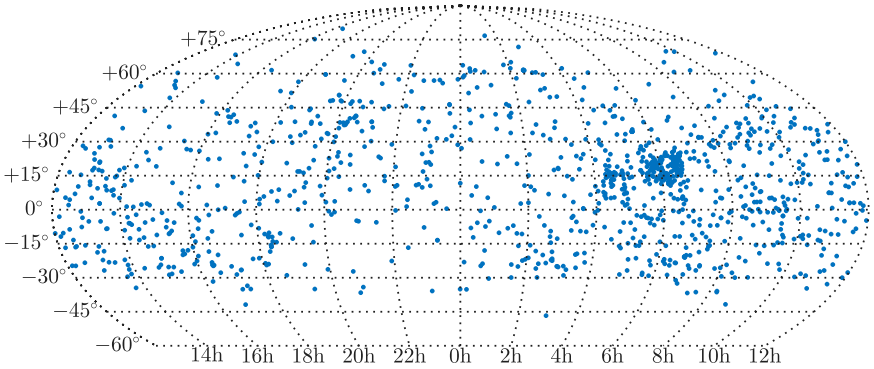


FIGURE 9 | Location on the sky in a RA-Dec projection of all stars observed by TIGRE.

$$T_{\exp,pred} < f * T_{exp,sugg}, \quad (2)$$

where $T_{\exp,pred}$ and $T_{\exp,sugg}$ are the predicted exposure time using the current conditions and the suggested exposure time. The factor f depends on $T_{\exp,sugg}$; it varies from $f = 8$ for $T_{\exp,sugg} <$ 5 min to $f = 1.3$ for $T_{\exp,sugg} > 1.5$ h. The predicted exposure time is calculated twice. First, the scheduler, knowing the last average conditions, i.e., extinction and seeing during the last observations, can predict the necessary total exposure time for the observation of each target to be successfully completed. In the next step, the robotic operator, using the information of the instrumental brightness obtained with the guiding star and the seeing, also predicts the necessary exposure time. When the predicted exposure time does not comply with **Eq. 2**, the scheduler does not select the target, or the robotic operator aborts the running observation and asks the scheduler for a new target.

In the future we plan to use weather forecasts to improve and optimize the selection of the targets to be observed, favoring some targets against others, when we have information about how the weather will be in the following hours or even days.

Figure 8 shows the evolution of the operational efficiency in these 8 years. Since 2017, the operational efficiency is high, usually higher than 70% on average and often > 80%, except for the periods that coincide with technical problems as described in Sect. 4.3. This performance has increased in the last 4 years due to the continuous improvement in software and hardware. In addition, the improvement in the operation described in this section has reduced the number of aborted spectra. Also, since more devices can be accessed and rebooted remotely when necessary, we can solve minor issues without waiting for the staff to drive to the observatory, reducing time losses.

5 SELECTED SCIENTIFIC RESULTS

In this section we provide a short overview over (some of) the scientific output produced by TIGRE since its operational start in 2013. In addition to quantitative measurements we describe some of the scientific results that especially highlight the robotic capabilities of TIGRE.

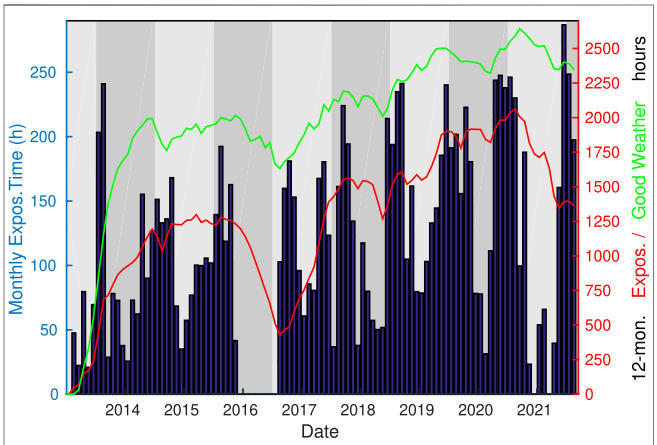


FIGURE 10 | The bars (left axis) show the monthly exposure time (in hours) since the start of TIGRE's operations. The lines (right axis) depict the 12-months total exposure time (red line) and the 12-months total number of good-weather hours (green line).

TABLE 1 | Number of nights with scientific spectra, total number of spectra and total exposure time for each year; note that in 2016 the mirror was re-aluminized, which explains the lower numbers.

Year	Nights	Spectra	Exp. time(h)
2013	60	1,692	240
2014	211	5,176	1,156
2015	242	5,870	1,233
2016	109	2,297	655
2017	239	6,428	1,391
2018	238	5,778	1,359
2019	295	8,738	1901
2020	308	7,455	1990
2021	187	5,064	1,393
Total	1889	48,498	11,318

5.1 Collected Science Spectra

In the time frame 2013–2021 TIGRE has collected more than 48,000 spectra of 1,151 different sources with a total exposure time of more than 11,000 h. To give an idea of where TIGRE

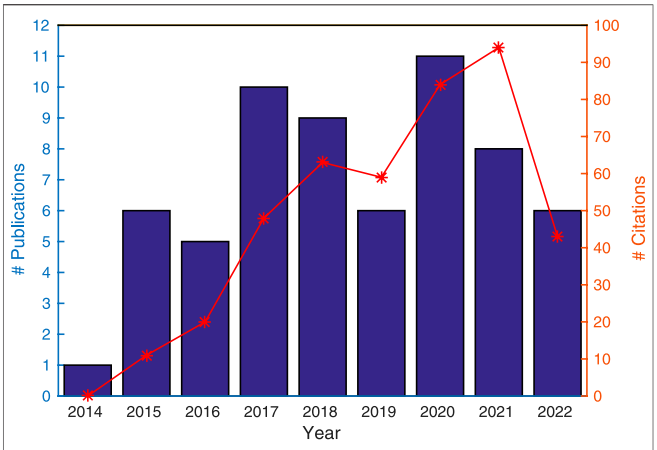


FIGURE 11 | Yearly number of TIGRE related publications (blue bars) and citations (red asteriks).

observes, we provide in **Figure 9** the distribution on the sky of all stars observed by TIGRE.

In **Figure 10** we show the monthly total exposure time in scientific targets, and in **Table 1** we provide an overview of the number of nights with at least one scientific spectrum, the total number of spectra and total exposure time obtained with TIGRE. As is clear from **Figure 10**, the total exposure time has continuously increased, except for 2021, when we were plagued with severe hardware failures. In 2019 and 2020, TIGRE obtained more than 1900 h of exposure per year, which is a considerable improvement compared to the first years, when we could expose only ~1,200 h. Also, the number of spectra taken has increased, from ~5,500 to ~8,000. However, this number is obviously less meaningful for the telescope's performance, because it also depends on the requirements of the astronomers and the stars' brightness. **Figure 10** also shows the evident seasonal variations as well as the rising trend. In the summer, the total exposure time per month is reduced by a factor of three due to the above described bad weather conditions between June and September.

5.2 Publications

Publications in recognized journals are clearly an indicator of the success of any telescope facility. We are maintaining a web site under https://hsweb.hs.uni-hamburg.de/projects/TIGRE/EN/hrt_publication.html, which is updated once or twice a year and lists all publications involving TIGRE data. At the time of writing (March 2022) we have 62 such publications. In **Figure 11** we provide a graphical display of the temporal evolution of the TIGRE publications and citations.

5.3 Stellar Parameters and Activity of Cool Stars

Long term monitoring is the basic requirement for the observational study of stellar activity cycles analogous to the well-known 11 year cycle of the Sun. However, stars remain –almost always– spatially unresolved, and furthermore, lower activity stars like the Sun are typically faculae-dominated rather than spot-dominated. As a consequence, cyclic variations are easier to study in chromospheric emission lines using moderate- to high-resolution spectroscopy rather than broad band photometry. The lines of choice are the deep Ca II H&K absorption lines, already recognized by Fraunhofer (1817) during his early solar studies. The cores of the absorption lines at 3,968.469 Å (Fraunhofer's H line) and 3,933.663 Å (Fraunhofer's K line) exhibit very low residual photospheric flux, thus allowing the detection of the overall much fainter chromospheric emission against a rather low photospheric background.

Eberhard and Schwarzschild (1913) were the first to recognize reversals of the Ca II H&K line cores in the spectra of a few stars observed by them and realized the potential offered by these line reversals to study stellar cycles, writing: "It remains to be shown whether the emission lines of the star show a possible variation in intensity analogous to the sun-spot period". However, a systematic study of these line core variations in a larger sample of stars over an extended period of time was taken up only much later, in the 1960ies, by O. Wilson in his Mount-Wilson H&K program. Rather than using photographic plates as done by Eberhard and Schwarzschild (1913), O. Wilson employed a specially designed H&K photometer, which used far more sensitive photoelectric photometers to measure the flux in two triangular spectral bands with a width of a little over 1 Å centered on the wavelengths of the H and K lines. To remove effects of seeing and variable atmospheric emission, the measured line core fluxes are provided relative to the flux in two nearby continua redwards and bluewards of the H&K lines as so-called S-indices. The first results of this monitoring program were reported by Wilson (1978), where he (tentatively) concludes that:" 1) no stellar chromospheres are likely to be constant in time; 2) short-term fluctuations tend to increase in size with average flux; 3) cyclical variations occur with periods ranging from about 7 years to probably at least twice as long; 4) the stellar cycles observed in H and K flux should be regarded as evidence for analogs of the solar cycle; and 5) the incidence of complete or probable partial cycles increases toward later spectral types."

The more or less final results (although some observations were carried out later) of the Mount-Wilson program were presented by Baliunas et al. (1995), however, it was becoming clear at the time that long-term monitoring of late-type stars could be (and should be) carried out in a far more efficient and economic way by using robotic telescopes (thus saving on man power) and using high-resolution spectrographs, which made any individual settings (to adjust e.g., for the individual star's radial velocity) unnecessary. Furthermore, an automated data-reduction pipeline reduces the spectra efficiently and reliably and produces, for example, S-indices or other activity indices quickly and automatically. Furthermore, spectrographs make full use of the whole spectral range covered, while the H&K photometer "threw away" a large part of the actually recorded spectral information.

TIGRE encompasses all these features, but it can –of course– also provide S-indices which are calibrated against the extensive Mount-Wilson S-index data base. As an example, in **Figure 12** we

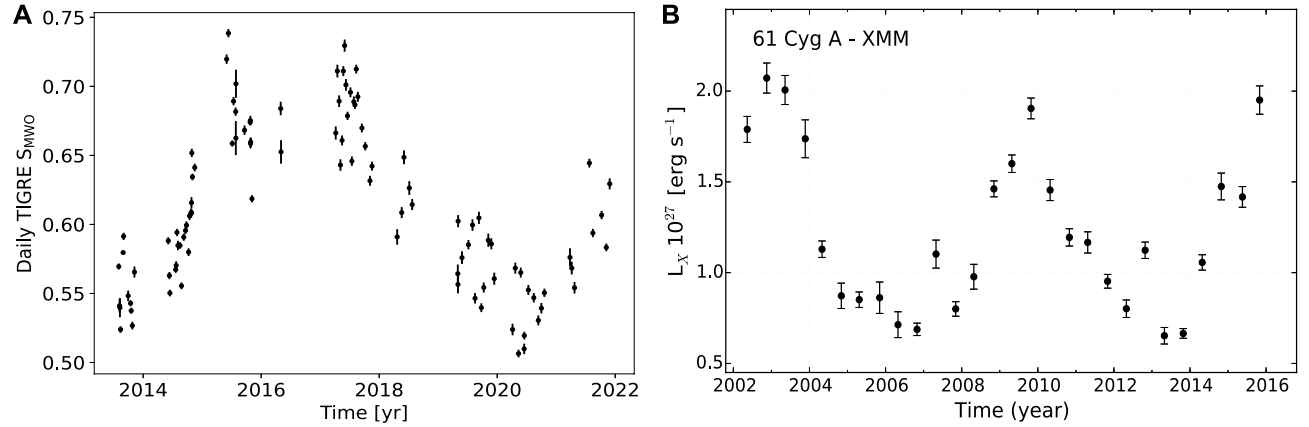


FIGURE 12 | (A): TIGRE S-index time series of the active star HD 201091; note the clear cyclic variability in the S-index **(B)**: Evolution of X-ray luminosity derived from *XMM-Newton* X-ray monitoring of HD 201091 between 2002 and 2016; note the cyclic variability of the recorded X-ray emission (courtesy J. Robrade).

show the S-index obtained from the TIGRE spectral time series of the active star HD201091 (= Cyg 61 A, spectral type K5V), which very beautifully demonstrates the cyclic nature of the chromospheric emission from HD 201091 with a cycle period of about 7 years and nicely confirms the Mount-Wilson results on the same star reported by Wilson (1978). HD 201091 is remarkable since we monitored the X-ray emission of this star using *XMM-Newton*. Due to operational constraints, X-ray monitoring is far less dense (typically only two observations per year) than optical monitoring, however, the long-term X-ray light curve of HD 201091, displayed in **Figure 12** shows a clear cyclic variability with the same period as the chromospheric emission. Once TIGRE went online in 2013, we accompanied the *XMM-Newton* observations with simultaneous TIGRE observations, and indeed, the chromospheric emission strongly increases between 2014 and 2016 in line with the increasing X-ray emission.

Furthermore, the comparison of observed spectra with synthetic spectra from model atmospheres yields the main physical parameters of the observed star, namely effective temperature, gravity, metallicity, rotation velocity and turbulent broadening parameters. TIGRE and the very homogeneous quality of its HEROS spectra, monitored each night as part of the reduction pipeline and regular calibration procedures, has proven very useful for such work on larger samples. The consistency of such an analysis for a whole sample depends a lot on the reproducibility of the spectral resolution, because a false consideration of the latter leads to confusion over the velocity parameters and gravity broadening, indirectly even affecting the effective temperature and metallicity determination, see Schröder et al. (2021). There is a good potential for characterizing larger samples of cool stars consistently, using TIGRE spectra, as would be of interest for, e.g., exoplanet host-stars and other samples of particular interest.

The reliance on homogeneous resolution in a sample of spectra is an especially sensitive issue with cool stars, and giants in particular, where spectral lines are intrinsically sharp (see Rosas Portilla et al., 2022; MNRAS, in print). A sample of TIGRE spectra for over 30 cool giants have already helped the revision of the physical relations behind the Wilson-Bappu effect, for which we measured the Ca II K emission line width in the blue channel spectra and used the red channel, which is providing a less line-crowded and so easier to analyze spectral range, for the physical parameter analysis.

5.4 Hot Stars

The instrumental stability and scheduling flexibility of TIGRE are major assets that are beneficial to a number of different studies. To illustrate this point, let us consider the case of a peculiar class of massive stars, the γ Cas stars. These objects are Be stars, i.e., rapidly rotating B-type stars surrounded by a Keplerian circumstellar disk containing material ejected by the star. Whilst the majority of the Be stars display a comparatively weak (log $L_X/L_{bol} \sim -7$) and soft ($kT \sim 0.2$–0.6 keV) X-ray emission, the so-called γ Cas stars feature a bright (log $L_X/L_{bol} \sim -6$ to ~ -4) and unusually hard (kT ~ 12 keV for γ Cas) X-ray emission (Smith et al., 2016). To date, about two dozen objects are known to belong to this category (Nazé et al., 2020). Various scenarios have been proposed to explain their peculiar X-ray properties, including accretion of the Be disk material by a compact companion, wind interactions with a hot stripped helium star companion, or magnetic interactions between the Be star and its disk. In this context, investigating the multiplicity of γ Cas stars is crucial to discriminate between the various hypotheses. Over the last 8 years, we have used TIGRE to monitor the optical spectra of a sample of γ Cas stars. Thanks to its robotic mode, TIGRE enabled us to collect well-sampled time series of spectra which were used to perform a systematic radial velocity monitoring of these stars (Nazé et al., 2022). This study showed that the properties of the known and newly found γ Cas binaries do not significantly differ from those of other Be binary systems and that companions are of low mass (0.6–1 $M_\odot$).

Beside the multiplicity investigation, the TIGRE spectroscopic monitoring allows to characterize the changing properties of the Be disk via the strength and shape variations of a variety of optical (H I Balmer series, He I, Fe II) and near-IR (H I Paschen series) emission lines arising within the disk. Indeed, establishing the link between the Be disk properties and the peculiar X-ray emission is an important

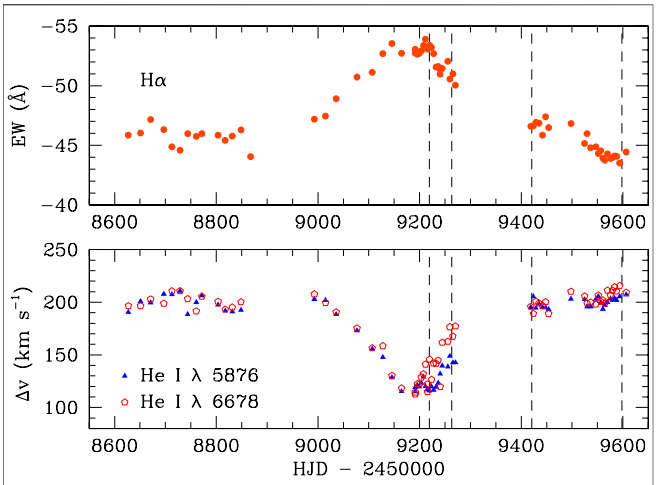


FIGURE 13 | TIGRE monitoring of γ Cassiopeiae between May 2019 and January 2022. The top panel illustrates the equivalent width (EW) of the Hα emission line as a function of time, whilst the lower panel displays the velocity separation (Δv) between the violet and red peaks of the double-peaked emission components of the He I λ 5,876 and λ 6,678 lines. The outburst around HJD 2459180 (January 2021) is clearly seen in EW(Hα) and via a significant change in Δv, indicating that the formation region of the He I lines had expanded significantly. The dashed vertical lines indicate the times of our *XMM-Newton* ToO observations.

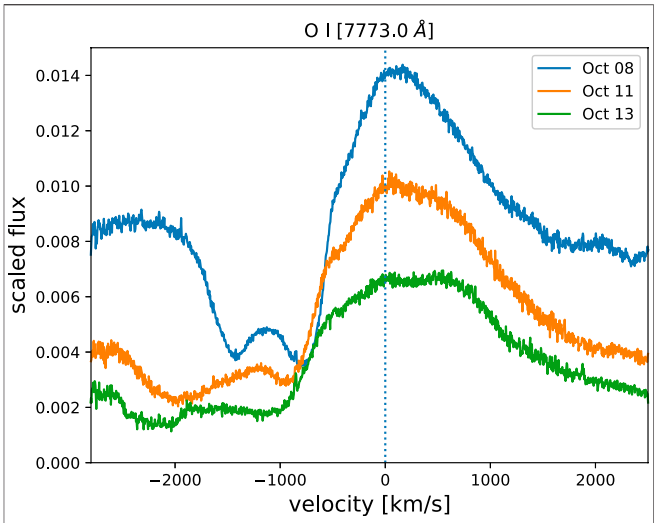


FIGURE 14 | Three spectra of Nova V6593 Sgr in the O I line at 7,773.0 Å observed in 2020; note the rapid changes in the recorded line profiles, indicating the increase in expansion velocity.

key towards understanding the origin of the hard X-ray emission. A coordinated monitoring with TIGRE and the *Swift* X-ray telescope over three orbital cycles of π Aqr unveiled no correlation between the X-ray emission and either the orbital phase or the Hα emission strength (Nazé et al., 2019).

Moreover, the TIGRE spectroscopic time series allow us to trigger target of opportunity observations with the *XMM-Newton*, *Swift* or *Chandra* X-ray observatories when the Be disks undergo a major transition, i.e. either an outburst or dissipation. Once these X-ray observations are scheduled, TIGRE further offers the possibility to collect optical spectra nearly simultaneously with the X-ray data. This strategy has been applied with success to the cases of a disk fading event in HD 45314 (Rauw et al., 2018), a cyclic outburst of the normal (i.e., not γ Cas) Oe-star HD 60848 (Rauw et al., 2021), and most recently an outburst of γ Cas itself (see **Figure 13**). The multi-wavelength data collected in those campaigns favour scenarios where the X-ray emission arises from the innermost parts of the Be disk, unlike what one would expect if the X-ray emission were due to interactions with a companion.

5.5 Transient Objects: Novae and Supernovae

A further strength of TIGRE is its quick response time for observations of targets of opportunity. Once we receive information about a newly discovered event such as a nova or supernova, we can usually observe the respective target already during the next night weather and sky position permitting. As a consequence we have been able to carry out successful observation campaigns of several supernova and nova targets.

Needless to say, given TIGRE's telescope aperture of 1.2 m, we can observe only the brighter novae and supernovae, yet during the past 8 years of TIGRE observations, we were able to observe two bright supernova events. The very bright supernova SN 2014J was observed with a long time series during the dry winter season in January and February of 2014 (Jack et al., 2015a). With TIGRE we took in total 43 spectra during a period of over 2 months, and this densely sampled time series helped identify the features that cause the secondary maximum in the near-infrared light curves of supernovae of type Ia (Jack et al., 2015b). We also took six spectra of the Supernova SN 2017eaw. However, this supernova was too faint for TIGRE observations, and we only obtained spectra with an acceptable S/N in the red channel. Thus, supernovae are in general difficult targets for a small sized telescope like TIGRE, but given bright enough SNe (such as SN 2014J) the spectral time series that can be obtained are quite unique.

Observations of galactic classical novae are much easier for TIGRE, since these events are much brighter and also more numerous. Our first TIGRE nova (i.e., the Nova V339 Del) was already observed in 2013 during the first months of TIGRE observations (De Gennaro Aquino et al., 2015). The longest nova time series was taken for Nova V5668 Sgr, for which we were able to obtain more than 200 spectra during a period of about 2 years. This nova of the type DQ Her showed rapid variations in both the light curve and spectra during the first 100 days after the outbreak. With TIGRE we were able to study the rapid changes in the spectral features of several lines during that phase in detail (Jack et al., 2017). Another time series was collected of Nova V659 Sct (Jack et al., 2020), for which we obtained eight spectra. A further nova that we observed was Nova V1112 Per, for which we obtained in total 34 spectra, and another twelve spectra were obtained of the Nova V6593 Sgr. The most recent observation of a TIGRE spectral time series are 16 spectra of the 2021 outbreak of RS Oph. Thus, we took in total over 300 spectra of six galactic novae during the 8 years of TIGRE observations. Another aspect of the intermediate resolution spectra of TIGRE is that it allows us also to study interstellar

absorption lines and diffuse interstellar bands in the nova spectra (Jack and Schröder, 2019).

To demonstrate the advantages of the TIGRE spectral time series observations we present as an example in **Figure 14** the evolution of the very common spectral line of O I at 7,773.0 Å of Nova V6593 Sgr. There are two absorption features at about −800 and −1,500 km s^{-1} in the spectrum from October 8th in 2020. These two features move during the following days to higher expansion velocities. This characteristic evolution was also observed in other novae. The robotic operation and the intermediate resolution of the HEROS spectrograph make TIGRE an ideal telescope for the observations of time series of novae and supernovae, and we are eager to observe more supernovae and novae targets of opportunity in the coming years.

6 SUMMARY AND OUTLOOK

In the present paper we have described our experiences collected in 8 years of operating TIGRE in a fully robotic mode, and here we attempt to summarize our main findings. We can state that our international TIGRE collaboration was really fruitful scientifically and extremely helpful operationally; having one of our partners on site, was essential for all logistic support and for trouble shooting in the cases of unexpected failures. And finally, at the La Luz site there is manpower ("vigilantes") who could be contacted in the case of absolute emergency.

Needless to say, a reliable internet connection is a "conditio *sine qua non*" for any robotic telescope, and it took some measures and time for TIGRE to arrive at that point. Reliable power supply is a further essential requirement, and the power supply at the La Luz Observatory site has definitely room for improvement. During the summer time thunderstorms are a constant source of concern, and further, the nominal power frequency of 60 Hz is not that stable. We have hardware to compensate for power failures in the form of batteries and a Diesel generator, but the reliability of the system can certainly be improved. On the long run, the use of solar power could make TIGRE self-sufficient power-wise, at least to a large extent.

While we can counteract at least some power supply problems ourselves, little can be done as far as dust contamination is concerned. The telescope is already mounted on a rather high pier (cf., **Figure 1**) to avoid the worst of dust, however, the main reduction of dust production occurred as a consequence of the nearby road being paved, which was actually beyond our control.

TIGRE has only one scientific instrument, i.e., its HEROS spectrograph. Although there are quite a few robotic telescopes in operation, only very few of them deal with spectroscopy; this is clearly TIGRE's unique selling point. While the spectral resolution of HEROS is "only" a little over 20,000, we feel that this resolution is entirely sufficient to address a wide range of scientific questions. Obviously, for some applications, such as high-precision RV work or Doppler imaging, HEROS, is not well suited, however, one also has to keep in mind that the phase space for such observations is quite small for telescopes in the 1 m class, and, for example, meaningful Doppler images can be produced only for a relatively small number of sources. What has proven beneficial, is the dual arm nature of HEROS. It provides, first, redundancy when one channel fails, and, second, allows the independent operation of the two channels which is very advantageous, for example, for rather red objects with small emissions in the Ca II H&K line cores with different exposure times in the red and blue channels.

Scientifically one may ask the question of what kind of role a (robotic) telescope of the 1 m-class can play in the epoch of 10 m (and in the future even larger) telescopes. In our view, our hitherto TIGRE record shows that indeed a scientific niche can be found for a robotic facility that is capable of producing valuable and visible scientific results. While essentially all of the TIGRE observations could in principle also be carried out by larger telescopes, the operation of larger facilities is far more costly, so in practice the spectral time series produced by TIGRE are unique. Similar considerations apply to the observations simultaneously carried out by TIGRE with space-based observatories such as XMM-Newton or eROSITA; TIGRE can carry out such observations in a rather straight-forward and uncomplicated fashion in an automated manner. For all practical purposes TIGRE works like a space-based robot, except that the conditions that TIGRE encounters are –at least sometimes– more adverse than that of a facility based in space. Thus, with TIGRE, the participating university institutes, which are all quite small compared to research institutes positioned outside the university system, have the opportunity to obtain access to very unique data sets.

For the TIGRE system the practical magnitude limit is between 8–10 mag depending on the specific requirements of the observer, fainter objects (such as the SN 2014J) can of course also be observed, however, the investment of observing time becomes excessive large. In this context we mention the ongoing TESS (photometric monitoring for exoplanet search) and eROSITA (X-ray all-sky survey) missions as well as the upcoming PLATO mission (photometric monitoring for exoplanet search) which are and will be yielding thousands of "interesting" brighter sources which require optical follow-up. As a matter of fact, the spectroscopic follow-up is already now the bottleneck for TESS, and the same will apply to PLATO. Thus it appears that also in the coming years there will be a strong scientific demand for spectroscopic robotic telescopes like TIGRE.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

ACKNOWLEDGMENTS

We acknowledge the continued support by various partners who helped to realize and operate TIGRE. In the first place, this is the

University of Hamburg, which gave support in terms of funding, manpower and workshop resources. Furthermore, grants from the Deutsche Forschungsgemeinschaft (DFG) in various funding lines are gratefully acknowledged, as well as travel money from both the DFG, DAAD and Conacyt in several bilateral grants. The Liège contribution to the TIGRE is funded through an opportunity grant from the University of Liège. The Universities of Guanajuato and Liège and the Mexican state of Guanajuato shared the funding of the infrastructure required by TIGRE at the La Luz site, and there is continued support by the University of Guanajuato in terms of manpower and running costs of the facilities. YN and GR acknowledge support from the FNRS-FRS (Belgium). We thank Alexander Hempelmann for his invaluable dedication and contribution to this project.

REFERENCES

Andersen, M. F., Handberg, R., Weiss, E., Frandsen, S., Simón-Díaz, S., Grundahl, F., et al. (2019). Conducting the SONG: The Robotic Nature and Efficiency of a Fully Automated Telescope. *PASP* 131, 045003. doi:10.1088/1538-3873/aaff7c

Baliunas, S. L., Donahue, R. A., Soon, W. H., Horne, J. H., Frazer, J., Woodard-Eklund, L., et al. (1995). Chromospheric Variations in Main-Sequence Stars. *Astrophysical J.* 438, 269. doi:10.1086/175072

Casagrande, L., and VandenBerg, D. A. (2018). On the Use of Gaia Magnitudes and New Tables of Bolometric Corrections. *MNRAS* 479, L102–L107. doi:10.1093/mnrasl/sly104

Castro-Tirado, A. J., Jelínek, M., Gorosabel, J., Kubánek, P., Cunniffe, R., Guziy, S., et al. (2012). "Building the BOOTES World-wide Network of Robotic Telescopes," in *Astronomical Society of India Conference Series. Vol. 7 of Astronomical Society of India Conference Series*, 313–320.

Castro-Tirado, A. J. (2010). Robotic Autonomous Observatories: A Historical Perspective. *Adv. Astronomy* 2010, 570489. doi:10.1155/2010/570489

De Gennaro Aquino, I., Schröder, K.-P., Mittag, M., Wolter, U., Jack, D., Eenens, P., et al. (2015). High Spectral Resolution Monitoring of Nova V339 Delphini with TIGRE. *Astronomy Astrophysics* 581, A134. doi:10.1051/0004-6361/201525810

Eberhard, G., and Schwarzschild, K. (1913). On the Reversal of the Calcium Lines H and K in Stellar Spectra. *Astrophysical J.* 38, 292–295. doi:10.1086/142037

Fraunhofer, J. (1817). Bestimmung des Brechungs- und des Farbenzerstreungs-Vermögens verschiedener Glasarten, in Bezug auf die Vervollkommnung achromatischer Fernröhre. *Ann. Phys.* 56, 264–313. doi:10.1002/andp.18170560706

Gaia CollaborationBrown, A. G. A., Vallenari, A., Prusti, T., de Bruijne, J. H. J., Babusiaux, C., et al. (2018). Gaia Data Release 2. Summary of the Contents and Survey Properties. *Astronomy Astrophysics* 616, A1. doi:10.1051/0004-6361/201833051

Gutiérrez, C. M., Arnold, D., Copley, D., Copperwheat, C. M., Harvey, E., Jermak, H., et al. (2019). The New 4-m Robotic Telescope. *Astron. Nachr.* 340, 40–45. doi:10.1002/asna.201913556

Hempelmann, A., Mittag, M., Gonzalez-Perez, J. N., Schmitt, J. H. M. M., Schröder, K. P., and Rauw, G. (2016). Measuring Rotation Periods of Solar-like Stars Using TIGRE. *Astronomy Astrophysics* 586, A14. doi:10.1051/0004-6361/201526972

Jack, D., Baron, E., and Hauschildt, P. H. (2015a). Identification of the Feature that Causes the I-Band Secondary Maximum of a Type Ia Supernova. *MNRAS* 449, 3581–3586. doi:10.1093/mnras/stv474

Jack, D., Mittag, M., Schröder, K.-P., Schmitt, J. H. M. M., Hempelmann, A., González-Pérez, J. N., et al. (2015b). Time Series of High-Resolution Spectra of SN 2014J Observed with the TIGRE Telescope. *Mon. Not. R. Astron. Soc.* 451, 4104–4113. doi:10.1093/mnras/stv1238

Jack, D., Robles Pérez, J. d. J., De Gennaro Aquino, I., Schröder, K. P., Wolter, U., Eenens, P., et al. (2017). Study of the Variability of Nova V5668 Sgr, Based on High-resolution Spectroscopic Monitoring. *Astron. Nachr.* 338, 91–102. doi:10.1002/asna.201613217

Jack, D., and Schröder, K.-P. (2019). Interstellar Absorption towards the Novae V339 Del and V5668 Sgr. *Rev. Mex. Astron. astrofísica* 55, 141–149. doi:10.22201/ia.01851101p.2019.55.02.02

Jack, D., Schröder, K. P., Eenens, P., Wolter, U., González-Pérez, J. N., Schmitt, J. H. M. M., et al. (2020). Time Series of Optical Spectra of Nova V659 Sct. *Astron. Nachr.* 341, 781–790. doi:10.1002/asna.202013818

Mittag, M. (2010). *Chromospheric Activity - First Results of the Observations with the Hamburg Robotic Telescope*. Ph.D. thesis (Hamburg, Germany: University of Hamburg).

Mittag, M., Hempelmann, A., Fuhrmeister, B., Czesla, S., and Schmitt, J. H. M. M. (2018). Detection of Radial Velocity Variability of HD 16673 with TIGRE. *Astron. Nachr.* 339, 53–59. doi:10.1002/asna.201713367

Mittag, M., Hempelmann, A., González-Pérez, J. N., and Schmitt, J. H. M. M. (2010). The Data Reduction Pipeline of the Hamburg Robotic Telescope. *Adv. Astronomy* 2010, 1–6. doi:10.1155/2010/101502

Mittag, M., Hempelmann, A., Gonzalez-Perez, J. N., and Schmitt, J. H. M. M. (2008). The Temperature Dependence of the Pointing Model of the Hamburg Robotic Telescope. *Publ. Astron Soc. Pac* 120, 425–429. doi:10.1086/533478

Mittag, M., Hempelmann, A., Schmitt, J. H. M. M., Fuhrmeister, B., González-Pérez, J. N., and Schröder, K.-P. (2017). Stellar Rotation Periods Determined from Simultaneously Measured Ca II H&K and Ca II IRT Lines. *Astronomy Astrophysics* 607, A87. doi:10.1051/0004-6361/201630262

Mittag, M., Schröder, K.-P., Hempelmann, A., González-Pérez, J. N., and Schmitt, J. H. M. M. (2016). Chromospheric Activity and Evolutionary Age of the Sun and Four Solar Twins. *Astronomy Astrophysics* 591, A89. doi:10.1051/0004-6361/201527542

Nazé, Y., Motch, C., Rauw, G., Kumar, S., Robrade, J., Lopes de Oliveira, R., et al. (2020). Three Discoveries of γ Cas Analogues from Dedicated XMM-Newton Observations of Be Stars. *MNRAS* 493, 2511–2517. doi:10.1093/mnras/staa457

Nazé, Y., Rauw, G., Czesla, S., Smith, M. A., and Robrade, J. (2022). Velocity Monitoring of γ Cas Stars Reveals Their Binarity Status. *MNRAS* 510, 2286–2304. doi:10.1093/mnras/stab3378

Nazé, Y., Rauw, G., and Smith, M. (2019). Surprises in the Simultaneous X-Ray and Optical Monitoring of π Aquarii. *Astronomy Astrophysics* 632, A23. doi:10.1051/0004-6361/201936307

Piskunov, N. E., and Valenti, J. A. (2002). New Algorithms for Reducing Cross-Dispersed Echelle Spectra. *Astronomy Astrophysics* 385, 1095–1106. doi:10.1051/0004-6361:20020175

Pollacco, D., Skillen, I., Cameron, A., Christian, D., Irwin, J., Lister, T., et al. (2006). The WASP Project and SuperWASP Camera. *Astrophys. Space Sci.* 304, 253–255. doi:10.1007/s10509-006-9124-x

Rauw, G., Nazé, Y., Campos, F., Fló, J. G., and Sollecchia, U. (2021). Irregular Emission Cycles in the Oe Star HD 60 848. *New Astron.* 83, 101462. doi:10.1016/j.newast.2020.101462

Rauw, G., Nazé, Y., Smith, M. A., Miroshnichenko, A. S., Guarro Fló, J., Campos, F., et al. (2018). Intriguing X-Ray and Optical Variations of the *γ* Cassiopeiae Analog HD 45314. *Astronomy Astrophysics* 615, A44. doi:10.1051/0004-6361/201731782

Rosas Portilla, F., Schrøder, C. F., and Jack, D. (2022). On the physical nature of the Wilson-Bappu effect: revising the gravity and temperature dependence. *MNRAS* 513, 906–924. doi:10.1093/mnras/stac929

Schmitt, J. H. M. M., Schröder, K.-P., Rauw, G., Hempelmann, A., Mittag, M., González-Pérez, J. N., et al. (2014). TIGRE: A New Robotic Spectroscopy Telescope at Guanajuato, Mexico. *Astron. Nachr.* 335, 787–796. doi:10.1002/asna.201412116

Schröder, K. P., Mittag, M., Flor Torres, L. M., Jack, D., and Snellen, I. (2021). Fast Synthetic Spectral Fitting for Large Stellar Samples: a Critical Test with 25 Bright Stars of Known Rotation. *MNRAS* 501, 5042–5050. doi:10.1093/mnras/staa2261

Schroeder, D. J. (2000). *Astronomical Optics*. Amsterdam, Netherlands: Elsevier.

Smith, M. A., Lopes de Oliveira, R., and Motch, C. (2016). The X-Ray Emission of the γ Cassiopeiae Stars. *Adv. Space Res.* 58, 782–808. doi:10.1016/j.asr.2015.12.032

Stokes, G. H., Evans, J. B., Viggh, H. E. M., Shelly, F. C., and Pearce, E. C. (2000). Lincoln Near-Earth Asteroid Program (LINEAR). *Icarus* 148, 21–28. doi:10.1006/icar.2000.6493

Strassmeier, K. G., Granzer, T., Weber, M., Woche, M., Andersen, M. I., Bartus, J., et al. (2004). The STELLA Robotic Observatory. *Astron. Nachr.* 325, 527–532. doi:10.1002/asna.200410273

Sybilski, P. W., Pawłaszek, R., Kozłowski, S. K., Konacki, M., Ratajczak, M., and Hełminiak, K. G. (2014). "Software for Autonomous Astronomical Observatories: Challenges and Opportunities in the Age of Big Data," in *Software and Cyberinfrastructure for Astronomy III. Vol. 9152 of Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*. Editors G. Chiozzi and N. M. Radziwill, 91521C. doi:10.1117/12.2055836

Wilson, O. C. (1978). Chromospheric Variations in Main-Sequence Stars. *Astrophysical J.* 226, 379–396. doi:10.1086/156618

Zito, R. R. (1990). "Cleaning Large Optics with CO2 Snow," in *Advanced Technology Optical Telescopes IV. Vol. 1236 of Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*. Editor L. D. Barr, 952–972. doi:10.1117/12.19176

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers | Frontiers in Astronomy and Space Sciences

ORIGINAL RESEARCH
published: 07 July 2022
doi: 10.3389/fspas.2022.897100

Pipeline for the Antarctic Survey Telescope 3-3 in Yaoan, Yunnan

Tianrui Sun[1,2], Lei Hu[1], Songbo Zhang[1], Xiaoyan Li[3], Kelai Meng[1], Xuefeng Wu[1,2], Lifan Wang[4] and A. J. Castro-Tirado[5]*

[1]Purple Mountain Observatory, Chinese Academy of Sciences, Nanjing, China, [2]School of Astronomy and Space Sciences, University of Science and Technology of China, Hefei, China, [3]Nanjing Institute of Astronomical Optical Instruments, Chinese Academy of Sciences, Nanjing, China, [4]Department of Physics and Astronomy, George P. and Cynthia Woods Mitchell Institute for Fundamental Physics and Astronomy, Texas A&M University, College Statio/n, TX, United States, [5]Stellar Physics Department, Instituto de Astrofísica de Andalucía, Granada, Spain

AST3-3 is the third robotic facility of the Antarctic Survey Telescopes (AST3) for transient surveys to be deployed at Dome A, Antarctica. Due to the current pandemic, the telescope has been currently deployed at the Yaoan Observation Station in China, starting the commissioning observation and a transient survey. This article presented a fully automatic data processing system for AST3-3 observations. The transient detection pipeline uses state-of-the-art image subtraction techniques optimized for GPU devices. Image reduction and transient photometry are accelerated by concurrent task methods. Our Python-based system allows for transient detection from wide-field data in a real-time and accurate way. A ResNet-based rotational-invariant neural network was employed to classify the transient candidates. As a result, the system enables the auto-generation of transients and their light curves.

Keywords: data analysis—astrometry—instrumentation, image processing, photometric—(stars), transient detection, convolutional neural networks—CNN

OPEN ACCESS

Edited by:
László Szabados, Konkoly Observatory (MTA), Hungary

Reviewed by:
Jean Baptiste Marquette, UMR5804 Laboratoire d'Astrophysique de Bordeaux (LAB), France
Enrico Congiu, University of Chile, Chile

****Correspondence:***
Xuefeng Wu
xfwu@pmo.ac.cn

Specialty section:
This article was submitted to Astronomical Instrumentation, a section of the journal Frontiers in Astronomy and Space Sciences

Received: *15 March 2022*
Accepted: *10 May 2022*
Published: *07 July 2022*

Citation:
Sun T, Hu L, Zhang S, Li X, Meng K, Wu X, Wang L and Castro-Tirado AJ (2022) Pipeline for the Antarctic Survey Telescope 3-3 in Yaoan, Yunnan. Front. Astron. Space Sci. 9:897100. doi: 10.3389/fspas.2022.897100

1 INTRODUCTION

The Antarctic Survey Telescope (AST) 3-3 is the third telescope planned for time-domain surveys at Dome A, Antarctica. Before shipping to Dome A, it was placed in the Yaoan observation station of the Purple Mountain Observatory for transient searching in the next several years. Yuan et al. (2015) have described an overview schedule and designation for AST3 series telescopes. The AST3 series includes three large field-of-view (FoV) and high photometric precision 50/68 cm Schmidt telescopes (Li et al., 2019). The AST3-3 is designed for time-domain surveys in the *K*-band to search for transients in infrared at Dome A. Due to the underdevelopment of infrared instruments of AST3-3, we temporarily used a CMOS camera (QHY411 with Sony IMX411 sensor) with a g-band filter for this commissioning survey instead. This camera has an effective image area of 54 mm × 40 mm and a pixel array of 14304 × 10748 with exposure times ranging from 20 μs to 1 h. With the CMOS camera, the FoV is 1.65° × 1.23°, the pixel scale is 0.41 arcsec, and the typical magnitude limit is 20 ~ 20.5 in the *g*-band for 60 s exposure images.

In the Yaoan observation station, we used the fully automatic AST3-3 telescope for a time-domain sky survey and follow-up observation. We have constructed an observation scheme for the follow-up observation of transients, according to the notices from Gamma-ray Coordinates Network (Barthelmy, 2008). The summary of our observation system and hardware, the survey and target-of-opportunity strategy, and the early science results will be presented in a forthcoming publication (Sun et al. in prep.).

This article presented a detailed overview of the AST3-3 data pipeline system. We have designed an automatic pipeline system containing data reductions, transient detection, and a convolutional neural network (CNN) framework for transient classifications. The data reduction pipeline includes instrumental correction, astrometry, photometry calibration, and image data qualification estimations. The transient detection pipeline consists of the alignments of images, image subtractions, and source detection on subtracted images.

The image subtraction algorithm automatically matches the point spread function (PSF) and photometric scaling between the reference and science images. In particular, one prevalent approach is the algorithm initially proposed by Alard and Lupton (1998) and further developed by a series of works (Alard, 2000; Bramich, 2008; Becker et al., 2012; Bramich et al., 2013; Hu et al., 2021). This technique has been extensively used in the transient detection pipelines (Zhang et al., 2015; Andreoni et al., 2017; Masci et al., 2019; Zhang et al., 2020; Brennan and Fraser, 2022). In the last decade, it has played an important role in many successful time-domain survey programs, including intermediate Palomar Transient Factory (Cao et al., 2016), Dark Energy Survey (Morganson et al., 2018), and Panoramic Survey Telescope And Rapid Response System-1 (PS1 hereafter, Price and Magnier, 2019).

Time-domain surveys are demanded to find transients as fast as possible, but many bogus candidates are detected from the image subtraction results. The human workload can be greatly reduced by the classifier methods such as machine learning and deep learning (Gómez et al., 2020; Yin et al., 2021). Random forest and some other machine learning methods previously attempted to solve the classification problem (Goldstein et al., 2015). We have applied a CNN framework to the estimation of the image qualification and candidate selections after transient extraction in this work. The CNN found optimal results with backpropagation (Rumelhart et al., 1986), and its accuracy has approached the human level in some classification and identifying tasks (Lecun et al., 2015). Similar to some sophisticated neural networks, these CNN models can naturally integrate features from different levels and classify them in end-to-end multilayer neural networks. Dieleman et al. (2015) introduced the first rotation invariant CNN to classify galaxies by considering the inclination of the object in the classification. The approach using the rotation-invariant CNN soon makes its way into the transient survey programs, e.g., The High cadence Transient Survey (HiTS; Förster et al., 2016) aimed at searching transients with short timescales. In the transient detection procedure of the HiTS, Cabrera-Vives et al. (2017) used the rotation-invariant CNN to classify the real transient candidates and the fake candidates from the image subtractions. Jia et al. (2019) modified the rotation-invariant CNN by adding the long short-term memory network (Hochreiter and Schmidhuber, 1997) to enhance the performance in the satellite trail identifications. The alert classification system for the Zwicky Transient Facility survey also uses the updated rotation-invariant CNN (Carrasco-Davis et al., 2021). The previous CNN structures for classification used superficial layers for feature extraction, and residual learning frameworks have been introduced in He et al. (2015) to avoid the loss of too much information and the difficulty of deeper CNN.

In **Section 2**, we described the data reduction pipeline and the qualification evaluation methods for image data. The transient detection pipeline for AST3-3 is presented in **Section 3**. **Section 4** describes the CNN structure and training for classifying transient candidates and their performance. We showed the conclusions in **Section 5**.

2 DATA REDUCTION PIPELINE

The data reduction pipeline aims at reducing the instrumental effects on the observational image to create the scientific image and apply the basic calibration information. This pipeline stage contains a group of subroutines for instrumental correction, astrometry, and photometry calibration. We also added a group of methods for evaluating the quality of images. The entire flow for the single-frame image processing is shown in **Figure 1**.

2.1 Instrumental Correction

The instrumental calibrations include overscan area removal and bias correction, flat-field correction, bad pixels, and cosmic-ray detection. The CMOS sensor of the camera on AST3-3 provides a 14304 × 10748-pixel image array including a narrow overscan area of 50 lines. which contains the data for bias corrections. We used the median of the overscan area for each line as a bias field value similar to the CCDPROC method (Craig et al., 2015) and produced an image array of 14206 × 10654 pixels. The bias correction effectively removes the background boost from the offset value in the camera settings.

The AST3-3 telescope observes about 80 flat field images at a half-full maximum of the pixel capacity of near 30000 ADUs every observable twilight. The flat field image was constructed with the sigma-clipped median method like IRAF (National Optical Astronomy Observatories, 1999) and applied to the observation images. The image also contains the cosmic ray defects on the detector, and we added the **L.A.Cosmic** package for the cosmic ray identification with the Laplacian edge detection method (van Dokkum, 2001). A growing number of satellites plot trails on images that shall be added to the mask image. The pipeline uses the **MaxiMask and MaxiTrack** methods for star trail detections (Paillassa et al., 2020).

2.2 Astrometry Calibration

The pipeline for astrometry calibration solves the solution for the world coordinate system (WCS, Calabretta and Greisen, 2000) and fits the WCS distortion parameters. The astrometry solution and the estimation of full-width half-maximum (FWHM) for images require source detection, and we optimized the sequence. The FWHM is a critical parameter in describing how the turbulence of the atmosphere and the properties of the optical system affect the observations of point sources. The pipeline uses the **SExtractor** (Bertin and Arnouts, 1996) at first to detect sources on the image and measure basic parameters of stars

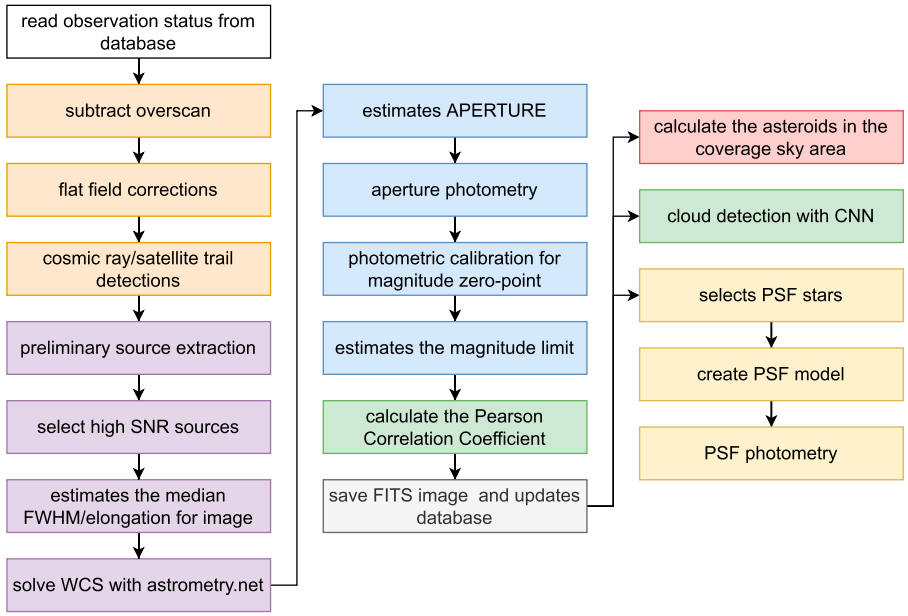


FIGURE 1 | Flow of the single-frame image processing for AST3-3.

with the automatic aperture photometry method. The preliminary photometry catalog results contain some bad sources, and we cleaned out the bad detection with the following restrictions:

- The neighboring, blended, saturated, or corrupted stars were excluded by removing the sources with FLAGS larger than zero.
- The sources with the automatic aperture flux parameter FLUX_AUTO that is not zero and the ratio of FLUX_AUTO and FLUXERR_AUTO larger than 20 were excluded.
- The isophotal and automatic aperture result parameter MAG_BEST lower than 99 was selected to exclude the bad magnitude fitting.
- The outlier of FWHM_IMAGE and elongations of sources were excluded by the sigma clip method with the 3σ threshold.
- The catalog was sorted by the star-galaxy classification CLASS_STAR, and the last 20% of the catalog was removed.

The remaining catalog contained the most of detected point sources. We used the median of the FWHM_IMAGE for sources in the remaining catalog as the FWHM of the image. The pipeline calls the solve-field program in Astrometry.net (Lang et al., 2010) to fit the WCS for their flexible local index files built from the Gaia Data Release 2 (Gaia Collaboration et al., 2018). We provided the X, Y, MAG_BEST list for point sources, the pointing RA and DEC, the search radius, and the estimated pixel scale for the solve-field program to speed up the searching and fitting procedure.

AST3-3 has a large FoV of 1.65° × 1.23°, which requires distortion corrections in the WCS. The pipeline also calculates the WCS with a fourth-order simple imaging polynomial (SIP, Shupe, et al., 2005) distortion for the accuracy of the coordinates. The mean solving time for ordinary observation is approximately 1.7–2 s for a catalog generated by the SExtractor.

2.3 Photometry

The pipeline extracts sources and estimates their flux and magnitudes on the image. We have applied the aperture photometry and PSF photometry methods in the pipeline. The aperture for aperture photometry is determined with **Equation 1**. A_IMAGE is the semimajor axis value in the catalog what matches the selection criteria in **Section 2.2** and the default value $C = 6.0$ as derived in Sokolovsky and Lebedev (2018),

$$\text{Aper} = C \times \text{median}(\text{A\_IMAGE}). \tag{1}$$

Aperture photometry is performed by the SExtractor with the DETECT_THRESH of 2σ and the estimated aperture. The pipeline cleans the catalog from aperture photometry with the same distilling criteria described in **Section 2.2**. The estimation of the magnitude zero-point calculates the difference between aperture photometry and a reference catalog. We have selected PS1 (Chambers and Pan-STARRS Team, 2017) as the reference catalog for magnitude calibration for its sky coverage and much better magnitude limits. We used a χ^2 minimization method introduced in the PHOTOMETRYPIPELINE (Mommert, 2017) to generate the magnitude zero-point as **Equation 2**:

$$\chi^2 = \sum_i^N \frac{\left(m_{zp} - \chi_i\right)^2}{\sigma_{\chi,i}^2}. \tag{2}$$

In **Equation 2**, the i-th parameter χ_i is the difference between the aperture magnitude and the g magnitude in PS1 for the i-th source in the catalog for all N-matched sources. The magnitude zero-point m_{zp} is determined by minimizing χ^2 with an iterative process to reject outlier samples.

The magnitude zero-point calibration has some residual due to the spatial variation in the atmospheric extinction across the large FOV. We used a quadratic form to fit the offsets between the reference catalog and the zero-point calibrated aperture magnitudes. The 2D polynomial formula to fit the residuals is shown in **Equation 3**, introduced in Irwin et al. (2007):

$$\Delta m(x, y) = c_0 + c_1 x + c_2 y + c_3 xy + c_4 x^2 + c_5 y^2. \quad (3)$$

Δm is the zero-point offset for a star, and c_i are the polynomial coefficients for the equation. Given that the background noise dominates the flux uncertainty of the faintest sources, the pipeline estimates the limiting magnitude based on the sky background with the method of Kashyap et al. (2010) with the root mean square of background by the Python library SEP (Barbary, 2016).

PSF photometry requires a group of well-selected stars for profile fitting. We selected the sources in the aperture photometry catalog with the same criteria in **Section 2.2** but limited the restriction of FWHM_IMAGE and ELONGATION to 1σ to keep the best ones.

The pipeline feeds the selected catalog to PSFEx (Bertin, 2013) to generate a position-dependent variation PSF model. The SExtractor accomplishes the PSF photometry with the result of PSFEx. It takes a relatively long time for PSF fitting and model fitting in PSF photometry. This part works in the background after the single-frame image process.

2.4 Data Quality Inspection

This subsection introduces the image quality inspection method, which estimates how the cloud affects the observation image. The magnitude limit is an excellent parameter for the qualification of an image. For a large FoV telescope, the image extinction and airmass may vary across the FoV of the instrument, especially at lower altitudes. The magnitude limit may also vary across the FoV of the instrument, producing spatial variations of the limiting magnitudes. In ideal observation conditions, we suppose that the extinction is consistent everywhere in the image, which causes all-stars to have the same magnitude difference between the machine magnitude and reference catalog. Small clouds also influence the photometry results due to their discontinuous extinction to the nearby image parts. In the affected parts of the image, the flux of stars and galaxies would be reduced by clouds more than in other regions. In addition to the effects of clouds and extinction, incorrect WCS fits resulting in false star catalog matches can seriously affect the magnitude corrections.

Based on the corrected aperture magnitudes and reference magnitudes being equal within the margin of error, we calculated the correlation between them. The measurement uses the Pearson correlation coefficient (also called Pearson's r, PCC hereafter, Pearson and Galton, 1895). The pipeline also selects stars that match the criteria, as described in **Section 2.2**, and calculates the PCC as **Equation 4**:

$$r = \frac{\sum\left((m_a - \overline{m_a})\left(m_g - \overline{m_g}\right)\right)}{\sqrt{\sum (m_a - \overline{m_a})^2 \sum \left(m_g - \overline{m_g}\right)^2}}. \quad (4)$$

Equation 4 gives the expression of PCC, where m_a means the aperture photometry magnitude, and m_g is the g-band magnitude in PS1. For a completely ideal situation, the value of PCC becomes 1. For a typical AST3-3 image, the PCC value should be larger than 0.98 to avoid the influence of the clouds. When using PCC to estimate the image quality, we noticed that the uniform thin clouds do not influence the PCC in some images in a very significant way. **Figure 2** shows a typical image of the good PCC with an obvious cloud on the image at an altitude of 53°.

The discontinuous extinction would cause errors in flux calibrations in the image subtraction process for transient detection. As we can see the cloud from the thumbnails in the left panel of **Figure 2**, we could screen out the images with cloud effects manually. The pipeline creates thumbnails with the Zscale (National Optical Astronomy Observatories, 1999) adjustment and normalization method in Astropy (Greenfield et al., 2013) to downscale it to an image size of 256 × 256 pixels.

Our manually checked results showed that the cloud effects were still visible in the thumbnails. Thus, the cloud image classification for images is a simple classification problem that could be solved with the CNN method.

Consequently, we converted the cloud image classification into a simple image classification problem. We attempted to check for clouds in the images using the 18-layer residual neural nets (ResNet-18, He et al., 2015), as shown in **Table 1**. ResNet-18 is the simplest structure of residual neural networks, allowing a deeper network with faster convergence and easier optimization. We have chosen the original ResNet-18 structure as there are no vast data. The AST3-3 data are monochrome, which means we have only one channel of data to the input layer of the CNN. We have modified the input layer to the input channel of 1, output channel of 64, kernel size of 7 × 7, and stride of 1 to match the thumbnail data and the second layer. Three fully connected layers construct the classifier for the features extracted by ResNet-18. In the classifier part, we selected the default rectified linear units (ReLU, Nair, and Hinton, 2010) function as the formula of max (0, x) as the activation function.

The CNN training data set contains 1000 clear images and 1000 cloudy images as a balanced dataset. To avoid overfitting in the CNN training, we used dropout to 0.5 in the classifier layers. We have selected 20% of the balanced data set as the test set for validation during the training of ResNet-18. The training uses the optimizer AdamW (Loshchilov and Hutter, 2017) and a scheduler for reducing the learning rate after each training epoch. The CNN-trained result shows accuracy in the test group of 98.35% and a recall rate of 98.28%. CNN's classification of the thumbnail results is helpful as an indicator for significant cloud effects from our results. In practice, if CNN's classification of the thumbnail is cloud-affected, the image would be marked in the database and the website. If there is an obvious problem with the result of transient detection, like a massive number of detected candidates with a cloud-affected report, the image would be dropped automatically.

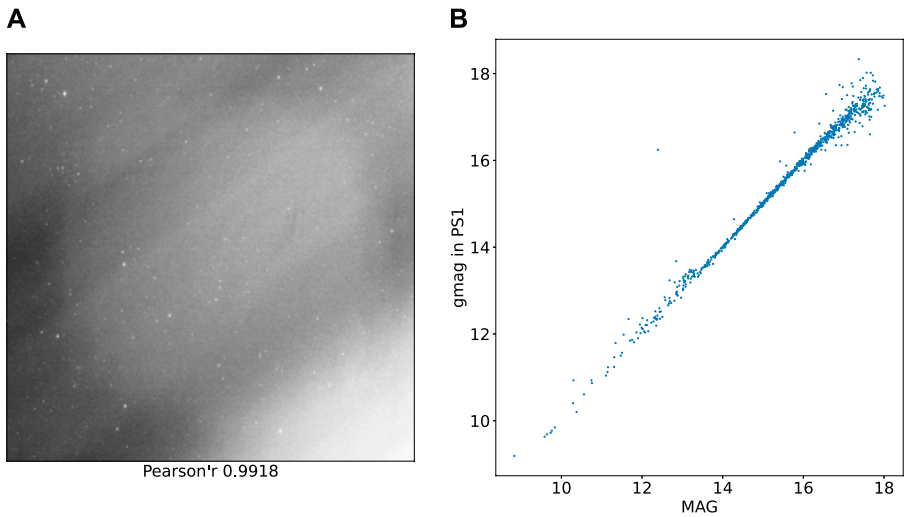


FIGURE 2 | (A) shows the thumbnails of an AST3-3 image with obvious cloud effects. **(B)** is a scatter plot of the calibrated aperture photometry magnitude and g-mag in PS1.

TABLE 1 | Residual neural network structure with the residual block.

Layer/Block	Input channel	Output channel	Kernel size	Stride
Conv2d	1	64	7 × 7	1
BatchNorm2d	64	64	-	-
Residual block	64	64	3 × 3	1
Residual block	64	64	3 × 3	1
Residual block	64	128	1 × 1	1
Residual block	128	128	3 × 3	1
Residual block	128	256	1 × 1	2
Residual block	256	256	3 × 3	1
Residual block	256	512	1 × 1	2
Residual block	512	512	3 × 3	1
GlobalAvgPool2d	512	512	4 × 4	3

The data reduction pipeline schedules the single-image process independently to each image as a standalone thread concurrently. The approximate average calculation time for the sparse starfield is 17 s per image after the observation. The pipeline also marks the necessary information into the FITS header and updates the database for the following step programs.

2.5 Data Reduction Pipeline Performance

We have estimated the accuracy of the WCS for images by comparing the star position difference between the WCS and the Gaia-DR2 catalog. For each image, we matched the catalog between the aperture photometry result and the Gaia-DR2 catalog and calculated the difference between right ascension and declination for the matched stars. The astrometry standard deviation in the data reduction pipeline is between 100 and 200 mas, as shown in **Figure 3**.

The local PSF photometry results have some differences from the Pan-STARRS DR1 catalog due to some noise that should be well-fitted. We selected an example of sky area "0735 + 1300" at an altitude of 75° above the horizon to show the result of PSF photometry fitting and the magnitude zero-point calibration. **Figure 4** shows the difference map with density color and the standard deviation trends from the detected bright to dark sources.

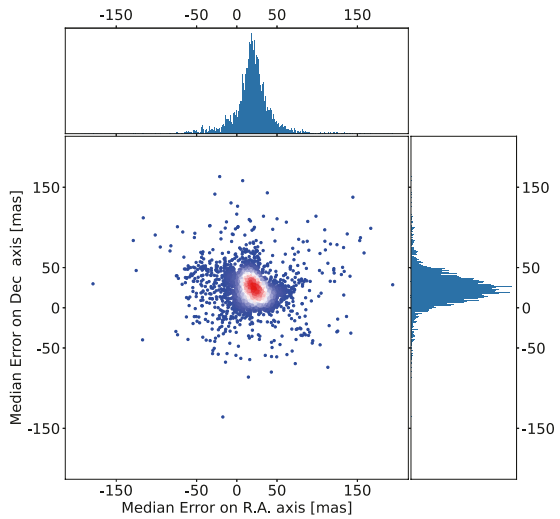


FIGURE 3 | Main scatter-plot shows the median deviation distribution of the astrometric error along each axis with respect to the Gaia-DR2 catalog for AST3-3 for stars in three thousand images. The upper and right panel shows the histograms of the distributions for both axes.

Figure 5 shows the PCC correlation distribution described in **Section 2** since the first light in the Yaoan observation station on 27 March 2021. In all the 60-s exposure images, 98% of them had a PCC value greater than 0.95, and 84% of them had a better PCC value of 0.99.

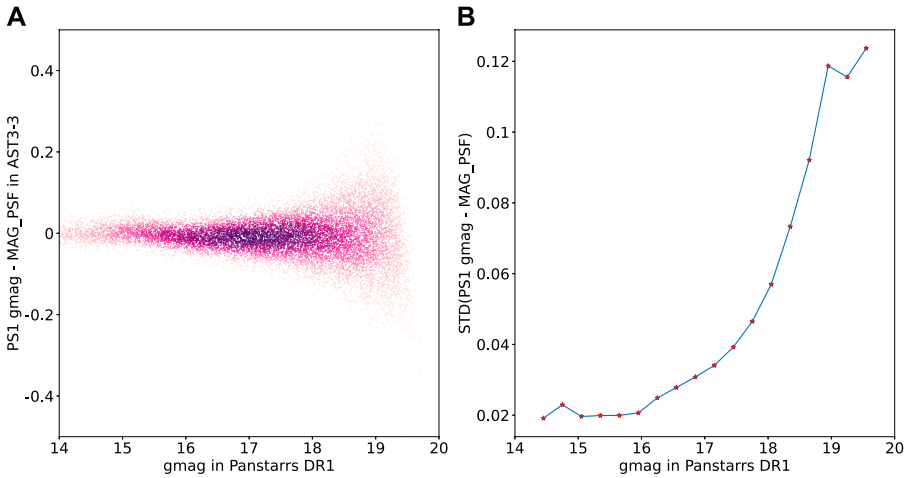


FIGURE 4 | (A): Difference between the AST3-3 *g*-band photometry and PS1 *g*-mag values. **(B)**: Standard deviation of the binned and sigma clipped difference between the AST3-3 mags and PS1 *g*-mags.

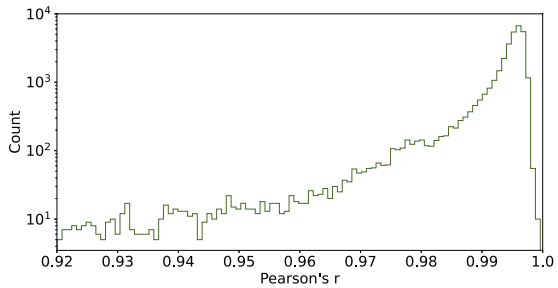


FIGURE 5 | PCC value distribution for 33539 images since AST3-3 starts observation in 60-s exposure modes.

3 TRANSIENT DETECTION PIPELINE

The transient detection pipeline compares the newly observed science image with the previously observed template image data to find the transients appearing on images. This section builds a fully automatic pipeline for transient searching with the alignment method, image subtractions, and source detection on the different images. We adopted the GPU version of the Saccadic Fast Fourier Transform (SFFT) algorithm (SFFT hereafter, Hu et al., 2021) to perform the image subtraction. The SFFT method is a novel method that presents the least-squares question of image subtraction in the Fourier domain instead of real space. SFFT uses a state-of-the-art δ function basis for kernel decomposition, which enables sheer kernel flexibility and minimal user-adjustable parameters. Given that SFFT can solve the question of image subtraction with fast Fourier transforms, SFFT brings a remarkable computational speed-up of an order of magnitude by leveraging CUDA-enabled GPU acceleration. In real observational data, some sources can be hardly modeled by the image subtraction algorithm, and we should exclude them to avoid the solution of image subtraction being strongly misled. In our work, we used the built-in function in SFFT to pre-select an optimal set of sub-areas for a proper fitting.

TABLE 2 | Processing time for different methods.

Pipeline	Method	Bin-type	Image size	Time (s)	Stamp size
QIP	SFFT	Bin3×3	4736 × 3552	2	31 × 31
DIP	SFFT	Entire	14206 × 10654	32	91 × 91

For the automatic follow-up observation of some transients, we need a rapid image subtraction process to search the possible transient candidates. AST3-3 science images are 14206 × 10654 pixels in size, with a very sparse star field for high galactic latitude observations. The large data array takes a long time for kernel fitting and convolutions in image subtraction. We built two pipeline systems for transient detection. One focused on the quick analysis of the newly acquired image, especially for target-of-opportunity observations (quick image processing, QIP), and another one aimed at obtaining more reliable detections (deep image processing, DIP). The comparison of the two systems is shown in **Table 2**. QIP uses the 3 × 3 binned image of size 3552 × 4736 pixels to boost the image alignment and subtractions. The 3 × 3 binned pixel scale increased to 1.23 from 0.41 arcsec per pixel, which increased the sky background noise. The flowchart of the pipeline system is shown in **Figure 6**.

It is essential to prepare optimal template images to enable transient detections. We chose the earliest acceptable image taken for each sky area with restrictions on image qualities. The magnitude limit for templates should be better than 18.5 magnitudes. The PCC value was greater than 0.98, and there was no apparent cloud structure. The template image is copied directly after the data reduction pipeline result and tagged as the reference in the database. We created the template images for the QIP from the template image by the bin 3 × 3 method. Since the

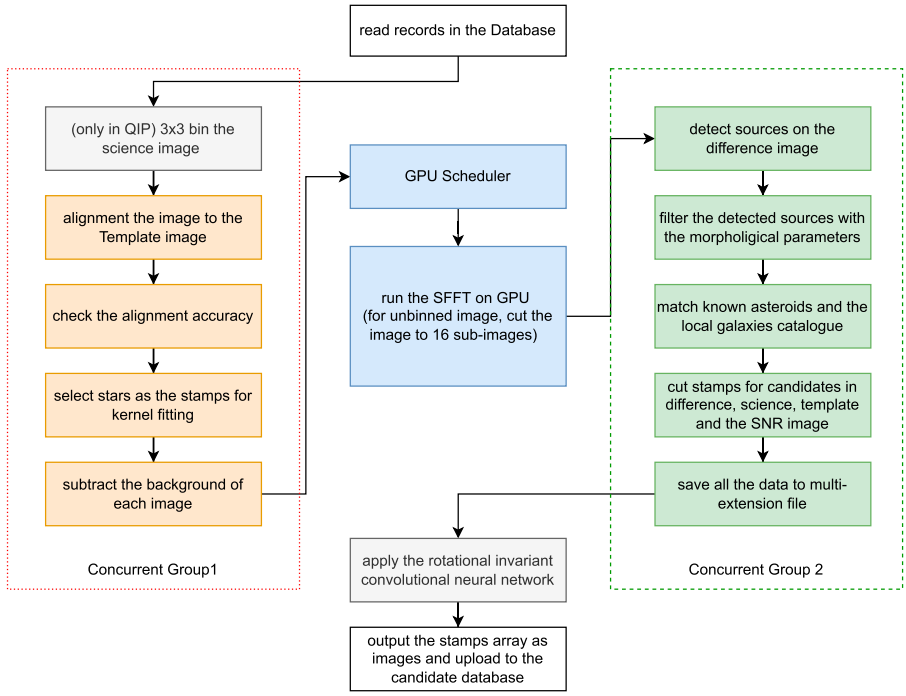


FIGURE 6 | This flowchart shows the transient detection pipeline for AST3-3. The pipeline is divided into four components. The left box shows the first concurrent group that contains the preparations for image subtractions. The middle part is the SFFT subtraction procedure, which calls the SFFT functions with a GPU resource scheduler. The right box shows the second concurrent group for candidate detection on the difference images. The bottom part shows the candidate classification and the data exchange of the database.

template image for the QIP is a new image, we run the photometry and astrometry on the image with the same method in **Section 2**. The header of the WCS part of the template image is saved as a separate file to facilitate the SWarp (Bertin, 2010) for the alignment of the science image to the template.

The strategies of each step for the QIP and DIP programs are the same: the resampling and interpolation of images for alignment, the kernel stamp selection, image subtractions, and the source detections on the difference images. As a survey telescope, AST3-3 has a fixed grid for observation. The shift and rotation between two images are tiny for each sky area but exist. Image alignment corrects the positional difference with a transformation matrix between the WCS of the template and science images. We used the SWarp for the resampling procedure with the LANCZOS3 algorithms. For the QIP part, the science image was binned to 3552 × 4736 pixels by scikit-image before resampling.

We performed image subtraction using SFFT for the binned images in QIP and the full-frame images in the DIP branch, respectively. It should be noted that the subtraction tasks are scheduled *via* the database, and the two branches are triggered concurrently. The resulting difference images are used to detect transient candidates.

The calculations involved in SFFT are carried out using the multiple NVIDIA A100 GPUs equipped on our computing platform. For the QIP case, we performed image subtraction straightforwardly for the 3 × 3 binned images. For the DIP case, we split the large full-frame image into a grid of sub-images with the size of 3072 × 4096 pixels to avoid memory overflow.

The pipeline uses the SExtractor to perform the target search on the different images. We run the SExtractor with aperture photometry and a threshold (DETECT_THRESH) of 2σ for searching the star-like objects on the different images. Some detected negative sources have obvious problems finding stars on the subtracted images. We cleared the bad sources with the following criteria:

- The sources with FWHM_IMAGE lower than one pixel or more extensive than two times the image FWHM were excluded.
- The sources with ELONGATION lower than 0.5 or larger than 6 were excluded.
- The sources with ISOAREA_IMAGE less than four were excluded.
- The sources with FLAGS less than four were excluded.
- The sources near the image edge in 16 pixels were excluded.

After cleaning up, the catalog of different images became the candidate catalog. Extragalactic transients have their host galaxies nearby, and most galaxies are already known. The pipeline cross-matches the candidate catalog with the GLADE catalog (Dalia et al., 2021) with coordinates to obtain some near galaxy transients.

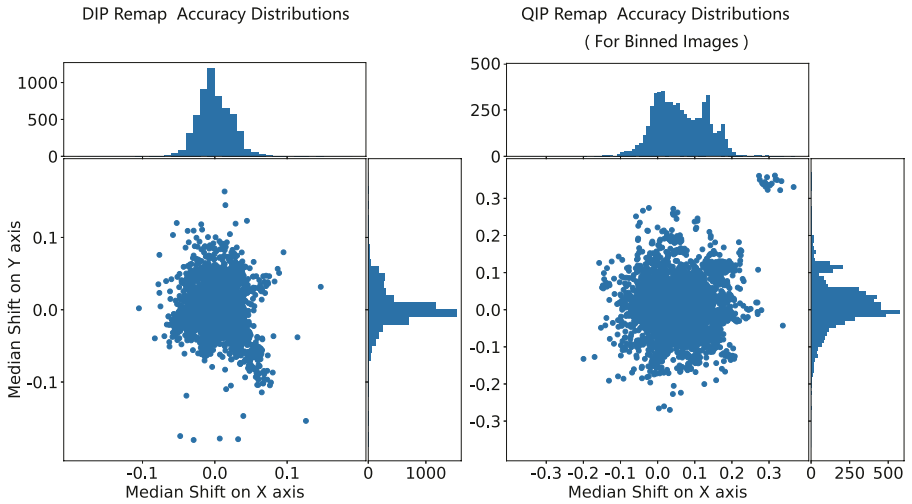


FIGURE 7 | Two panels show the standard deviation of the x and y position differences in matched stars between the reference and remapped images. The right panel shows the unbinned images' remap accuracy, and the left panel shows the remap accuracy for QIP.

It is easy to detect many asteroids at the Yaoan station due to its latitude. The pipeline calculates the location of asteroids by PyEphem (Rhodes, 2011) with the Minor Planet Center Orbit (MPCORB) Database for each image. It also cross-matches all the candidate catalogs with the local asteroid catalog to exclude the known asteroids.

The transient detection program pipeline ends when the provisional source detection is complete. As the transient detection program could reproduce all the data with the science images, they are retained for only 30–60 days to save storage space. We have built a multi-HDU (Header Data Unit) FITS format regulation to facilitate program error checking to store the data. The primary HDU stores only header information, describing a summary of this image subtraction process results. The other HDUs store the subtracted image, the aligned image, the template image, and the star list of the temporal source candidates through a compressed FITS image (Pence et al., 2009).

The image alignment is handled by SWarp based on the WCS information of the template and science images. The position difference should be near zero for stars on both the template and alignment images to avoid the error occurrence during image subtraction. It is a fast method to check the accuracy of image alignment by examining the position difference between the catalogs from the alignment image and the reference image. Thence, we match the alignment catalog by SExtractor and the reference catalog with the grmatch program in FITSH (Pál, 2012). The position difference is calculated from the position of matched stars in the X and Y planes. **Figure 7** shows the median distributions of matched star-position deviations of the DIP and QIP pipelines. The image alignment accuracy of our pipeline is typically less than 0.05 pixels, with a standard deviation of fewer than 0.2 pixels. The QIP has only slightly degraded accuracy due to the pixel scale binned to 1.23 arcsec.

The observation of asteroid 1875 is shown in **Figure 8** by the time-domain survey and selected by the transient detection pipeline. It is a well-detected example of the pipeline described in this section. The target magnitude is 18.24 in an image with a magnitude limit of 19.9. The target can be seen clearly in the difference image and the pattern of nearby bright stars.

The performance of QIP and DIP procedures in images is only relevant to their pixel binning properties. As a result, the background and background's standard deviation increases, decreasing the limiting magnitude of the images in QIP. Since the QIP is only designed for the image with high priority observations, the resources used for QIP are restricted in both GPU time and the threshold of source extraction on the difference image. The QIP finished after the image was taken about 60 s, and the DIP finished after the image observation of about at least 5–10 min due to the GPU time queuing.

4 CANDIDATE CLASSIFICATIONS

The AST3-3 telescope is monochromatic in the *g*-band. It is difficult to distinguish among different types of transient sources with their morphological information in only several images. We divided the detected candidates into two categories: positive and negative candidates. The positive candidates are new point sources or variable sources on the science image. The positive candidates could be any astrophysical origin targets, while the negative candidates mainly originate from residuals and errors of the image subtraction pipelines. In this section, we chose to use the CNN-based approach to filter out the negative candidates from the image subtraction procedures.

4.1 Rotation-Invariant Neural Network

The original rotation-invariant CNN for classifying natural and artificial sources from transient detection pipelines is introduced in Cabrera-Vives et al. (2017) for HiTS as the CNN model named Deep-HiTS. The Deep-HiTS uses the data array by combining

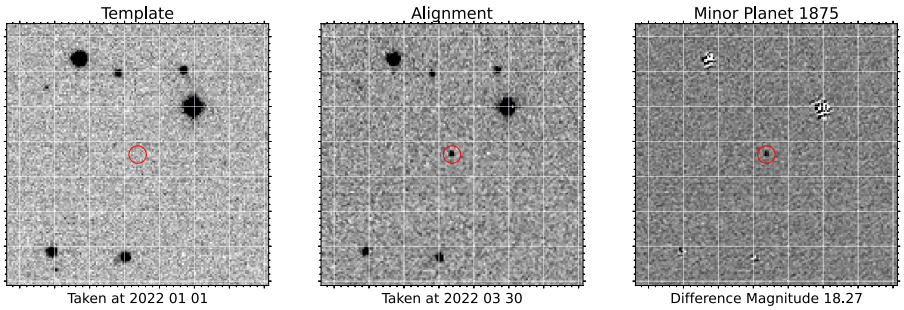


FIGURE 8 | Example of the QIP: the detection of minor planet # 1875 at magnitude 18.27. The left panel shows the template image taken on 1 January 2022, and the middle panel shows the remapped image taken on 30 March 2022. The right panel shows the difference images produced by the QIP pipeline.

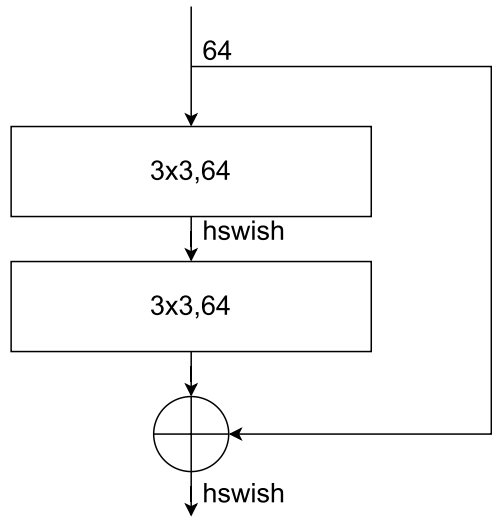


FIGURE 9 | Residual block with the H-swish function.

difference images, template images, science images, and signal-to-noise ratio images as the stamps to classify the candidates generated from the detection after image subtraction. The original Deep-HiTS network rotates the combination of images to 0, 90, 180, and 270° to feed four independent CNNs for feature extraction. The CNN parts of Deep-HiTS only use a simple seven-layer structure that more complex neural networks can replace.

It is expected that there should be nonlinear activation functions between the layers of the CNN to enhance the performance of multilayer structures. The rectified linear units (ReLU, Nair, and Hinton, 2010) and their modifications are activation functions widely used in recent years. The ReLU function discards all negative values with max (0, x) for the sparsity of the network. The original Deep-HiTS network uses the Leaky ReLU function, which improves the negative part by multiplication with 0.01 instead of zero as the formula max (0.01 × (x, x)). In the development of Mobile-Net V3 (Howard et al., 2019), the h-swish function in **Equation 5** is used as the activation function to improve the accuracy of neural networks as a drop-in replacement for ReLU,

$$\begin{aligned} \mathrm{ReLU6}(x) &= \min(\max(0, x), 6), \\ \mathrm{H-Swish}[x] &= x\frac{\mathrm{ReLU6}(x+3)}{6}. \end{aligned} \tag{5}$$

We have modified the residual blocks in **Figure 9** in the residual neural networks by changing the activation function to H-Swish. By organizing the residual blocks to the structure of ResNet-18, we can replace the CNN parts of Deep-HiTS with ResNet-18, as shown in **Figure 10**.

We built the three-dimensional array by combining each candidate's difference, template, science, and SNR stamps to feed the CNN models. For candidates from DIP, the array size is 91 × 91 × 4. For the candidates from QIP, the array size is 31 × 31 × 4 due to its 3 × 3 binned image properties. Before the CNN calculation, the pipeline rotates the stamp arrays 90, 180, and 270° to feed the four branches of our modified Deep-HiTS network. The main structure of our rotational invariant neural network is given in **Figure 10**.

Our convolutional parts used the modified ResNet-18 for feature extraction as the structure **Table 1**. The input channel is 4 to match the stamp arrays, and the output channel is 64 to feed the residual blocks. For the stamps with the 31 × 31 × 4 structure, the kernel size of the input layer is 3 × 3, and the stride step is 1 in the QIP. For the stamps with 91 × 91 × 4 from the DIP, the first layer used a 4 × 4 kernel size and stride steps of 3 to match the residual block inputs.

For each rotation of the stamp array, the modified ResNet-18 could create a vector of 2048 values as the feature extractions. We concatenated the output vectors of all rotations to a vector of 4 × 2048. The fully connected layers are constructed by three linear layers and two H-Swish activation functions. These feature values were classified into two fully-connected layers.

Due to the limitations of single-band observations, we only classified the candidates into positive and negative categories rather than performing a multicategory analysis. The classification of the results could be a binary problem that could use the simple cross-entropy loss function. The $p(x)$ in the function represents the true value $q(x)$ and represents the neural network classification value. The whole training problem of the neural network is, thus, to find the minimum cross-entropy under the training samples:

$$H(p, q) = -\sum_{x} p(x)\log q(x). \tag{6}$$

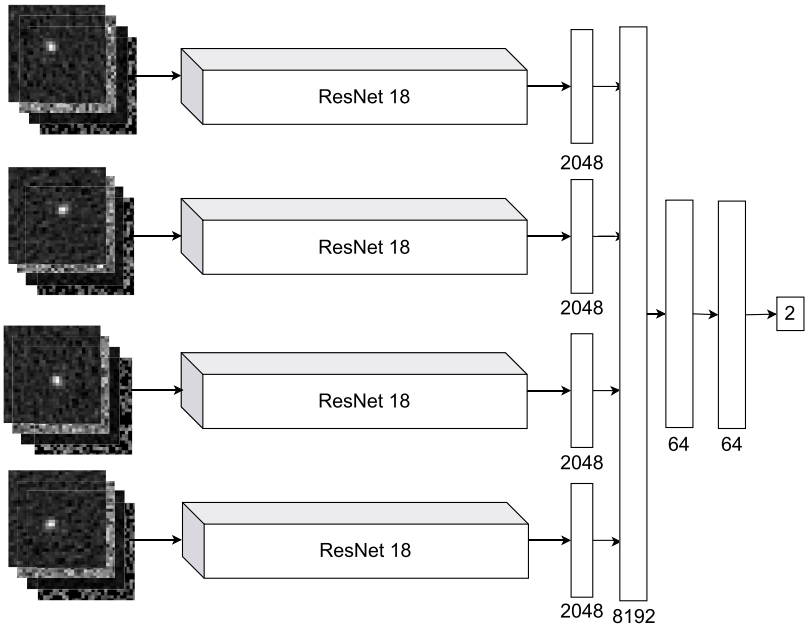


FIGURE 10 | Rotational-invariant residual neural network.

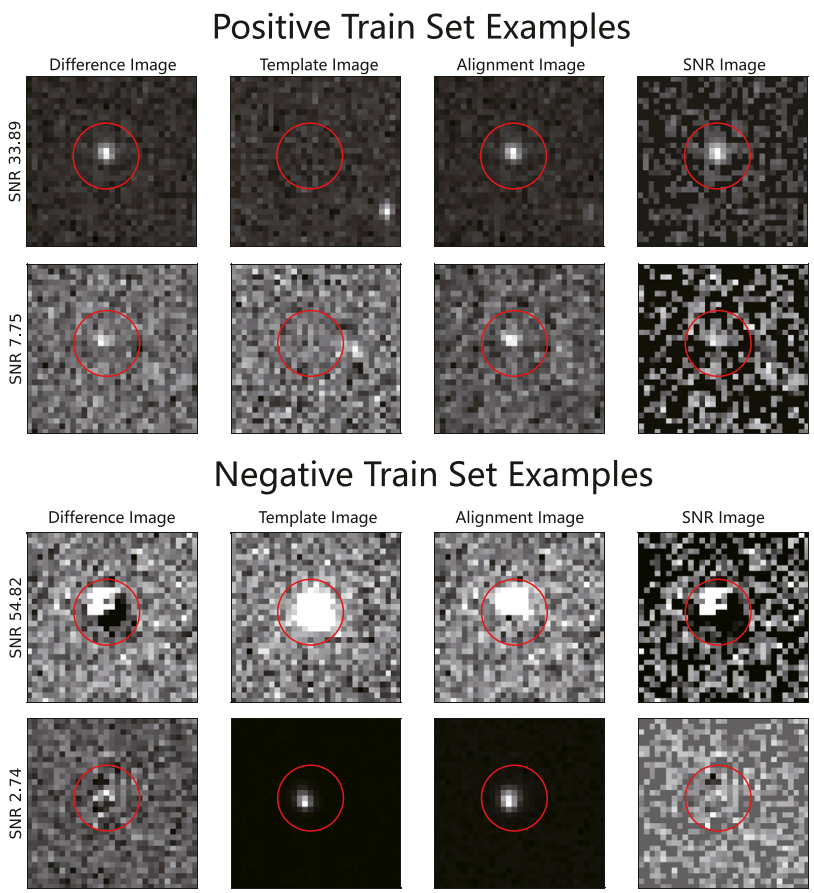


FIGURE 11 | Examples of source samples for the positive and negative candidates used for training the neural networks.

TABLE 3 | Precision of the CNN model for stamps from QIP and DIP.

Method	Stamp size	Accuracy (%)	Recall (%)	Precision (%)	F1 score (%)
Bin3×3	31 × 31	99.88	99.87	99.87	99.88
Unbinned	91 × 91	99.20	99.20	99.21	99.20

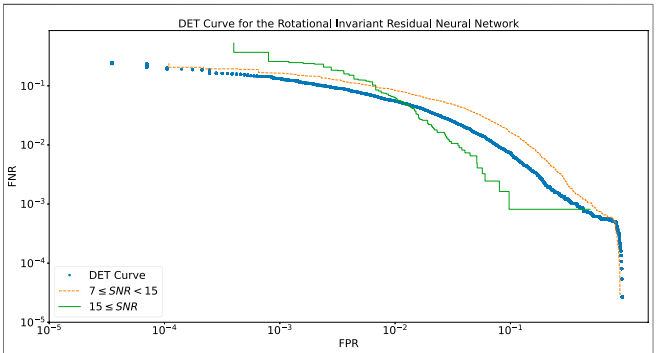


FIGURE 12 | Blue dots show the detection error trade-off curve for the stamps from 3 × 3 binned images. The green and orange curves show the DET curve for different signal-to-noise ratio groups. The x-axis is the false negative rate, and the *y*-axis is the false positive rate.

4.2 Training and Accuracy

The CNN training requires an extensive data set to avoid overfitting. The training data set should preferably be a balanced sample for CNN models. However, it is impossible to construct a balanced sample from the observational data. The negative candidates generated by transient detection are enormous, while the positive candidates are very rare in comparison. Thus, we used the simulated positive candidates to address this imbalance problem.

To create the simulated positive candidates, we selected hundreds of science images with the best magnitude limits, lower airmass of observation, and an excellent full-width half maximum of detected stars. For each image, the space-variation PSF is generated by PSFEx from the aperture catalog with the same method described in PSF photometry. The PSF model is constructed with a stamp (VIGNET) size of 31 × 31 and a space-variation polynomial order of 3. We added the artificial stars to the selected images with magnitudes from 17.0 to 19.5 magnitudes at random positions. All positive candidates are selected with the human check and rejected for the wrong results.

The negative candidates are bogus stamps caused by the residuals of alignments, isolated random hot pixels after resampling, cosmic rays, saturation stars, and some failed fitting sources. The negative candidates are produced from the image subtraction pipeline with real images and also selected by humans. We cut the stamps for positive and negative candidates with the image subtraction pipeline. We constructed data sets with 10^4 positive and 10^4 negative candidates. Examples of the train set are shown in **Figure 11**.

Before the CNN training, we split 20% of candidates into the test group for training and validation groups. The rotational-invariant residual neural network trains with the optimizer AdamW, batches of 256 stamp arrays, and a dropout rate of 0.5 in the fully connected layer. The learning rate is set to a relatively low value of 0.01 at the beginning, and it steps down by multiplying by 0.9 after each epoch. We built the neural network with PyTorch (Paszke et al., 2019) and trained it on the NVIDIA A100 graphic processor unit. The training of our model requires a huge GPU memory, especially for the stamp size of 91 × 91 for the stamps from the DIP pipeline. The accuracies and precisions of the CNN models for QIP and DIP programs are shown in **Table 3**.

4.3 Rotational-Invariant Residual Neural Network Performance

We calculated the false-negative rate (FNR) and the false-positive rate (FPR) with scikit-learn to analyze the neural network's performance. The detection error trade-off (DET) curve demonstrates how FNR is correlated with FPR. The DET curves could show the performance of the CNN models used for candidate classification and provide direct feedback on the detection error trade-offs to help analyze the neural network. In **Figure 12**, we presented the DET curve for candidates in different signal-to-noise (SNR) groups. The higher SNR curve shows a quicker move to the bottom left, better fitting the plot.

Figure 12 shows that the rotational-invariant model is well-operated for the higher signal-to-noise ratio sources, which is in line with our expectations. The curve for higher SNR shows a vertical line, which may be caused by having only 10^4 sources as positive stamps.

5 CONCLUSION

This article described the science data reduction pipeline and transient detection pipeline for the AST3-3 telescope at the Yaoan Observation Station. The science image pipeline uses the statistical method to estimate the quality of the observed image, taking into account the effects of poor weather, such as clouds passing through in the FoV.

The transient detection pipeline uses multiple binned and unbinned science images to extract the candidates faster and deeper. In terms of transient source detection, the robustness and flexibility of the program are improved through a combination of multiple detection methods and the CNN method. We introduced a rotation-invariant residual neural network to classify the candidate stamps from the transient detection pipeline. The CNN trained on the negative and simulated

positive stamps cut. The CNN accuracy achieved 99.87% for the QIP and 99.20% for the DIP. AST3-3 has been designed for robotic observation and has a complete pipeline system with specific software, a well-trained CNN model, and management for the observation at the Yaoan Observation Station. This work allowed us to effectively participate in the LIGO-Virgo O4 ground optical follow-up observing campaign.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material; further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

TS contributed to the design and whole system of the study. TS and LH organized the database of transient detection and classification. XL and KM worked on the hardware supplement. All authors contributed to manuscript revision and read and approved the submitted version.

FUNDING

This work is partially supported by the National Natural Science Foundation of China (Grant Nos. 11725314 and 12041306), the Major Science and Technology Project of Qinghai Province (2019-ZJ-A10), the ACAMAR Postdoctoral Fellow, the China Postdoctoral Science Foundation (Grant No. 2020M681758) and the Natural Science Foundation of Jiangsu Province (grant No. BK20210998). TS and AC also acknowledge financial support from the State Agency for Research of the Spanish MCIU through the " Center of Excellence Severo Ochoa" award to the Instituto de Astrofísica de Andalucía (SEV-2017-0709). TS acknowledges the China Scholarship Council (CSC) for funding his PhD scholarship (202006340174).

ACKNOWLEDGMENTS

The AST3-3 team would like to express their sincere thanks to the staff of the Yaoan Observation Station. The authors are also grateful to anonymous referees whose opinion has significantly improved this manuscript. This research has made use of data and services provided by the International Astronomical Union's Minor Planet Center. This work has made use of data from the European Space Agency (ESA) mission *Gaia* (https://www.cosmos.esa.int/gaia), processed by the *Gaia* Data Processing and Analysis Consortium (DPAC, https://www.cosmos.esa.int/web/gaia/dpac/consortium). Funding for the DPAC has been provided by national institutions, in particular the institutions participating in the *Gaia* Multilateral Agreement. Software packages: this research made use of Astropy,[1] a community-developed core Python package for Astronomy (Astropy Collaboration et al., 2013, 2018). The python packages: L.A. Cosmic (van Dokkum et al., 2012), PyEphem (Rhodes, 2011), Skyfield (Rhodes, 2019), PyTorch (Paszke et al., 2019), scikit-image (Van der Walt et al., 2014), scikit-learn (scikit-learn developers, 2020), Matplotlib (Hunter, 2007), SciPy(Jones et al., 2001), statsmodels (Seabold and Perktold, 2010), sep (Barbary, 2016), and SFFT (Hu et al., 2021). Software programs: SExtractor (Bertin and Arnouts, 2010), SWarp (Bertin, 2010), Astrometry.net (Lang et al., 2012), FITSH (Pál, 2011), and fpack (Seaman et al., 2010).

[1]http://www.astropy.org

REFERENCES

Alard, C. (2000). Image Subtraction Using a Space-Varying Kernel. *Astron. Astrophys. Suppl. Ser.* 144, 363–370. doi:10.1051/aas:2000214

Alard, C., and Lupton, R. H. (1998). A Method for Optimal Image Subtraction. *ApJ* 503, 325–331. doi:10.1086/305984

Andreoni, I., Jacobs, C., Hegarty, S., Pritchard, T., Cooke, J., and Ryder, S. (2017). *Mary, a Pipeline to Aid Discovery of Optical Transients*, 34. New York: Publications of the Astron. Soc. of Australia, e037. doi:10.1017/pasa.2017.33

Astropy CollaborationPrice-Whelan, A. M., Price-Whelan, A. M., Sipőcz, B. M., Günther, H. M., Lim, P. L., Crawford, S. M., et al. (2018). The Astropy Project: Building an Open-Science Project and Status of the v2.0 Core Package. *Aj* 156, 123. doi:10.3847/1538-3881/aabc4f

Astropy CollaborationRobitaille, T. P., Robitaille, T. P., Tollerud, E. J., Greenfield, P., Droettboom, M., Bray, E., et al. (2013). Astropy: A Community Python Package for Astronomy. *A&A* 558, A33. doi:10.1051/0004-6361/201322068

Barbary, K. (2016). SEP: Source Extractor as a Library. *Joss* 1, 58. doi:10.21105/joss.00058

Barthelmy, S. (2008). GCN and VOEvent: A Status Report. *Astron. Nachr.* 329, 340–342. doi:10.1002/asna.200710954

Becker, A. C., Homrighausen, D., Connolly, A. J., Genovese, C. R., Owen, R., Bickerton, S. J., et al. (2012). Regularization Techniques for PSF-Matching Kernels - I. Choice of Kernel Basis. *Mon. Notices R. Astronomical Soc.* 425, 1341–1349. doi:10.1111/j.1365-2966.2012.21542.x

Bertin, E., and Arnouts, S. (1996). SExtractor: Software for Source Extraction. *Astron. Astrophys. Suppl. Ser.* 117, 393–404. doi:10.1051/aas:1996164

Bertin, E., and Arnouts, S. (2010). *SExtractor: Source Extractor*. Houghton, Michigan: Astrophysics Source Code Library. record ascl:1010.064.

Bertin, E. (2013). *PSFEx: Point Spread Function Extractor*.

Bertin, E. (2010). *SWarp: Resampling and Co-adding FITS Images Together*.

Bramich, D. M. (2008). A New Algorithm for Difference Image Analysis. *Mon. Not. Ras. Lett.* 386, L77–L81. doi:10.1111/j.1745-3933.2008.00464.x

Bramich, D. M., Horne, K., Albrow, M. D., Tsapras, Y., Snodgrass, C., Street, R. A., et al. (2013). Difference Image Analysis: Extension to a Spatially Varying Photometric Scale Factor and Other Considerations. *Mon. Notices R. Astronomical Soc.* 428, 2275–2289. doi:10.1093/mnras/sts184

Brennan, S. J., and Fraser, M. (2022). *The AUTOmated Photometry of Transients (AutoPhOT) Pipeline*. arXiv e-prints, arXiv:2201.02635.

Cabrera-Vives, G., Reyes, I., Förster, F., Estévez, P. A., and Maureira, J.-C. (2017). Deep-HiTS: Rotation Invariant Convolutional Neural Network for Transient Detection. *ApJ* 836, 97. doi:10.3847/1538-4357/836/1/97

Calabretta, M., and Greisen, E. W. (2000). "Representations of World Coordinates in FITS," *Astronomical Society of the Pacific Conference Series*. Editors N. Manset, C. Veillet, and D. Crabtree (San Francisco: Astronomical Data Analysis Softw. Syst. IX), 216, 571.

Cao, Y., Nugent, P. E., and Kasliwal, M. M. (2016). *Intermediate Palomar Transient Factory: Realtime Image Subtraction Pipeline*, 128. Bristol: Publications of the ASP, 114502. doi:10.1088/1538-3873/128/969/114502

Carrasco-Davis, R., Reyes, E., Valenzuela, C., Förster, F., Estévez, P. A., Pignata, G., et al. (2021). Alert Classification for the ALeRCE Broker System: The Real-Time Stamp Classifier. *Aj* 162, 231. doi:10.3847/1538-3881/ac0ef1

Chambers, K. C.Pan-STARRS Team (2017). "The Pan-STARRS1 Survey Data Release," in *American Astronomical Society Meeting Abstracts #229* (American Astronomical Society Meeting Abstracts), 223–03.

Craig, M. W., Crawford, S. M., Deil, C., Gomez, C., Günther, H. M., Heidt, N., et al. (2015). *Ccdproc: CCD Data Reduction Software*.

Dálya, G., Díaz, R., Bouchet, F. R., Frei, Z., Jasche, J., Lavaux, G., et al. (2021). *GLADE+: An Extended Galaxy Catalogue for Multimessenger Searches with Advanced Gravitational-Wave Detectors*. arXiv e-prints, arXiv: 2110.06184.

Dieleman, S., Willett, K. W., and Dambre, J. (2015). Rotation-invariant Convolutional Neural Networks for Galaxy Morphology Prediction. *Mon. Notices RAS* 450, 1441–1459. doi:10.1093/mnras/stv632

Förster, F., Maureira, J. C., San Martín, J., Hamuy, M., Martínez, J., Huijse, P., et al. (2016). The High Cadence Transient Survey (HITS). I. Survey Design and Supernova Shock Breakout Constraints. *AstronoAstrophysical J.* 832, 155. doi:10.3847/0004-637X/832/2/155

Gaia CollaborationBrown, A. G. A., Vallenari, A., Prusti, T., de Bruijne, J. H. J., Babusiaux, C., Bailer-Jones, C. A. L., et al. (2018). Gaia Data Release 2. Summary of the Contents and Survey Properties. *Astronomy Astrophysics* 616, A1. doi:10.1051/0004-6361/201833051

Goldstein, D. A., D'Andrea, C. B., Fischer, J. A., Foley, R. J., Gupta, R. R., Kessler, R., et al. (2015). Automated Transient Identification in the Dark Energy Survey. *Astronomical J.* 150, 82. doi:10.1088/0004-6256/150/3/8210.1088/0004-6256/150/5/165

Gómez, C., Neira, M., Hernández Hoyos, M., Arbeláez, P., and Forero-Romero, J. E. (2020). Classifying Image Sequences of Astronomical Transients with Deep Neural Networks. *Mon. Notices RAS* 499, 3130–3138. doi:10.1093/mnras/staa2973

Greenfield, P., Robitaille, T., Tollerud, E., Aldcroft, T., Barbary, K., Barrett, P., et al. (2013). *Astropy*. Houghton, Michigan: Community Python library for astronomy.

He, K., Zhang, X., Ren, S., and Sun, J. (2015). *Deep Residual Learning for Image Recognition*. arXiv e-prints, arXiv:1512.03385.

Hochreiter, S., and Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Comput.* 9, 1735–1780. doi:10.1162/neco.1997.9.8.1735

Howard, A., Sandler, M., Chu, G., Chen, L.-C., Chen, B., Tan, M., et al. (2019). *Searching for MobileNetV3*. arXiv e-prints, arXiv:1905.02244.

Hu, L., Wang, L., and Chen, X. (2021). *Image Subtraction in Fourier Space*. arXiv e-prints, arXiv:2109.09334.

Hu, L., Wang, L., and Chen, X. (2021). *Sfft*. doi:10.5281/zenodo.5521634

Hunter, J. D. (2007). Matplotlib: A 2D Graphics Environment. *Comput. Sci. Eng.* 9, 90–95. doi:10.1109/mcse.2007.55

Irwin, J., Irwin, M., Aigrain, S., Hodgkin, S., Hebb, L., and Moraux, E. (2007). The Monitor Project: Data Processing and Light Curve Production. *Mon. Notices RAS* 375, 1449–1462. doi:10.1111/j.1365-2966.2006.11408.x

Jia, P., Zhao, Y., Xue, G., and Cai, D. (2019). Optical Transient Object Classification in Wide-Field Small Aperture Telescopes with a Neural Network. *Aj* 157, 250. doi:10.3847/1538-3881/ab1e52

Jones, E., Oliphant, T., and Peterson, P. (2001). *SciPy: Open Source Scientific Tools for Python*.

Kashyap, V. L., van Dyk, D. A., Connors, A., Freeman, P. E., Siemiginowska, A., Xu, J., et al. (2010). On Computing Upper Limits to Source Intensities. *ApJ* 719, 900–914. doi:10.1088/0004-637X/719/1/900

Lang, D., Hogg, D. W., Mierle, K., Blanton, M., and Roweis, S. (2012). *Astrometry.net: Astrometric Calibration of Images*. Houghton, Michigan: Astrophysics Source Code Library. record ascl:1208.001.

Lang, D., Hogg, D. W., Mierle, K., Blanton, M., and Roweis, S. (2010). Astrometry.net: Blind Astrometric Calibration of Arbitrary Astronomical Images. *Astronomical J.* 139, 1782–1800. doi:10.1088/0004-6256/139/5/1782

Lecun, Y., Bengio, Y., and Hinton, G. (2015). Deep Learning. *Nature* 521, 436–444. doi:10.1038/nature14539

Li, X., Yuan, X., Gu, B., Yang, S., Li, Z., and Du, F. (2019). "Chinese Antarctic Astronomical Optical Telescopes," in *Revista Mexicana de Astronomia y Astrofisica Conference Series* (Mexico: Revista Mexicana de Astronomia y Astrofisica Conference Series), 51, 135–138. doi:10.22201/ia.14052059p.2019.51.23

Loshchilov, I., and Hutter, F. (2017). *Decoupled Weight Decay Regularization*. *arXiv e-prints* arXiv:1711.05101.

Masci, F. J., Laher, R. R., Rusholme, B., Shupe, D. L., Groom, S., Surace, J., et al. (2019). *The Zwicky Transient Facility: Data Processing, Products, and Archive*, 131. Bristol: Publications of the ASP, 018003. doi:10.1088/1538-3873/aae8ac

Mommert, M. (2017). PHOTOMETRYPIPELINE: An Automated Pipeline for Calibrated Photometry. *Astronomy Comput.* 18, 47–53. doi:10.1016/j.ascom.2016.11.002

Morganson, E., Gruendl, R. A., Menanteau, F., Kind, M. C., Chen, Y.-C., Daues, G., et al. (2018). *The Dark Energy Survey Image Processing Pipeline*, 130. Bristol: Publications of the ASP, 074501. doi:10.1088/1538-3873/aab4ef

Nair, V., and Hinton, G. E. (2010). Rectified Linear Units Improve Restricted Boltzmann Machines. *ICML*, 807–814.

National Optical Astronomy Observatories (1999). *IRAF: Image Reduction and Analysis Facility*.

Paillassa, M., Bertin, E., and Bouy, H. (2020). MAXIMASK and MAXITRACK: Two New Tools for Identifying Contaminants in Astronomical Images Using Convolutional Neural Networks. *A&A* 634, A48. doi:10.1051/0004-6361/201936345

Pál, A. (2012). FITSH- a Software Package for Image Processing. *Mon. Notices RAS* 421, 1825–1837. doi:10.1111/j.1365-2966.2011.19813.x

Pál, A. (2011). *FITSH: Software Package for Image Processing*. Houghton, Michigan: Astrophysics Source Code Library. record ascl:1111.014.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., et al. (2019). "Pytorch: An Imperative Style, High-Performance Deep Learning Library," in *Advances in Neural Information Processing Systems 32*. Editors H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, and R. Garnett (Cambridge, MA: Curran Associates, Inc.), 8024–8035.

Pearson, K., and Galton, F. (1895). Vii. Note on Regression and Inheritance in the Case of Two Parents. *Proc. R. Soc. Lond.* 58, 240–242. doi:10.1098/rspl.1895.0041

Pence, W. D., Seaman, R., and White, R. L. (2009). *Lossless Astronomical Image Compression and the Effects of Noise*, 121. Bristol: Publications of the ASP, 414–427. doi:10.1086/599023

Price, P. A., and Magnier, E. A. (2019). *Pan-STARRS PSF-Matching for Subtraction and Stacking*. arXiv e-prints.

Rhodes, B. C. (2011). *PyEphem: Astronomical Ephemeris for Python*.

Rhodes, B. (2019). *Skyfield: High Precision Research-Grade Positions for Planets and Earth Satellites Generator*. Houghton, Michigan: Astrophysics Source Code Library. record ascl:1907.024.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning Representations by Back-Propagating Errors. *Nature* 323, 533–536. doi:10.1038/323533a0

scikit-learn developers (2020). *Scikit-learn : Machine Learning in Python*.

Seabold, S., and Perktold, J. (2010). "Statsmodels: Econometric and Statistical Modeling with python," in *9th Python in Science Conference*. doi:10.25080/majora-92bf1922-011

Seaman, R., Pence, W., and White, R. (2010). *Fpack: FITS Image Compression Program*.

Shupe, D. L., Moshir, M., Li, J., Makovoz, D., Narron, R., and Hook, R. N. (2005). "The SIP Convention for Representing Distortion in FITS Image Headers,"In *Astronomical Data Analysis Software and Systems XIV*. Editors P. Shopbell, M. Britton, and R. Ebert (San Francisco: Astronomical Society of the Pacific Conference Series), 347, 491.

Sokolovsky, K. V., and Lebedev, A. A. (2018). VaST: A Variability Search Toolkit. *Astronomy Comput.* 22, 28–47. doi:10.1016/j.ascom.2017.12.001

Van der Walt, S., Schönberger, J. L., Nunez-Iglesias, J., Boulogne, F., Warner, J. D., Yager, N., et al. (2014). Scikit-Image: Image Processing in python. *PeerJ* 2, e453. doi:10.7717/peerj.453

van Dokkum, P. G., Bloom, J., and Tewes, M. (2012). *L.A.Cosmic: Laplacian Cosmic Ray Identification*. Houghton, Michigan: Astrophysics Source Code Library. record ascl:1207.005.

van Dokkum, P. G. (2001). *Cosmic-Ray Rejection by Laplacian Edge Detection*, 113. Bristol: Publications of the ASP, 1420–1427. doi:10.1086/323894

Yin, K., Jia, J., Gao, X., Sun, T., and Zhou, Z. (2021). Supernovae Detection with Fully Convolutional One-Stage Framework. *Sensors* 21, 1926. doi:10.3390/s21051926

Yuan, X., Cui, X., Wang, L., Gu, B., Du, F., Li, Z., et al. (2015). The Antarctic Survey Telescopes AST3 and the AST3-NIR. *IAU General Assem.* 29, 2256923.

Zhang, J.-C., Wang, X.-F., Mo, J., Xi, G.-B., Lin, J., Jiang, X.-J., et al. (2020). *The Tsinghua University-Ma Huateng Telescopes for Survey: Overview and Performance of the System*, 132. Bristol: Publications of the ASP, 125001. doi:10.1088/1538-3873/abbea2

Zhang, T.-M., Wang, X.-F., Chen, J.-C., Zhang, J.-J., Zhou, L., Li, W.-X., et al. (2015). The THU-NAOC Transient Survey: the Performance and Results from the First Year. *Res. Astron. Astrophys.* 15, 215–224. doi:10.1088/1674-4527/15/2/006

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers | Frontiers in Astronomy and Space Sciences

METHODS
published: 11 July 2022
doi: 10.3389/fspas.2022.891486

pyobs - An Observatory Control System for Robotic Telescopes

Tim-Oliver Husser[1]*, *Frederic V. Hessman*[1], *Sven Martens*[1], *Tilman Masur*[1,2], *Karl Royen*[1] *and Sebastian Schäfer*[1]

[1]Institute for Astrophysics and Geophysics, Göttingen University, Göttingen, Germany, [2]TNG Technology Consulting GmbH, Unterföhring, Germany

We present a Python-based framework for the complete operation of a robotic telescope observatory. It provides out-of-the-box support for many popular camera types while other hardware like telescopes, domes, and weather stations can easily be added via a thin abstraction layer to existing code. Common functionality like focusing, acquisition, auto-guiding, sky-flat acquisition, and pipeline calibration are ready for use. A remote-control interface, a "mastermind" for truly robotic operations as well as an interface to the Las Cumbres Observatory observation portal is included. The whole system is fully configurable and easily extendable. We are currently running pyobs successfully on three different types of telescopes, of which one is a siderostat for observing the Sun. pyobs uses open standards and open software wherever possible and is itself freely available.

Keywords: methods: observational, telescopes, techniques: image processing, techniques: photometric, techniques: spectroscopic

OPEN ACCESS

Edited by:
Aaron Golden, National University of Ireland Galway, Ireland

Reviewed by:
Ronan Cunniffe, Institute of Physics (ASCR), Czechia
Jean Baptiste Marquette, UMR5804 Laboratoire d'Astrophysique de Bordeaux (LAB), France
Martin Dyer, The University of Sheffield, United Kingdom

***Correspondence:**
Tim-Oliver Husser
thusser@uni-goettingen.de

Specialty section:
This article was submitted to Astronomical Instrumentation, a section of the journal Frontiers in Astronomy and Space Sciences

Received: *07 March 2022*
Accepted: *23 May 2022*
Published: *11 July 2022*

Citation:
Husser T-O, Hessman FV, Martens S, Masur T, Royen K and Schäfer S (2022) pyobs - An Observatory Control System for Robotic Telescopes. Front. Astron. Space Sci. 9:891486. doi: 10.3389/fspas.2022.891486

1 INTRODUCTION

At the turn of the millennium, a major change was starting to take place in observational astronomy: at first unnoticed by most astronomers, many new telescopes were refitted or newly built for remote or even fully autonomous observations. While there were lots of discussions and even a few robotic telescopes in the early 90 s (see, e.g., Perlmutter et al., 1992; Alcock et al., 1992), their number grew significantly in the decades thereafter.

While most of the very early robotic telescopes simply monitored known variable stars (e.g., Henry et al., 1995; Strassmeier et al., 1997), those that followed were designed to permit very rapid follow-up of gamma-ray bursts–e.g., ROTSE-III (Akerlof et al., 2003), REM (Antonelli et al., 2003), and BOOTES (Castro-Tirado et al., 2004)—or search for exoplanets (e.g., with SuperWASP, Street et al., 2003), or to survey galaxies for supernovae (e.g., Filippenko et al., 2001; Lipunov et al., 2007). As automation became easier and pipeline software more powerful, it was possible to survey automatically for any transients or moving Solar System objects, e.g., with the Intermediate Palomar Transient Factory (iPTF, Law et al., 2009), which later was refitted to become the Zwicky Transient Facility (ZTF, Bellm et al., 2019; Riddle et al., 2018). About the same time, Las Cumbres Observatory (LCO) started building a whole network of robotic telescopes (Hidas et al., 2008), now one of the largest in the world.

Robotic telescopes can be used for many things–from the automated performance of a heterogeneous list of independent observations to the dedicated performance of a particular scientific project. The unique science that can be done with robotic telescopes almost exclusively concerns transients, i.e., changes over time on the sky of any kind. The ultimate source for such targets in the near future will be the Legacy Survey of Space and Time (LSST) at the Vera C. Rubin

Observatory, an 8.4 m telescope designed for surveying the sky for any kind of transients (Ivezić et al., 2019). When the LSST starts operating in 2023, a legion of other robotic telescopes will begin doing follow-up observations on the detected transients.

All truly robotic telescopes require a wide palette of hardware–e.g., computer-controlled telescopes, cameras, filter wheels, enclosures, and weather stations–and software for autonomous operation of the entire system. While this software can be rather basic (to avoid the fully unwarranted word "simple") for surveys that just do the same thing over and over again, it gets immensely more complicated for all-purpose telescopes. Unfortunately, this software almost never gets published or placed in a form which is useful for another project, mostly for the reason that it is very specific to the hardware and the science case at hand.

Luckily, there is some software available that tries to be applicable to many different hardware devices and kinds of observations. Especially popular with amateur astronomers is the Windows COM-based ASCOM system[1], which defines generic interfaces for different kinds of devices and can be used by several client applications. A couple of years ago, a HTTP REST based interface called Alpaca was released, which allows the use of ASCOM in Unix-like systems as well. Additional powerful software like ACP from DC3.com can be used to help automate operations within an ASCOM network. A very different system but with the same basic philosophy and breadth of support is the Instrument Neutral Distributed Interface (INDI).[2], which was designed for network transparency from the beginning and can be used from any system and programmed in any language, although the core libraries are written in C++. The 2nd version of the Remote Telescope System (RTS2 for short; Kubánek et al. (2004)) is widely used in a variety of mostly scientific projects. It provides a complete framework–including a back-end database–and is designed for fully autonomous operations. RTS2 is written in C++ and runs on Linux only.

For our own MONET telescopes (Hessman, 2004, see also **Section 6**) we first successfully used the robotic control software developed for their twins, the STELLA telescopes (Granzer, 2006; Granzer et al., 2012) on Tenerife, operated by the AIP in Potsdam, which was thankfully made available to us by our colleagues there. While the system itself is written in Java, over time we started to implement some functionality using more familiar *Python* scripts. These scripts grew and at some point became pyobs, a fully functional observation control system for robotic telescopes on its own. There was originally no other good reason for developing pyobs than this; without pyobs and starting from scratch, we probably would have chosen INDI. However, pyobs has now grown to a level where it is just as powerful as INDI, RTS2, or ASCOM: it is highly flexible, uses open standards, and is programmed in the language most commonly used by astronomers. Indeed, it stands on the shoulders of giants that are the many amazing open source *Python* projects used in computer science and astronomy. In this paper we will present its architecture and the basic functionality.

We strongly believe in acknowledging the work other people put into publicly available (open-source) software, and thus, references for all the third party software projects used in pyobs are listed in the Acknowledgments. All pyobs packages themselves are published as open-source under the MIT license at GitHub[3], and its documentation is also available online.[4]

2 ARCHITECTURE

The astronomical community has spent the last 2 decades migrating from diverse programming languages like IDL, FORTRAN, or C/C++ to a common denominator, which turned out to be *Python*. As a result, today we have powerful scientific libraries available like NumPy, SciPy, and AstroPy. Following this progress, *Python* was an easy pick as the language of choice for a new Observatory Control System (OCS).

Nevertheless, *Python* has some drawbacks for a large project like this, with the "global interpreter lock" (GIL) being the most significant. The GIL is a multi-threading lock (or "mutex") that can only be acquired by one thread at a time. So, although *Python* supports the creation and running of multiple threads, they never run in parallel. The only way to achieve true parallelism is to use multi-processing, so a decision was made to run pyobs in multiple processes, i.e., one process per block of functionality, which, in pyobs terminology, is called a "module". A module can be everything from a controller for an actual hardware device to routines for, e.g., an auto-focus series. With the OCS being split up into multiple processes, the communication between them became one of the most important parts of pyobs.

2.1 Asyncio

Given the already mentioned problems with multi-threading in *Python*, it is only logical to rethink the use of threads in pyobs in the first place. Most modules in pyobs do one thing most and foremost: waiting. Waiting for a command to execute, waiting for an exposure to finish, waiting for the dome to move into position. However, in pyobs many things still need to be run concurrently, e.g., a module should still be accepting commands while moving a telescope. Luckily, *Python* introduced a new way of handling concurrency in version 3.5 and improved it steadily in the years thereafter. The new asyncio package uses a main loop and switches between tasks on request, all on a single CPU core and in a single thread. This avoids typical problems in multi-threading like deadlocks and run conditions. However, calling a blocking function in asyncio blocks all other tasks as well, so there is also an easy way for running single methods in an extra thread and waiting for it.

Functions that are running within the asyncio loop are called *coroutines* and are defined with the `async` keyword, as will be shown for the interfaces:

[1]https://ascom-standards.org
[2]https://indilib.org
[3]https://github.com/pyobs
[4]https://www.pyobs.org

```
class IPointingRaDec(Interface):
  async def move_radec(ra: float, dec: float):
    ...
```

Coroutines can only be called directly from other coroutines and always need to be "awaited":

```
async def test():
  await telescope. move_radec(1., 2.)
```

They can also be called without actually waiting for them to finish. In those cases, a task should be created which can be awaited later:

```
task = asyncio. create_task(telescope.
move_radec(1., 2.))
...
await task
```

In pyobs, this is, for instance, used for requesting FITS headers from other modules before an exposure is started. The module creates tasks for requesting the headers, but only awaits them after the image has finished, in order not to delay the start of the exposure.

As mentioned above, asyncio heavily reduces the risk of multi-threading related problems. That is, because tasks never run in parallel, but are only switched when one has finished or when something is awaited. In multi-threading, parts of the code that should not be interrupted are often secured using a mutex (or lock), which is mostly unnecessary when using asyncio.

With asyncio, one just needs to be careful with long running functions that are not defined `async`, e.g. the readout processes of some cameras. Those method calls would block the whole module, so asyncio provides an easy way to run them in an extra thread:

```
loop = asyncio. get_running_loop()
data = await loop. run_in_executor(None,
camera. read_out())
```

Altogether, pyobs make heavy use of asyncio. For instance, all interface methods and all event handlers must be defined `async`. Switching from multi-threading to asyncio massively reduced the number of difficult-to-debug errors and made developing a lot easier.

2.2 Communication

Instead of inventing our own protocol for communication, we decided to use XMPP (Saint-Andre, 2004), an XML-based chat protocol. With it being mainly used for instant messaging (e.g., by Jabber, WhatsApp, Zoom, Jitsi, and others), it naturally supports multi-user chat, i.e., sending messages to multiple users. But due to its wide variety of extensions (XMPP Extension Protocol, XEP), it also supports remote procedure calls (RPC, calling methods on another client), and a feature called auto-discovery, which allows one client to determine the capabilities of another.

The use of XMPP also frees us from writing and maintaining our own server software, since there are multiple industrial-grade servers available, like ejabberd[5] and Openfire[6]. They can run with tens of thousands of users, compared to maybe a few dozen pyobs clients in a typical observatory. Although, admittedly, pyobs sends more messages than even the most ambitious teenager in WhatsApp.

While we use the *Python* package Slixmpp for pyobs itself, there are also XMPP libraries available for all major programming languages[7]. Therefore the "py" (for "*Python*") in "pyobs" refers only to the core package, but extension modules can be written in any language that supports XMPP.

As **Figure 1** shows, the communication in pyobs is based on three pillars (remote procedure calls, interfaces, events), which all will be discussed in more detail in the following.

2.2.1 Remote Procedure Calls

All methods within a module that are derived from an interface (see below) can be called remotely. The easiest way to do so, is to get a `Proxy` object for another module from pyobs. These objects mimic the behavior of the original module and therefore any of their methods can be called directly as if they were local.

For instance, a camera module might implement this method:

```
async def set_exposure_time(
    self,
    exposure_time: float,
    **kwargs: Any
) -> None:
```

For another module, calling this method is a simple as:

```
camera = await self. proxy(name_of_camera
_module)
await camera. set_exposure_time(2.0)
```

While some methods should usually return immediately (e.g., requesting a position), some might take a longer time (e.g., exposing an image or moving a telescope). For the caller of a method it would be good to have an estimate for the call duration in order to avoid waiting forever in case of an error. To achieve this, pyobs extends the XEP-0009 extension for RPCs with a timeout mechanism: all methods can define a time after which they should be finished. This time is sent back to the caller immediately after a method is called. If this waiting time is exceeded, a timeout exception is raised and the caller can decide what to do about this. If a method takes longer than 10 s, it should be decorated with the `@timeout` decorator, which defines the maximum duration:

```
@timeout(1,200)
async def move_radec(
```

[5]https://www.ejabberd.im

[6]https://www.igniterealtime.org/projects/openfire/

[7]see, e.g., https://xmpp.org/software/libraries/

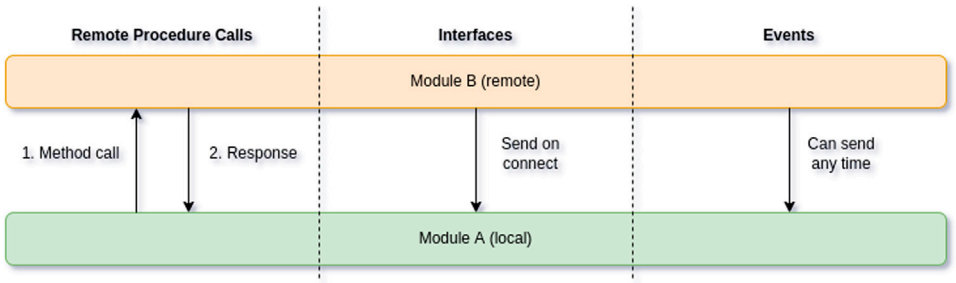


FIGURE 1 | The three pillars of communication in pyobs, described as interactions between a local module A and a remote module B. On the left, remote procedure calls are actively called on another module. The list of interfaces, in the middle, is automatically retrieved after the connection to the XMPP server has been established. And events can be sent at any time, as shown on the right. Only those events can be handled in a module that it had registered before.

```
    self,
    ra: float,
    dec: float,
    **kwargs: Any
) -> None:
```

Calling a method like this works the same way as before, although it now raises an exception only after 1,200 s, compared to 10 s for un-decorated methods.

2.2.2 Interfaces

The basis for all RPCs in pyobs are the interfaces in `pyobs.interfaces`, which describe methods that a module must implement in order to provide a given functionality. For instance, all telescope modules should implement the `ITelescope` interface. While not defining any methods on its own, it inherits the two methods `move_radec` and `get_radec` from `IPointingRaDec` (shortened for clarity):

```
class IPointingRaDec(Interface):
  @abstractmethod
  async def move_radec(
    self,
    ra: float,
    dec: float,
  ) -> None:
    ...

  @abstractmethod
  async def get_radec(self, ) -> Tuple[float, float]:
    ...

class ITelescope(IPointingRaDec):
  ...
```

Therefore, in order to be a valid `ITelescope`, a module must implement these methods.

All interfaces implemented by a module are published via XMPP's auto-discovery extension, so all other modules can easily determine what functionality is available from a given module. This allows for easy construction of `Proxy` objects for RPC. Furthermore, it is extremely simple for a module to find all other modules that implement a given interface. A good example for this are the interfaces `IFitsHeaderBefore` and `IFitsHeaderAfter`. When a camera starts a new exposure, we usually want to collect FITS headers from different modules. Instead of having this list pre-defined, the camera can just request all modules that implement these interfaces and call their respective methods before and after the exposure:

```
clients = await self.comm.clients_with_interface(
  IFitsHeaderBefore
)
for client in clients:
  proxy = await self.proxy(client,
  IFitsHeaderBefore)
  headers[client] =
  await proxy.get_fits_header_
  before()
```

This way, we can easily add a new module to the system that simply provides new headers for new FITS files (e.g., with weather data).

As an example, **Figure 2** shows parts of the inheritance for DummyTelescope, a simulated telescope that can be used for testing.

2.2.3 Events

While RPC is an active process of communicating with other modules, there is also a passive one, which is reacting to events. Each module can define types of events that itself creates and that it wants to receive from other modules.

For instance, a camera might want to declare that it can send events, when a new image has been taken

```
await self.comm.register_event(
    NewImageEvent
)
```

and can actually send those events:

```
await self.comm.send_event(
    NewImageEvent(filename, image_type)
)
```

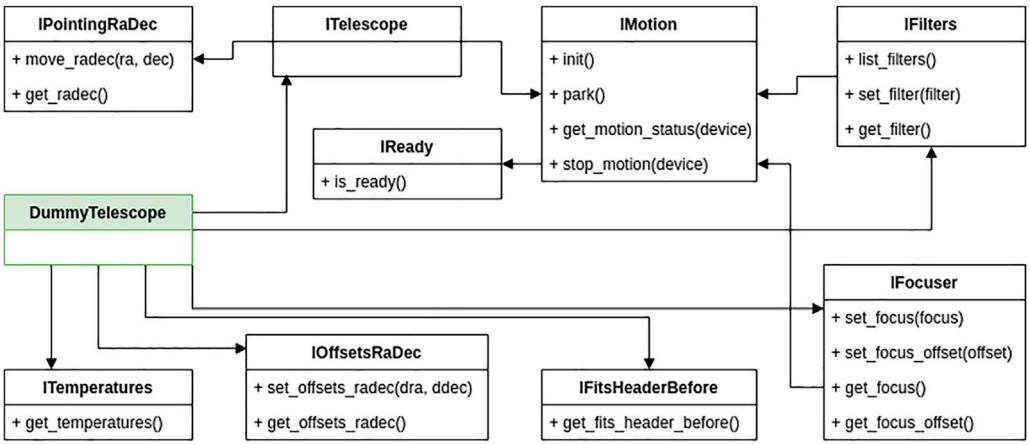


FIGURE 2 | Part of the interface inheritance for `DummyTelescope` (on the left in green), a simulated telescope that accepts RA/Dec coordinates and offsets and has a filter wheel, a focus unit, and some temperature sensors. All methods available for remote calls are defined in the interfaces. ITelescope does not define any method of its own, but is just a collection of other interfaces and can be used as a device definition, i.e., "this is a telescope".

while another module might want to receive those events and handle them in a callback method:

```
await self.comm.register_event(
    NewImageEvent,
    self.on_new_image
)

[...]

async def on_new_image(
    self,
    event: Event,
    sender: str
) -> bool:
    print(event)
```

The events can be chained by sending new events within a handler method. As an example, events on new images from a camera could be handled by an image pipeline, which in turn sends events that are handled by a module that measures seeing on the reduced images.

2.3 Configuration

Pyobs gets its high flexibility from configuration files in YAML format. The most simple configuration consists of only a single line like:

```
class: pyobs.modules.test.StandAlone
```

When running this configuration via `pyobs config. yaml` from the command line, a new module is created from the given class and started. The class to use is given by its full package name, the same as one would use to import it in a *Python* shell. Therefore, its definition could be anywhere within the *Python* path and not just in the pyobs package.

The example in the documentation is a little longer:

```
class: pyobs.modules.test.StandAlone
message: Hello world
interval: 10
```

Comparing this with the signature of the constructor of the given class:

```
class StandAlone(Module):
def __init__(
    self,
    message: str = "Hello world",
    interval: int = 10,
    **kwargs: Any
):
```

This makes it clear that all items in the configuration are simply forwarded directly to the constructor of the given class. pyobs goes even a step further and allows many parameters to be either an object or a configuration dictionary (mostly given in a YAML file as in the example above), describing an object of the same type. For instance, every module class has also a parameter comm (derived from pyobs' `Object` class) for defining its method for communication with other modules, given as this:

```
comm: Optional[Union[Comm, Dict[str, Any]]] = None
```

So this parameter accepts both a Comm object directly or a description thereof. A valid configuration file could therefore look like this:

```
class: pyobs.modules.test.StandAlone
comm:
    class: pyobs.comm.slixmpp.XmppComm
    jid: test@example.com
```

```
    password: topsecret
```

Looking at the constructor of given class `XmppComm` explains the given parameters (shortened for clarity):

```
class XmppComm(Comm):
  def __init__(
    self,
    jid: Optional[str] = None,
    password: str = "",
  ):
```

Note that this makes it possible to replace the whole communication system via XMPP with another method by just implementing a new class derived from `Comm`. In an environment, in which it is impossible to run an XMPP server, this could simply be replaced by, e.g., direct socket communication or HTTP REST.

A similar configuration style is used for names of remote modules, which are called within a module. Here is the constructor of the default class for taking an auto-focus series (shortened):

```
class AutoFocusSeries(Module, IAutoFocus):
  def __init__(
    self,
    focuser: Union[str, IFocuser],
    camera: Union[str, IImageGrabber],
  ):
```

The class needs two remote modules to work, a camera for taking the images and a focus unit with which it can change the actual focus value. Both are defined to accept either a string or an object implementing the interface that is actually required. While for testing, it might be easier to pass an actual object, at the observatory we usually just set the name of the other module. From this name, a proxy object is being created, which is checked for implementing the given interface. Therefore, in production, a configuration for a focus series might look like this:

```
class: pyobs.modules.focus.AutoFocusSeries
camera: fli230
focuser: telescope
```

Note that all this behavior is completely up to the class that you want to use. So it must implement the flexibility to accept both an object and a description or a remote name. This should be the case for all modules from the core package and the additional packages.

The default is to have one YAML configuration file per module, but pyobs also has a built-in MultiModule, which can run multiple modules in a single process. This is especially helpful in cases when multiple modules need access to the same hardware, which can be implemented using an object that is shared between those modules. A basic example for this is given with `DummyTelescope` and `DummyCamera`, which can share a common world simulation, so that the camera actually can simulate images at the position the telescope is pointing to. However, if not necessary, `MultiModule` should be avoided in favor of a single module per configuration.

If possible, the configuration even allows changing the core behavior of a module. Coming back to the `AutoFocusSeries` class from above, this class itself only defines the functionality for taking a series of images at different focus values. The actual analysis of the images and the calculation of the final best focus is delegated to an object of type `FocusSeries` as defined as a parameter in the constructor:

```
series: Union[Dict[str, Any], FocusSeries]
```

The default implementation in pyobs (`ProjectionFocusSeries` in `utils.focusseries`, see **Section 3.3.1**) collapses the images along their x and y axes, respectively, and calculates moments to get a rough size of the stars. The final best focus is calculated using a hyperbola fit to the series of focus and size data. But, given that this class is explicitly specified in the configuration file, it can easily be changed to another (custom) implementation that derives from `FocusSeries`.

A module might want to make some configuration settings changeable during runtime. This can be handled via the `IConfig` interface, which is implemented by default by all modules and calls internal methods of the form `_set_config_<name>` (if exists) for changing the given variable `<name>`.

2.4 Virtual File System

In a simple pyobs system, all its modules might run on a single computer. In that case, a module storing a file on a local disk can be certain that another module can access it at the same location. An easy workaround for using this system with modules on different machines is to mount (e.g., via NFS or SMB) the required directories on both machines, but even in that case one has to be careful to mount to the same directory, otherwise filenames would not be the same on both.

This is where a virtual file system (VFS) becomes useful: if we could define a "virtual" directory that points to the correct location on all computers, the problem would be solved. pyobs provides a VFS in `pyobs.vfs` and uses it wherever files are accessed. The VFS is automatically available in all modules, although it needs to be configured. A simple VFS configuration (within the module configuration) might look like this:

```
vfs:
  class: pyobs.vfs.VirtualFileSystem
  roots:
    temp:
      class: pyobs.vfs.LocalFile
      root: /data/images
```

The VFS in pyobs uses the concept of "roots" to define where a file is actually located. In this case, one root, `temp`, is defined as a `LocalFile`, which itself has a `root` parameter, pointing to a real directory in the file system–note that `root` here has nothing to do with the

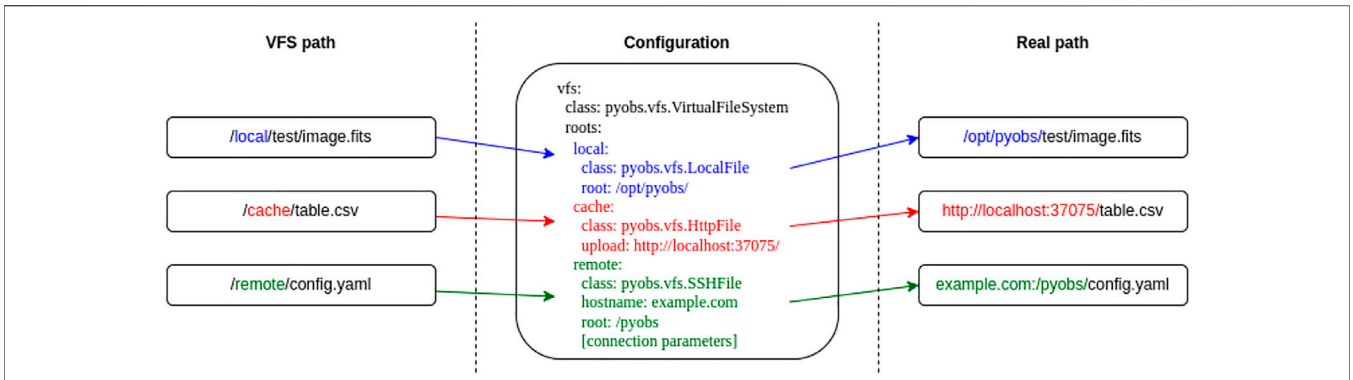


FIGURE 3 | Some examples, how a VFS path (left) maps to a real path (right) with a given configuration (middle). Note that for the `remote` root the class `SSHFile` requires more parameters for the connection, which have been omitted here for clarity.

roots system in pyobs' VFS, but comes from the term "root directory".

Now, within a pyobs module with this configuration we can open a file like this:

```
fd = self.vfs.open_file("/temp/new/image.fits", "r")
```

Internally, pyobs maps the first part of the path (the root), i.e., `temp` in this case, to the root of the same name given in the configuration, so it actually creates a `LocalFile`. When opening the file, the path is changed accordingly to`/data/images/new/image.fits`. Following up on the example from above, now the `temp` root can point to different directories on all computers, but still the same filenames can be used on all.

Since the mounting of remote directories might not be possible in some cases, pyobs offers some more classes for file access within the VFS:

- `ArchiveFile` connects to the pyobs-archive image archive (see **Section 4.1**). Currently only writing is permitted, i.e., uploading an image to the archive.
- `HttpFile` represents a file on a HTTP server, e.g., the pyobs file cache (see **Section 3.4.4**).
- `LocalFile` is a local file on the machine the module is running on.
- `MemoryFile` stores a file in memory.
- `SMBFile` allows access to a file on a Windows share without mounting it.
- `SSHFile` accesses a file on a remote machine that is accessible via SSH.
- `TempFile` works on a temporary file that will be deleted after being closed.

A file opened via VFS almost works like a normal file-like object in *Python*, with the one difference that all its methods are `async`, so they need to be awaited. pyobs also offers some convenience functions for reading and writing FITS, YAML, and CSV files in the VFS.

Figure 3 shows some examples, how a VFS path maps to a real path with a given configuration.

2.5 Image Processors and Pipelines

With the `Image` class in `pyobs.images`, pyobs offers a class for reading and writing images that also has support for additional data like a good pixel mask, a star catalog and pixel uncertainties. It is a simple wrapper around the FITS functionality in astropy and is used within pyobs whenever images need to be passed along.

Building on this image class, pyobs has the concept of "image processors" (defined in `pyobs.images.processors`), which simply take an image, process it in some way, and then return it. Currently, these types of processors are available:

- An **astrometry** processor takes an existing catalog attached to the image and tries to plate-solve it (see also **Section 4.3**).
- The **detection** processors try to detect objects in the image and write a catalog.
- The processors in **exptime** try to estimate a good exposure time from an image, the one existing implementation is for star fields.
- In **offsets** are processors that calculate some kind of offsets, usually used for guiding and acquisition.
- The **photometry** processors perform photometry on the image (usually at positions determined using a detection step) and write/extend the catalog.
- There are some more **misc** processors that can, e.g., add a good pixel mask, calibrate the image, or bin it.

Since image processors take an image as input as well as returning one as output, they can easily be chained into an image pipeline. This is done by many modules for pre-processing images in some (fully customizable) way before working on them. Adding new image processors is easily done and provides a perfect way for handling images.

Pyobs also offers a full (offline) image pipeline (in `utils.pipeline.Night`) that is also based on image processors, permitting the fully automatic processing of a night's images.

2.6 Error Handling

Handling errors in a single program is sometimes difficult enough, but it can get rather complicated in a distributed

TABLE 1 | List of available modules in the core package and in external packages.

Core Package (pyobs.modules.)

Module	Package	Description
DummyCamera	camera	Dummy camera for testing
DummySpectrograph	camera	Dummy spectrograph for testing
FlatField	flatfield	Taking a flat-field series
FlatFieldPointing	flatfield	Pointing for flat-fields
FlatFieldScheduler	flatfield	Scheduler for flat-fields
FocusModel	focus	Temperature model for focus
FocusSeries	focus	Auto-focus series
ImageWatcher	image	Watch directory for new images
ImageWriter	image	Write new images to disk
Seeing	image	Measure seeing in images
AutoGuiding	pointing	Auto-guiding with external camera
Acquisition	pointing	Fine acquisition
ScienceFrameGuiding	pointing	Auto-guiding with science camera
DummyAcquisition	pointing	Dummy acquisition for testing
DummyGuiding	pointing	Dummy guiding for testing
Mastermind	robotic	Main robotic module
PointingSeries	robotic	Automated pointing series
Scheduler	robotic	Task scheduler
DummyRoof	roof	Dummy roof for testing
DummyTelescope	telescope	Dummy telescope for testing
AutonomousWarning	utils	Acoustic warning in robotic mode
HttpFileCache	utils	File cache
Kiosk	utils	Take images and publish on website
Telegram	utils	Telegram interface
Trigger	utils	Event trigger
Weather	weather	Connection to pyobs-weather

External packages

Module	Package	Description
AlpacaTelescope	pyobs_alpaca	Telescope connected via ASCOM Alpaca
AlpacaFocuser	pyobs_alpaca	Focus unit connected via ASCOM Alpaca
AlpacaDome	pyobs_alpaca	Dome connected via ASCOM Alpaca
AravisCamera	pyobs_aravis	Aravis network cameras
AsiCamera	pyobs_asi	ZWO ASI cameras
AsiCoolCamera	pyobs_asi	ZWO ASI cameras with active cooling
FliCamera	pyobs_fli	FLI cameras
GUI	pyobs_gui	Graphical user interface for remote access
Pilar	pyobs_pilar	Pilar telescopes
SbigCamera	pyobs_sbig	SBIG cameras
SbigFilterCamera	pyobs_sbig	SBIG cameras with filter wheel
Sbig6303eCamera	pyobs_sbig	SBIG 6303e

system like pyobs. The basic requirement for every module is that it should handle errors on its own as well as possible (e.g., resolve errors states in hardware devices) but sometimes a calling module needs to be informed about a problem, e.g., if a camera does not respond to requests anymore.

Since error handling can be very specific to the problem at hand, pyobs only provides a framework for dealing with this, not a final solution. It introduces its own set of exceptions that are all derived from `PyObsError` in `pyobs.utils.exceptions`, and new exceptions can easily be added if required.

A module can call `register_exception()` and define a callback that is called whenever a given exception is raised and a given condition is met: the function accepts a limit of how often this can happen (optionally in a given time span) before the problem is escalated. In that case, the raised exception is changed into a `SevereError`, keeping the original exception as an attribute. That means, catching one of these severe errors means that an error has occurred too often (in a given time span).

This gets more interesting in a real pyobs system with several modules. There are some cases, in which a module should stop working at all and inform other modules about this. So, e.g., the `BaseCamera`, which is the base class for all cameras in pyobs, registers an exception like this:

```
register_exception(
  GrabImageError,
  3,
  timespan = 600,
  callback = self._default_remote_error_
  callback
)
```

This defines that after three occurrences of the `GrabImageError` exception within 600 s, the given method should be called, which is a default implementation in `Module`. It simply logs the error and sets the module to an error state that prevents (almost) any of its methods to be invoked remotely. If another module tries to call methods anyway, it receives a `ModuleError`.

If a method is invoked remotely and an exception is raised, this exception is wrapped in a `RemoteError` with the original exception stored in an attribute. This is useful to register exceptions with the module parameter, which only registers an exception on a given remote module. For instance, the `FocusSeries` module uses this:

```
register_exception(
  RemoteError,
  3,
  timespan = 600,
  module = camera,
  callback = self._default_remote_error
  _callback
)
```

So, whenever the remote module `camera` raises too many exceptions, the `FocusSeries` module itself goes into error state, which can be cleared remotely by calling `reset_error()`— and of course might reappear when the exception is raised again.

Note that registering an exception always also registers parent exceptions. So if exception B is derived from `A`, all occurrences of `B` also count for the registered limits for `A`.

3 AVAILABLE MODULES

In general there are two types of modules coming with pyobs: those that control actual hardware and those that do not. While the latter are part of the core package, the former are outsourced to separate packages, since they will not be required by everyone and often need special drivers to be installed. All modules can be found on the central GitHub page.

For developing your own modules, please refer to the documentation or just have a look at the existing ones as examples. There is also a simulation available that can be used for first tests. Please see the documentation for details on how to set it up.

Table 1 lists all modules available in the core package and in external packages.

3.1 Cameras

pyobs knows two kinds of cameras: classic cameras (derived from the interface `ICamera`), for which one actually starts and stops an exposure, and webcam-like cameras (interface `IVideo`), which constantly provide a video (or a series of images) as output. In addition, spectrographs are also supported (interface `ISpectrograph`), which output a spectrum instead of an image–therefore, most spectrographs would be implemented as a camera, since they return an image, from which the spectrum needs to be extracted.

In the following those camera types are listed, for which stable modules exist and are available via GitHub and PyPi. In addition, we also have modules for Andor and QHYCCD cameras, as well as normal USB webcams (via Video4Linux2), but they are all not in a publishable state. If you need one of those, please contact the author of this paper.

3.1.1 SBIG

The pyobs-sbig package builds on the SbigDevKit Linux driver for SBIG cameras. It is based on a Cython wrapper around that library's `CSBIGCam` and `CSBIGImg` classes. The different modules support SBIG cameras with and without filter wheel. There is a additional implementation for the STXL-6303E, due to its different gain at different binnings. Note that this special treatment of single models might be necessary for other cameras. The module has been tested on STXL-6303E, STF-402M, and STF-8300M cameras.

3.1.2 Finger Lakes Instrumentation

A Cython wrapper around the official `libfli`[8] library for FLI cameras is the core of the pyobs-fli. The module has been tested on a FLI ProLine 230.

3.1.3 ZWO ASI

pyobs-asi is a thin wrapper around the zwoasi package to support the cameras by ZWO ASI. It has been tested on a ZWO ASI071MC Pro.

3.1.4 Aravis

Aravis[9] is a library for Genicam cameras connected via gigabit ethernet or USB3. The module in pyobs-aravis uses a modified version of the python-aravis package for communicating with the cameras. It has been tested with several cameras from The Imaging Source[10].

3.2 Other Hardware

While astronomical cameras are often bought off the shelf and a few brands are most common between observatories, this is mostly quite different for the other hardware in the dome–and the dome itself. Those devices are often operated by custom controllers and need special treatment. However, if a driver of any kind exists, it is very simple to write a wrapper for it to be used within a pyobs system.

An attempt to standardize the communication between all kinds of devices has been made with ASCOM. A pyobs module for ASCOM will be described in detail below. Another of those attempts is INDI, for which we do not have a pyobs wrapper yet. Interfaces to those two standards are an easy way to add hardware to a pyobs system, for which ASCOM/INDI drivers already exist.

[8]https://www.flicamera.com/downloads/FLI_SDK_Documentation.pdf
[9]https://github.com/AravisProject/aravis
[10]https://www.theimagingsource.de

3.2.1 ASCOM

ASCOM is a standard for communicating with astronomical devices in Windows and is supported by a wide range of cameras, telescopes, domes, etc. Furthermore, there a many client applications like "The Sky" or "Stellarium" that can operate an ASCOM based system.

While pyobs can run on Windows, we made the experience that some things are a little more prone to error on that operating system–pyobs processes sometimes quit without warning. There is a (private) pyobs package for calling ASCOM interfaces directly on Windows, but due to these problems, we never published it. However, we can provide access on request.

The restriction to Windows systems is due to the use of Windows COM as means for communication, which is not available for other operating systems. Luckily, in 2018 ASCOM presented a new interface, called Alpaca, which is based on HTTP REST requests, and therefore can also be accessed from Unix-like systems. The pyobs-alpaca package provides modules for telescopes, domes and focus units via Alpaca. However, in contrast to most other modules in the pyobs ecosystem, these ones are not meant to be used directly, but more as some kind of inspiration for an observatory specific implementation. They are not a general implementation of the ASCOM protocol, but tailored specifically for the use case of the 50 cm Cassegrain telescope based at the Institute for Astrophysics and Geophysics in Göttingen.

3.2.2 Pilar

Pilar is a telescope control software from "4pi Systeme"[11] based on the Open Telescope Software Interface (OpenTSI), and currently used by our MONET telescopes via the pyobs-pilar package. While the specific implementation of this module might not be of interest for most observatories, it shows an example for a socket based communication protocol wrapped in a pyobs module.

3.3 Automating

While the modules described so far are all built around a specific piece of hardware, there are also those that purely consist of software to automate the boring stuff.

3.3.1 Auto-Focus

A common problem in astronomy is focusing the image on the camera sensor. In most cases this will be done by moving either a mirror (mostly the secondary) or the camera back and forth until stars appear sharp, i.e., with the smallest possible width. The `AutoFocusSeries` (in `pyobs.modules.focus`) module accomplishes this by taking a series of images at different focus values (i.e., position of M2 or camera), and tries to find an optimal focus by fitting a hyperbola through the estimated star widths in each image as a function of focus value. For this, references to a camera and a focus unit must be specified so that they can be controlled remotely.

The estimation of star sizes is fully configurable by injecting a class implementing the `FocusSeries` (in `pyobs.utils.focusseries`) interface. Our current default implementation is defined in `ProjectionFocusSeries`, which projects the image along its x and y axis, respectively, and measures moments on the resulting 1D data. Another possibility is to use a method for star detection/photometry for estimating star widths, as used in `PhotometryFocusSeries`.

While especially smaller telescope will typically work well with a constant focus value throughout the whole night, larger telescopes (with a steel structure) are constantly changing their size (and therefore the position of the perfect focus) due to temperature changes. For these cases, pyobs provides a temperature model for the focus, which is implemented in the `modules.focus.FocusModel` module and can adjust the focus continuously throughout the night. The configuration needs to specify a model function like this:

```
model: -0.043*T1 - 0.03*T2 + 0.06*temp + 41.69
```

While the value for `temp` is automatically fetched from a given weather module (see **Section 3.4.1**), those for `T1` and `T2` must also be specified in the configuration. In this case they are mirror temperatures and are supposed to be requested from the telescope module:

```
temperatures:
  T1:
    module: telescope
    sensor: T1
  T2:
    module: telescope
    sensor: T2
```

For this to work, a module named `telescope` must exist and its `get_temperatures` method must return values for `T1` and `T2`. With these values the module now calculates new focus values at a given interval and sets them accordingly.

An `AutoFocusSeries` also sends an event, when it has successfully determined a new focus, which can be handled by the `FocusModel` automatically to optimize its temperature model. For that to work, the model function must be defined with variables that can be fitted:

```
model: a*T1 + b*T2 + c*temp + d
```

In this case, a set of default values must also be provided:

```
coefficients:
  a: -0.043
  b: -0.031
  c: 0.062
  d: 41.694
```

If this is set up correctly, a fully robotic system can perform multiple focus series during each night (e.g., if there is nothing else to do) and automatically optimize the focus temperature model over time.

[11]http://www.sonobs.de/company/company.html

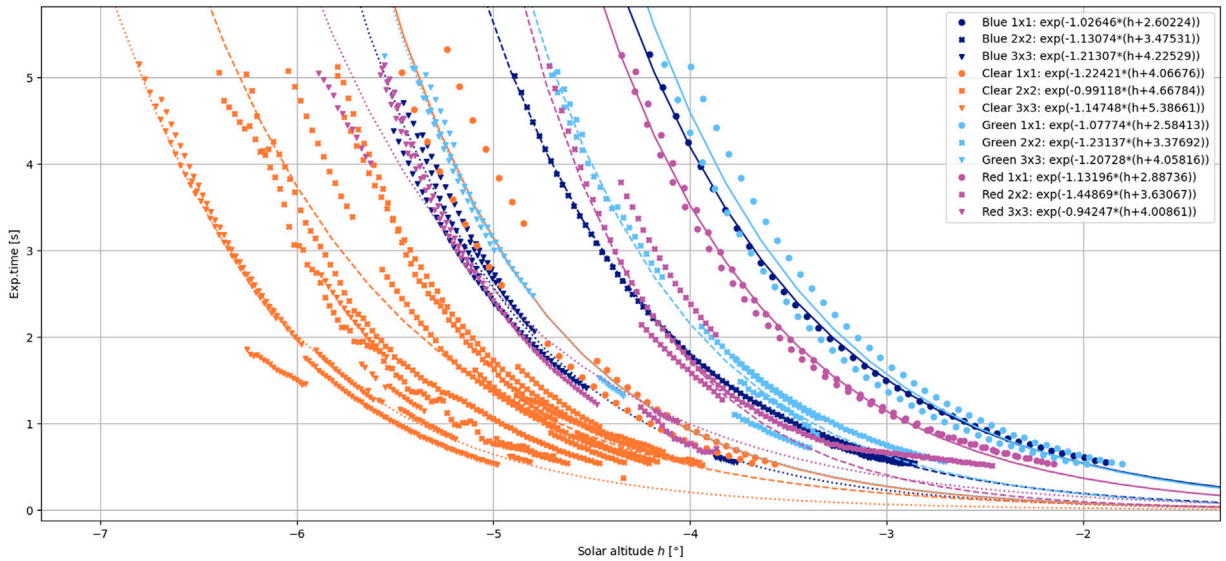


FIGURE 4 | An example for empirical models for flatfield exposure times. The points are optimal exposure times for getting a mean flux of 30,000 counts in the image as a function of solar altitude. The colors indicate different filters and binnings. A fit with an exponential function was performed and the best coefficients are given in the legend and plotted as lines.

3.3.2 Flat-Fielding

A task that is prone to be automated as early as possible is flat-fielding. While this is quite simple in a controlled environment with a closed dome, e.g., with a flat-field screen, it becomes more challenging when done on-sky during twilight. In that case, exposure times have to be adjusted continuously to obtain optimal count rates on the images.

To perform this task in a fully automatic way, it is best to first measure optimal exposure times as a function of solar altitude. For taking flat-fields, we always point the telescope at the same sweet spot on the sky, right opposite the Sun at an altitude of 80° (see Chromey and Hasselbacher, 1996). That way, we get comparable count rates for a given solar altitude and exposure time. We take a series of flat-fields, for which we try to get a constant flux level–in our case 30, 000 counts –, and calculate the optimal exposure time that would be required to get exactly the given level. **Figure 4** shows this for a set of RGBC filters and three different binnings as measured at the 50 cm Cassegrain telescope based at the Institute for Astrophysics and Geophysics in Göttingen.

As one can see, the measured points do not overlap perfectly over several nights, which can be caused, e.g., by clouds. But the data is good enough to fit exponential functions to it (see lines in plot), which we can use to roughly estimate the optimal exposure time for a given solar altitude.

During dusk twilight, a flat-field module picks a filter and binning combination and estimates the exposure time t for the current solar altitude. If the time is shorter than a given minimum $T_{\min}$, it does nothing and waits. When t reaches $0.5 \cdot T_{\min}$, test exposures are started, actually measuring the counts in the image, and calculate a new best exposure time. Only when $t \geq T_{\min}$ the module starts taking actual flat-fields until either a given number of images has been taken or t gets larger than a given maximum $T_{\max}$. In dawn twilight, the procedure can be performed accordingly with $T_{\min}$ and $T_{\max}$ swapped and opposite comparisons. This is implemented in the `FlatField` module in `modules.flatfield`.

The class handling the actual flat-fielding is, again, fully configurable. An example for the `flat_fielder` parameter of the module might look like this, defining functions for the exposure time for different binnings and filters:

```
class: pyobs.utils.skyflats.FlatFielder
pointing:
  class: pyobs.utils.skyflats.pointing.
  SkyFlatsStaticPointing
combine_binnings: False
functions:
  1 × 1:
    Clear: exp(-1.22421*(h+4.06676))
  2 × 2:
    Clear: exp(-0.99118*(h+4.66784))
  3 × 3:
    Clear: exp(-1.14748*(h+5.38661))
```

The given class for `pointing` can also be used in the `FlatFieldPointing` module, which only points the telescope to a specific position without taking flat-fields. This can be useful, if multiple instruments are supposed to be flat-fielded at the same time.

A twilight is usually long enough for taking flat-fields in more than one filter/binning combination. The `FlatFieldScheduler` module provides a way to run multiple ones as long as the twilight lasts. It can also read

priorities from a customizable source, which can be, e.g., an image archive, so that the priorities are the larger the longer ago the last flat-fields in this combination were taken. The most simple solution, though, is not covered by this scheduler: just a list of filter/binning combinations, which are flat-fielded in the order as they are given. However, the implementation would be so simple that we can leave it as an exercise to the reader.

3.3.3 Acquisition

After moving a telescope to a target, it is often off by some arcseconds or even arcminutes. Sometimes this is unacceptable, especially when the light of a star, e.g., needs to be coupled into a small fiber. In those cases, a fine acquisition based on images from some camera is required. The `Acquisition` module (in `modules.pointing`) takes images, runs them through a pipeline (see **Section 2.5**) to determine what offset to move the telescope, and then applies this offset. This is repeated until the offset is smaller than a given limit.

The configuration for the pipeline typically consists of three steps:

```
pipeline:
  - class: pyobs.images.processors.detection.
    SepSourceDetection
  - class:pyobs.images.processors.astrometry.
    AstrometryDotNet
    url: https://astrometry.example.com/
    radius: 5
  - class: pyobs.images.processors.offsets.
    AstrometryOffsets
```

First, a source detection is run on the images, followed by an attempt to plate-solve it using the service of Astrometry.net (Lang et al., 2010), for which we provide a self-hosted solution (see **Section 4.3**). In the last step, the found coordinates are compared to those from the pointing, and an offset is calculated. Alternative methods are possible by simply changing the pipeline. For instance, an image processor could find the brightest star in the image and set the offset to move the telescope there.

Applying the offset to the telescope is also fully configurable. For a telescope that accepts RA/Dec offsets, it might look like this:

```
apply:
  class: pyobs.utils.offsets.Apply
  RaDecOffsets
  max_offset: 3,600
```

The given class simply takes the offsets from the image (written by an image processor) and moves the telescope accordingly.

3.3.4 Auto-Guiding

The task of auto-guiding is quite similar to that of acquisition, so the configuration is as well: it also mainly consists of a `pipeline` and an `apply` step. But instead of running until the calculated offset is small enough, the auto-guiding runs forever, or until stopped, to correct for any shift in pointing that the telescope is doing over time.

In pyobs, two kinds of auto-guiding are ready to use:

- **Science-frame auto-guiding** (module `ScienceFrameAutoGuiding`) uses the images of the science camera for guiding. This works quite well if the exposure time is small enough to correct for any shifts of the telescope over time.
- In contrast, what we just call **auto-guiding** (module `AutoGuiding`) requires an extra camera that is mounted, e.g., at the same focal plane as the science camera or at an extra guiding telescope that moves along with the main telescope. In this case, with a bright enough star in the field, the auto-guiding can perform its corrections, independently of the actual science taken, in intervals as short as required.

While the astrometric method used in the acquisition would also work for auto-guiding, it is usually too slow. These alternative methods are provided with pyobs:

- A projection method as implemented by `ProjectedOffsets` projects the images separately along x and y axis and cross-correlates both individually with a reference image. The resulting x/y pixel offset can be translated into a RA/Dec or Alt/Az offset.
- Cross-correlating full images is usually too slow, so `NStarOffsets` uses star positions from a source detection that needs to run before, and cross-correlates only small images around the N brightest stars in the image.

3.4 Utilities

A couple of smaller utility modules for common tasks are provided for convenience.

3.4.1 Weather

For fully autonomous observatories, the most important task is not to get observations done, but to close the roof on bad weather and to keep it closed–an expensive telescope and camera is worth nothing if regularly rained on. With pyobs-weather (see **Section 4.2**) there is an affiliated project that acts as an aggregator for data from several weather stations and evaluates some logic to determine, whether the weather is good or bad, i.e., suitable for observations or not.

The `Weather` module connects to an instance of pyobs-weather and can provide several functions within a pyobs network:

- It provides FITS header entries with weather information for science data.
- It has a simple `is_weather_good()` method returning a Boolean, indicating whether the weather is good or not.
- It sends events when the weather status changes, `GoodWeatherEvent` and `BadWeatherEvent`, which other modules can handle and react accordingly.

FIGURE 5 | A screenshot of the graphical user interface (GUI) as provided by pyobs-gui. It shows the list of connected modules that are supported by the GUI on the left. When selecting one, a custom widget for each kind of module is shown in the main area right of it. Below is the logging area, which shows log entries from all connected modules.

Note that the safety net cast by this module is mainly for the robotic system to react on changes. It is not a replacement for an emergency shutdown in case of, e.g., rain, which should work even without network.

3.4.2 Telegram

Even the best logging is only good, if someone reads it. Therefore, the module `Telegram` can forward all messages of a given level (info, warning, error, . . .) into a Telegram[12] chat – the default configuration would be to have only error messages sent. That way the telescope administrator usually gets notified of a problem within seconds.

The Telegram bot used for this provides several commands that can be issued to it by simply opening a chat on the smart phone. For security reasons, every user has to login (`/login` command) before receiving any logs and before being able to issue any other command./`loglevel` changes the current log level and/`modules` lists all online modules. The most powerful command is/`exec`, which allows the user to issue any pyobs command to any module, similar to what is possible within a module or in the Shell of the GUI (see **Section 3.5**). Using this, the administrator can easily shut the roof or abort an observation from within a Telegram chat.

[12]https://telegram.org

While Telegram currently is the only supported chat system, adding other ones should be as simple, as long as an API is provided that can be used by pyobs.

3.4.3 Trigger

Events are a powerful system in pyobs and for some of them a default action should be performed every time they are encountered. Instead of writing a new module for this, one can simply use the existing `Trigger` module. It defines events and the method on a given module that should be executed, when the event is triggered. For example:

```
triggers:
  - event: pyobs.events.GoodWeatherEvent
    module: dome
    method: init
  - event: pyobs.events.RoofOpenedEvent
    module: telescope
    method: init
```

This configuration calls `dome.init()` on a `GoodWeatherEvent` and `telescope.init()` on a `RoofOpenedEvent`, thus opening roof and telescope when the weather changes from bad to good–which, in case of pyobs-weather, is usually also the case after sunset for a night telescope. Note that there is no trigger configuration for

the bad weather case, since all modules handle that on their own.

3.4.4 FileCache

While a camera module can be configured to store its files locally, that can be quite impractical, if it runs on a different computer than the rest of the pyobs system, which might be the case quite often. So there is need for a place to store the images that can be accessed from all modules–or at least those that need access to the images.

A network mount using, e.g., SMB or NFS does the job well, but with `HttpFileCache` there is also a module available for that in pyobs. It opens a web server on a given port, which can be used to upload images from the camera and download them somewhere else. It can simply be accessed via the VFS (see **Section 2.4**) using a `HttpFile` root.

3.4.5 ImageWriter and ImageWatcher

When the camera uploads its images to a FileCache (see above), they should still be stored somewhere, since the cache only holds a limited amount of files. An easy way to do that is the `ImageWriter` module that waits for `NewImageEvents`, downloads those images and stores them at a different VFS location.

To make this a little safer and reduce the risk of losing images, an `ImageWriter` should always write images to a local disk. If they are supposed to be copied to a remote location, the preferred way is an additional `ImageWatcher`, which watches a given path for new files, copies the files somewhere else, and only deletes the original files if there was no error. So a typical setup would configure the `ImageWriter` to store its files into a local directory like this, assuming that the camera stores its images at `/cache/` and `/some/temp/dir/` is some local temp directory:

```
class: pyobs.modules.image.ImageWriter
vfs:
  class: pyobs.vfs.VirtualFileSystem
  roots:
    cache:
      class: pyobs.vfs.HttpFile
      download: http://somewhere:37075/
    archive:
      class: pyobs.vfs.LocalFile
      root:/some/temp/dir/
```

Note that the root `archive` is used since the default value for the `filenames` parameter of the module is`/archive/FNAME`.

After the images have been stored locally, an `ImageWatcher` should pick them up and copy them into an archive (note that curly brackets in `destinations` indicate placeholders which are filled from FITS header values):

```
class: pyobs_iagvt.filewatcher.FileWatcher
watchpath:/temp/
destinations:
  -/archive/{FNAME}
vfs:
  class: pyobs.vfs.VirtualFileSystem
  roots:
    temp:
      class: pyobs.vfs.LocalFile
      root:/some/temp/dir/
    archive:
      class: pyobs.vfs.ArchiveFile
      url: https://archive.example.com/
```

After the images have been copied into the archive, they will be delete from the temp directory by the `ImageWatcher`. Currently the copied files are not validated in order to make sure that they are identical to the original, but this would be a simple feature to add.

3.5 Graphical User Interface

While all the other modules presented here are fully autonomous, pyobs also provides a graphical user interface (GUI) for easy (remote) access to the system. Technically it is also just another module, which opens a window for interaction with the user.

Figure 5 shows a screenshot of the GUI right after a bias image has been taken with the selected SBIG camera. The main window of the GUI consists of three major parts:

- The **list of module pages** on the left, including the three special pages Shell, Events, and Status.
- The **system log** on the bottom, showing all log entries from all connected modules as well as a list of all those modules on the lower right.
- The **module page**, filling the rest of the window, which changes depending on the selected module.

The three special pages mentioned above are:

- The **Shell** is an interactive command prompt, in which the user can execute any command on any module in the form `<module>.<method>(<params>)`. This makes the shell a very powerful tool for admins and for debugging.
- The **Events** page shows a chronological list of all events that have been sent in the pyobs network. It also allows to send events on its own with parameters defined by the user.
- The **Status** page shows the current status of a module, e.g., whether it is in an error state. It also shows the pyobs version of every module to keep track of updates.

In the list of modules on the left, not all modules are listed, but only those for which a graphical user interface has been designed. The GUI is fully dynamic, which means that it changes according to the list of connected modules. Single module pages also adapt to the capabilities of the associated module, e.g., the camera page only shows options for window and binning, if the camera supports it.

The customization of the GUI goes even further with user-defined pages. For example, pyobs does not provide a user

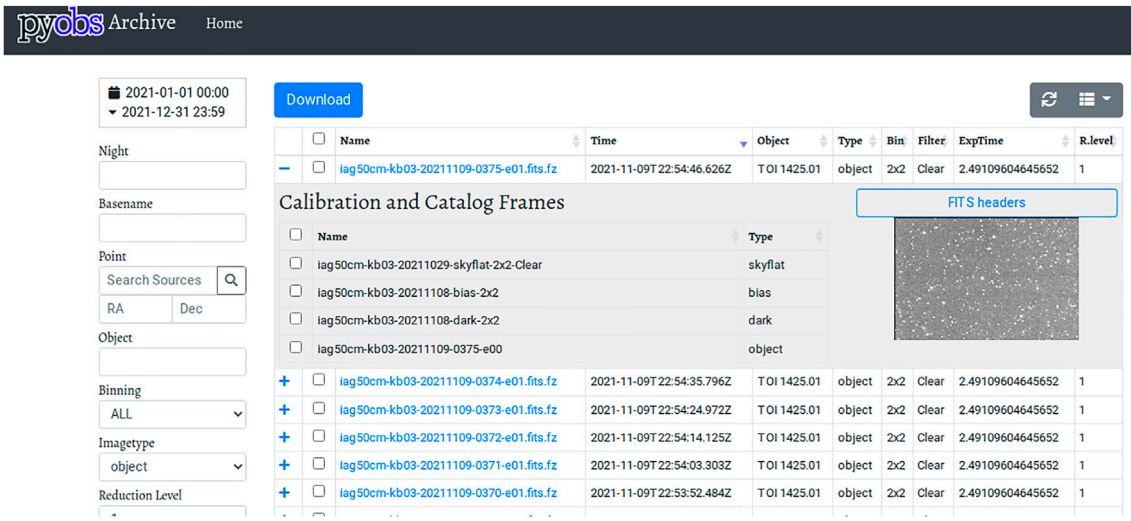


FIGURE 6 | Screenshot of the pyobs-archive instance that we use for the MONET telescopes and the IAG50 cm telescope.

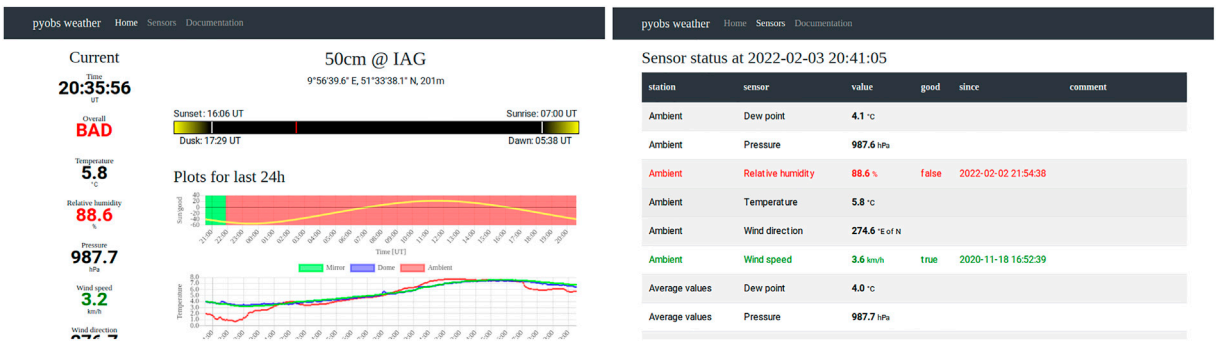


FIGURE 7 | Two screenshots from the pyobs-weather web frontend for the IAG 50 cm telescope. The main page is shown on the left and the Sensors page on the right.

interface for acquisition and guiding, but in the case of our solar telescope, a visual feedback is important. So we created a new widget and defined it in the configuration of the GUI:

```
widgets:
- module: guiding
  overwrite: True
  widget:
    class: pyobs_iagvt.guidingwidget.
    GuidingWidget
    acquisition: acquisition
```

This tells the GUI to overwrite an existing widget for the `guiding` module with the given class. Using custom widgets, one can adapt the GUI to work with any special requirements.

The other way around, restricting access in the GUI, can also be accomplished in the configuration via the `show_shell`, `show_events`, and `show_status` parameters, which, if set to False, hide the corresponding page. An explicit list of allowed module pages can be provided with the `show_modules` parameter. Here is an example for a very limited access to the camera only:

```
show_shell: False
show_events: False
show_status: False
show_modules: [camera]
```

Altogether, the GUI tries to allow access to all modules as well as it can, but it is also highly customizable to match any requirements of an observatory. With ports for the XMPP server (and probably the file cache) open to the public, this enables a safe and easy remote access to the pyobs system.

4 AFFILIATED PROJECTS

There are a few projects with "pyobs" in their name that do not provide any new modules but some external services that are essential for operating a fully-autonomous telescope.

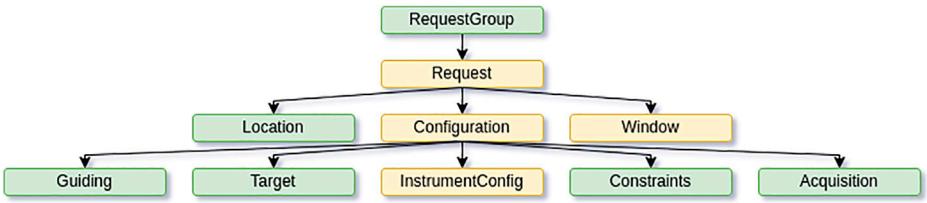


FIGURE 8 | Structure of a task in the LCO portal. While the green fields can occur only once, there can be multiple entries for the yellow fields (Request, Configuration, Window, InstrumentConfig).

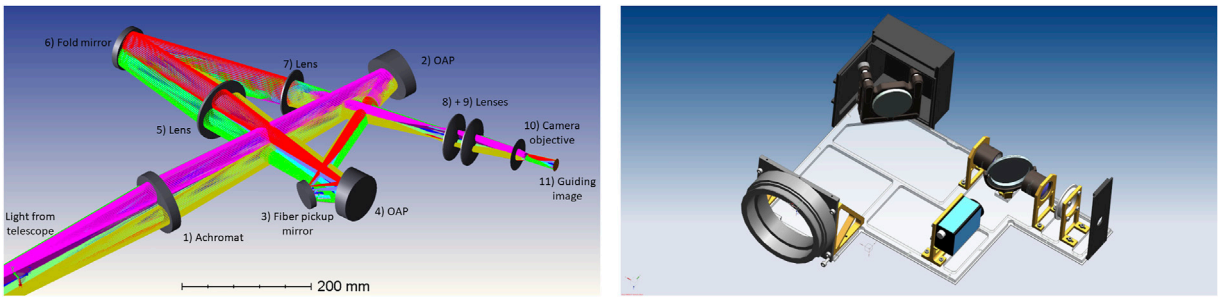


FIGURE 9 | Left: Mid-resolution resolved Sun fiber setup for the VVT: The full image of the Sun is re-imaged onto the fiber pickup mirror that is hosting a 525 μm fiber (corresponding to a 32 arcsec field of view). The fiber leads to a Fourier-Transform-spectrograph. Behind the pickup mirror the light is again re-imaged, this time onto the guiding camera which is used by pyobs for both pointing and guiding. Right: CAD-model of the fiber-guiding unit for the 50 cm telescope. Starlight is re-imaged onto a fiber-pickup mirror and the remaining light is redirected into the guiding camera, allowing for nearby stars to act as guidestars.

4.1 Image Archive

In classic astronomy an observation consists of three steps:

1) Planning an observation, i.e., finding targets, defining filters and exposure times, evaluating best times for the observation, etc.
2) Actually performing the observation at the telescope.
3) Calibrating and analyzing the data.

Nowadays it is absolutely possible to automate all three and avoid human interaction at all. While this topic goes far beyond the scope of this paper, we want to mention that a full automation does not only work for large surveys, but also for small telescopes in the middle of a town like Göttingen (see Masur et al., in prep). However, for robotic observations at least the second step falls away from the observer's responsibility, but also parts of step one (defining observing times) and three (calibrate data). In that case, probably not knowing exactly when an observation was taken, an efficient way to find data becomes more important.

This is where an image archive comes into play. There is the LCO science archive[13],[14] to use, but it stores the images in Amazon AWS S3, while we wanted to store data locally. So we developed our own backend, which also supports the LCO API, and took parts of the LCO web frontend with permission and adapted it to our needs. We also added a HTTP endpoint for uploading images. Within pyobs there are classes for both an easy upload using the VFS (via `ArchiveFile`), and a full wrapper for accessing the archive in `PyobsArchive`.

Figure 6 shows a screenshot of the archive that we use for the two MONET telescopes and the IAG50 cm telescope. On the left, there is a list of options to filter the data by. On the right is the list of images matching the selected criteria. More details–including connected data (for calibrated images), a link to the FITS headers and a thumbnail preview–can be accessed by clicking on the plus symbol. Single or multiple images can also easily be downloaded using this web frontend.

4.2 Weather Aggregator

In **Section 3.4.1** we already mentioned a project for aggregating data from different weather stations and evaluate the values in order to determine, whether the weather is good for observing. **Figure 7** shows two screenshots from pyobs-weather as used by the IAG50 cm telescope.

On the left, the main page is shown, with average values from all sensor types as well as plots for the current night (top), weather status (green and red shaded areas in the plot below, indicating good or bad) and solar altitude (yellow line in same plot), and plots for all sensors, grouped by type (i.e., temperature, humidity, etc). On the right, the Sensors page is shown with current values for all sensors from all stations, times of last changes (for evaluated sensors, see below) and comments on the current status. There is a public API for the weather data, which can easily be accessed via the weather module (see **Section 3.4.1**).

The system is fully customizable. The basic unit in pyobs-weather is a *station*, which usually defines a single physical weather station. There are some station classes already present, of which some are more generic (getting data from a MySQL or CSV table) and some are

[13]https://archive.lco.global/

[14]https://github.com/observatorycontrolsystem/science-archive

specific to the observatories where our telescopes are located (e.g., for the weather station on Mt. Locke at McDonald Observatory). Additional classes for more stations can easily be added. There are three special stations that do not represent an actual weather station: `Current` contains the current average values from all stations, while `Average` keeps a 5-min average. Finally, `Observer` calculates current conditions, which, at the moment, is only the solar altitude.

Every station then contains one or more *sensors*, which provide values for a sensor of a given type: temperature, relative humidity, pressure, wind speed and direction, particle count, rain, sky temperatures, or solar altitude. To each sensor, one or more *evaluators* can be attached, which take the current value and decide whether it allows for observations or not. Currently, pyobs-weather offers four different kinds of evaluators:

- A `Boolean` is a simple logic evaluator, which is `True` if the sensor value is `True`, and vice versa–or the opposite, if `invert` is set to `True`.
- `Switch` is a simple switch, which is `True`, if the sensor value is above a given `threshold`, and vice versa. And, again, the other way around, if `invert` is set to `True`.
- A `Schmitt` trigger is similar to a `Switch`, but it takes two values: for it to become `True`, the sensor value must be below a given `good` value, but to become `False` again, it must rise above a given `bad` value.
- Sensor values have a valid flag, which is mostly used (and set to False), if the value is older than 5 minutes. The Valid evaluator only evaluates to True, if the value is valid.

As an example, we assume getting relative humidities from two weather stations. For both we would typically set a `Schmitt` evaluator with values like `good` = 80 and `bad` = 85, which means that the weather is marked as bad, if the humidity rises above 85%, but is only marked good again, if the humidity falls below 80%. It would not be a good idea to attach a `Valid` evaluator to both, since weather stations can break. However, we still always want a valid reading for the humidity, so we assign it to the humidity sensor in the `Current` station. That way, if we get no valid value at all, the weather is marked as bad. Evaluators on the `Average` station are never evaluated, but they are used for color coding the plots, i.e., mark areas that would mean bad weather.

Every sensor can also have a delay before switching from good to bad or vice versa. This can be used so that, e.g., the rain sensor only reports good weather if the last rain was at least an hour ago. Or, the other way around, a sensor that tends to flapping, i.e. wrongly reports bad weather for a short time before going back to normal, could be set to switch to bad only if this condition lasts for a given time.

4.3 Astrometry

Getting astrometric solutions for images (i.e., "plate-solving" them) is a task required at multiple occasions (see, e.g., **Sections 2.5 and 3.3.3**). For this we use a self-hosted version of Astrometry.net (Lang et al., 2010), adding a HTTP interface for accessing its service. Similar to Astrometry. net's own web service, [15] it accepts a list of X/Y positions of stars on an image, but in addition some parameters for the fit can be provided, like a first guess for the coordinates and an estimate for the plate-scale. A successful call returns FITS header entries that can be added to an existing file in order to get a valid world coordinate system (WCS). The whole process usually takes well below one second. pyobs provides an image processor that uses this web service for easy use in a pipeline (see **Section 2.5**).

[15]http://nova.astrometry.net

5 FULL ROBOTIC MODE

With everything described so far, we already have a working observatory. We can control all devices, automate some things, and remotely control the system with the GUI. All that is needed for a fully autonomous telescope is some piece of software that coordinates everything. These robotic systems come in all shapes and colors: from a rather simple survey mode, in which a pre-defined list of targets is executed from top to bottom, probably all to be done with the same settings, whenever the conditions are right, to a system with user-defined tasks, maybe multiple instruments, and a scheduler that tries to fit all together.

5.1 Scheduling

The most simple robotic system imaginable is a simple list of targets that are to be observed one after the other, top to bottom. An algorithm for that might look like this:

1. Select a target from a list, probably the first one.
2. Move the telescope to the given coordinates.
3. Take an image and store it.
4. Repeat.

A system like this still has some other things to take care of, e.g., open up at dusk (for night observations) and close down at dawn–or when the weather gets bad. Any interruption (like daylight or rain) would just delay the selection of the next target. While very simple, this kind of system is suitable for many types of observations. There is no module implementing a survey mode in pyobs, but it could easily be added with very few lines of new code, specialized on the use case at hand.

This "survey mode" is also easily extendable, e.g., add an exposure time and a filter to the table of targets and set them before starting the exposure. However, the targets would still be observed in the order that they appear in the table. Therefore, the next step might be to filter the table of targets by visibility and sort it by some kind of priority. If we do that every time the system is idle, we get some kind of "just-in-time" (JIT) scheduler, always picking the next target when needed, but never planning further ahead. Some control systems, like the one for STELLA on Tenerife (Granzer et al., 2012), developed this idea further and have been using it successfully for years. A JIT scheduler can be very powerful, because it can easily adapt to changing observing conditions like seeing or transparency and picks its next target accordingly.

There is, however, one major disadvantage for these kind of systems: selecting only the next targets means there is no full plan for the night (or day), so it may be difficult to impossible to predict, whether a specific target will be observed or not. It may even be difficult to decide, which parameters need to be changed in order to

make sure the observation will take place. Furthermore, the selection of targets may never be "optimal", i.e., it is challenging to fill the observing time with the best possible targets. For example, take an object *A* that can be observed at the beginning and at the end of the night (maybe a transit event). Another object *B* can only be observed at the beginning of the night. Even if *A* has a higher priority, it might be better to observe *B* first and then *A* at the end of the night.

An astronomer, going on an observing run, would probably plan the nights in advance and make a schedule, when to observe which target. This is an optimizing problem and so we can call these kinds of schedules "optimal". This is a different approach to selecting targets and not as straightforward as the one for JIT schedulers described before. Luckily, there are free schedulers available for use, e.g., the adaptive scheduler developed by LCO[16] and the Astropy-affiliated project astroplan, just to name two. In principle, they all try to optimize the placement of observations for the whole night so that a given value is maximized, e.g., the total observing time or something like the time-integrated priorities of the tasks.

While the LCO scheduler can run fully independent from pyobs, there is a module based on astroplan: `Scheduler`. It takes schedulable tasks from a `TaskArchive` object, calculates a schedule, and writes it to a `TaskSchedule` object. A `TaskArchive` simply holds a list of tasks (in the form of `Task` objects) and returns them on request. The scheduler takes these tasks, converts them into astroplan's `ObservingBlocks`, applies given constraints, and starts the scheduler. The result is a time table, giving start and end times for all scheduled tasks, which is passed to the `TaskSchedule`, storing it to be accessed by the robotic telescope system.

All these three classes (`TaskArchive, TaskSchedule, and Task`) are abstract and need specific implementations for a method to store tasks and schedule. The implementation coming with pyobs is one tailored to be used with the LCO observing portal, but access to other task databases can easily be added.

Furthermore, this gives a simple framework for changing the used scheduler at a later time. The one implemented in astroplan is a "greedy" one, i.e., it schedules the task with the highest priority first, then the one with the next lower priority, and so on. While it ensures that the highest priority target is observed, this is not true for all other targets. Thus, the result of a "greedy" scheduler is still far away from an optimal one. However, changing the actual scheduler will not affect the rest of the robotic system at all.

5.2 LCO Observing Portal

The central part of the LCO observing portal is a database, mainly storing tasks, schedules, and observations, and a HTTP REST interface for accessing it–see details about the API on LCO's developers page[17].

When the portal is set up correctly and running, a new account must be created with "Staff" permissions to access all the necessary endpoints. The security token for this account must be provided in the configuration of `LcoTaskArchive` and `LcoTaskSchedule`, which are the LCO-specific implementations of the classes discussed above. They both make use of LcoTask that simply stores the JSON object returned from the portal. These classes are enough to run the scheduler in connection with an LCO portal.

Figure 8 shows the structure of a task in the LCO portal:

- The top-most element is a request group, which has a name and belongs to a proposal. It can contain one or more requests.
- A request contains a location (i.e., the telescope to use), one or more observation windows and one or more configurations.
- A configuration stores settings for acquisition and guiding, observing constraints (airmass, moon distance, etc.), the target information and one or more instrument configurations.
- Finally, an instrument configuration holds information like exposure time and count, and filter to use, all depending on the selected instrument.

Each of these elements also contains an `extra_params` field, which can be used for any extra information that is not supported by the default parameters.

Configurations have a `type` parameter, which will be important for running the task. The default value for an imaging camera would usually be `EXPOSE`, which just exposes as many images as given in the instrument configuration. Another possibility is `REPEAT_EXPOSE`, which loops all instrument configurations, until a given `repeat_duration` is reached. There are also other, more specific types, like `AUTO_FOCUS` for performing an auto-focus series.

5.3 Running Tasks

With the schedule in place, we actually need to observe the tasks. In pyobs this is done by a `TaskRunner`, which only has two methods: `can_run()` checks, whether a given task can run right now, and `run_task()` actually executes it. For this, pyobs uses the concept of "scripts", which can be fully customized in the `runner` section of a configuration for a task runner. While the following will concentrate on running tasks from an LCO portal, implementations for other task archives should be easily implemented.

In the case of the LCO portal, the script to use is defined by the configuration type. A possible configuration might look like this:

```
runner:
  class: pyobs.robotic.TaskRunner
  scripts:
    BIAS:
      class: pyobs.robotic.lco.scripts.
      LcoDefaultScript
      camera: sbig6303e
    EXPOSE:
      class: pyobs.robotic.lco.scripts.
      LcoDefaultScript
      telescope: telescope
      filters: sbig6303e
      camera: sbig6303e
      roof: dome
      acquisition: acquisition
      autoguider: autoguider
```

This uses the same script (`LcoDefaultScript` in `robotic.lco.scripts`) for two different configuration types.

[16]https://github.com/observatorycontrolsystem/adaptive_scheduler
[17]https://developers.lco.global

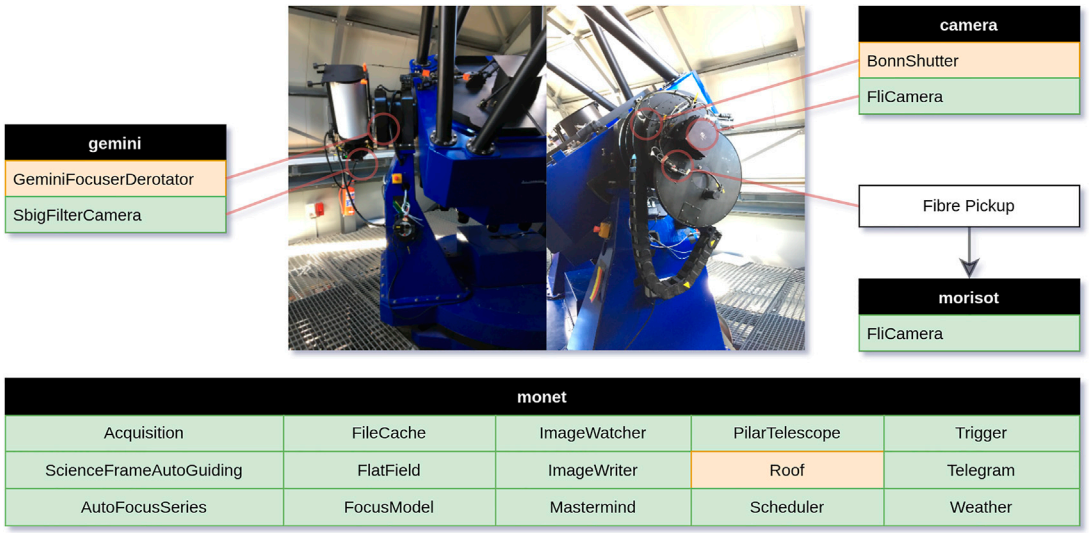


FIGURE 10 | List of all pyobs modules that are currently running at MONET/S and how they are distributed to four different computers. Marked in green are those modules that are available in pyobs-core or one of the additional packages. The three other modules are custom implementations for the given hardware.

The script checks internally, what kind of observation to perform. As usual, all parameters in the configuration are forwarded to the constructor of the given class–in this case all of them are names of modules handling specific tasks. Note that a request can contain multiple configurations, and each might use a different script and might be observable or not.

An `LcoTask` checks and runs all configurations in a request. The procedure that the `LcoDefaultScript` runs for a single configuration looks like this:

- If the configuration type is `EXPOSE`, which is a science observation, move the telescope to the given coordinates.
- If a fine acquisition is requested, do it.
- If guiding is requested, start it.
- Now loop all instrument configurations and for each set the filter and binning and take the given number of images.

If the configuration type is `BIAS` or `DARK`, this procedure would simplify to just taking images.

In the case of this `LcoDefaultScript`, the script needs to know about internals of the task that are not available via the public interface of `Task`, so an LCO specific script is required. The same is true for the auto-focus script `LcoAutoFocusScript`. On the other hand, there are some scripts that do not need any information from the task and can therefore be run in any system, e.g. the `SkyFlats` script. When using this in a LCO environment, the configuration type `SCRIPT` is required and a special script `LcoScript`, which evaluates a `script` parameter in the `extra_params` of the configuration to delegate execution to another script. It can be configured like this:

```
runner:
 class: pyobs.robotic.TaskRunner
 scripts:
  SCRIPT:
    class:
    pyobs.robotic.lco.scripts.LcoScript
  scripts:
    skyflats:
     class: pyobs.robotic.scripts.SkyFlats
     [...]
```

Now, whenever the configuration type is `SCRIPT` and the `script` is set to `skyflats`, the given script `SkyFlats` is executed. The whole script system is designed to be as flexible as possible and should allow for writing custom scripts for any requirement.

For our solar telescope we also use a (modified) LCO portal, but the robotic mode is a lot simpler: there is no scheduler, but the task archive just requests the schedulable blocks and returns the one with the highest priority. This is possible, because all positions on the solar disc are visible as soon as the Sun is up in the sky. There is also a different default script that just moves the telescope and triggers the spectrograph.

5.4 The Mastermind

Using the scripts system, building a central module that runs them becomes very simple–we call this module the "mastermind". It creates a `TaskSchedule` and a `TaskRunner` from its configuration and then continuously gets the tasks from the former and executes them via the latter. It also sends events when starting and finishing a task and writes information about the task into the FITS headers of the images.

The whole system is flexible enough that we run two 1.2 m and one 0.5 m night telescope with it, as well as a 0.5 m solar telescope–however, for the last one the default scripts are not used (they use, e.g., a different coordinate system) and even the LCO portal had to be adapted for this use case. Nevertheless, the changes were minimal and we can use the same code base for all telescopes.

The solar telescope is also a good example on how to customize the mastermind. Since all functionality is included by referencing *Python* classes in the configuration, the whole execution of a task can be changed, even when sticking with the LCO portal–with other task backends one needs to write own code anyway. It is completely possible to change the code to operate multiple instruments or even telescopes. For instance, we are currently adapting the system to calibrate three instruments at two telescopes simultaneously for MONET/S (see next Section).

6 TELESCOPES

The Institute for Astrophysics and Geophysics in Göttingen (IAG) operates four telescopes, of which two are located within the faculty building, one is in Texas, and the last one is in South Africa. In this section we will describe the hardware for each one and their level of automation with pyobs.

6.1 IAG 50 cm

The IAG 50 cm is a Cassegrain telescope located on the roof of the institute with a main mirror with 0.5 m diameter and a focus length of 5 m (f/10), housed in a classical rotating dome. The telescope is mainly used for educational purposes and public outreach. With the use of pyobs we now also use the rare days of good weather in Göttingen for science observations, but it is mainly a testing platform for the two MONET telescopes (see below). The main instrument is a SBIG STL-6303E with a pixel scale of 0.55 ″/px (with a focal reducer). Attached to the telescope is a smaller 110 mm f/7 refracting telescope with a ZWO ASI 071 MC camera.

Dome, telescope and focusing unit are running with ASCOM and are connected to pyobs via `pyobs-alpaca`. The two cameras use their respective modules (`pyobs-sbig` and `pyobs-asi`). The other modules that we use perform the following tasks (for all see **Section 3**):

- Fine acquisition with both of the cameras,
- auto-focus for the main telescope and camera,
- flat-fielding for both cameras,
- file cache and image writer and watcher,
- scheduler and mastermind for robotic mode,
- telegram bot,
- weather from a connected `pyobs-weather` (see **Section 4.2**) page.

The main telescope runs fully robotically with a copy of the LCO portal (shared with the MONET telescopes), while the smaller telescope is not yet supported in this mode. We are currently working on guiding with the small telescope and on implementing the necessary pointing model in pyobs. We are also currently adding a fiber pick-up to transfer the light from the main telescope to a spectrograph in the optical lab (see next section). For this, the guiding uses a camera from The Imaging Source pointed at the fiber pin hole in a 45° mirror, which has already been tested (see **Figure 9**, right).

6.2 IAG Vakuum-Vertikalteleskop

The Vakuum-Vertikalteleskop (VVT) consists of a siderostat on the top of the faculty building, redirecting the light two stories down into the building, where the 0.5 m primary mirror is reflecting the light back up one story and into the optical lab. It provides both a f/11 primary focus and a Gregory f/50 secondary output.

There are a total of six observing modes for the telescope, with five of them using pyobs for pointing and guiding using a custom module via an interface to its control system. These modes include different spatial resolved observing modes (with field of views between about 4 and 100 arcsec) and Sun-as-a-star integrated light modes. As an example, **Figure 9** (left) shows a Zemax raytracing of the mid-resolution resolved Sun fiber setup. The light from the primary mirror is collimated and re-imaged onto a fiber pickup mirror and re-imaged a second time onto the CCD guiding camera that is used for acquisition and guiding via detecting the solar disk.

The light entering the fiber is sent to our Fourier-Transform-spectrograph (FTS), a Bruker IFS 125HR with a maximum resolving power of $>700,000$ at 600 nm. For the FTS, another custom module is used for HTTP communication with a LabView instance, which in turn is connected to the instrument software OPUS. For more details on the resolved Sun setup–see Schäfer et al. (2020b), and for more details on the coupling into the FTS see Schäfer et al. (2020a). We are currently commissioning the fully robotic mode, based on a modified LCO portal, which now accepts coordinates in the Stonyhurst Heliographic system (HGS).

6.3 MONET/N

The two MONET Alt/Az telescopes (Hessman, 2001; Bischoff et al., 2006) have (almost) identical hardware with a 1.2 m main mirror at f/7. They were optimized for fast operations with up to 10°/s on both axes and therefore also have a clam-shell roof that opens completely. The northern telescope, MONET/N, is located at McDonald Observatory in Western Texas, United States the process of designing a fiber-fed high resolution spectrograph for high-precision radial velocity observations of G-type stars on the m/s-level.

The level of automation is about the same as for the IAG 50 cm, with the exception of the piggyback telescope.

6.4 MONET/S

With MONET/S, located at the South African Astronomical Observatory (SAAO) near Sutherland, South Africa, having mostly the same hardware as MONET/N, there are still some differences. The science camera is currently a FLI PL230 and there is a 0.25 m f/8 piggyback telescope mounted at the (unused) second Nasmyth port, with a SBIG STX-8300M camera attached to a Gemini derotator and focuser. Furthermore, outside the field of view of the main camera we installed a pickup for a fiber bundle leading to MORISOT, a low-budget, low-resolution spectrograph.

Again, the level of automation is similar to its twin in Texas and the IAG 50 cm. **Figure 10** shows an illustration of all pyobs module running at MONET/S, how they are distributed over several computers, and how they are connected to the actual hardware. As one can see, there is an additional module for the derotator of the piggyback telescope that we will publish as soon as it is fully tested. Acquisition (with an offset) and guiding of the spectrograph is supposed to be done with the science camera, and we hope to be able to do parallel photometry of the target with the piggyback. The custom module for the roof simply calls HTTP REST endpoints on our roof controller, and `BonnShutter`

continuously checks the health of our Bonn shutter, and resets it, if any error occurs.

When everything is finished, there will probably be three modules for acquisition (one for each instrument) and three modules for guiding: science-frame auto-guiding on the main camera, guiding via piggyback, and guiding via science camera for the spectrograph. There will also be auto-focusing for both telescopes and flat-fielding for all three instruments. A challenge will be to calibrate all three instruments during twilight, but with the flexible scripts in the robotic part of pyobs, this should not be too much of a problem.

7 DEVELOPMENT

The development of pyobs is completely public, all the projects are hosted on GitHub and we use its "Issues" page as a bug tracker. At the moment, all the code comes from a single institution. We would, however, love to see contributions from other people, and would be glad to accept pull requests.

Development for pyobs has two different sides: creating new modules or change existing ones is quickly done and we can include as many into the main package as we went, as long as they provide some functionality that is otherwise missing and required by some observatories. Some possible extensions that come to mind are, e.g., support for dithering, focus offsets with respect to filters (which is already supported by the `FocusModel` module), guiding with PHD2, and so on.

Changing the core of pyobs (like, e.g., interfaces, error handling, communication, . . .) on the other hand would have to include some discussions in order to make sure that no existing code is broken and that it fits the general design philosophy of pyobs. Feel free to contact the author about any changes you would like to see.

As the leading zero in pyobs' version number indicates, we do not assume pyobs to be in a "stable" condition, i.e. major (even breaking) changes can occur with every new version. However, the number of these changes has reduced significantly as of late. A potential user still needs to understand that things can and will change, which, of course, also gives the opportunity to actively shape those changes. As soon as we reach a stable version, we will fully implement semantic versioning and only apply breaking changes for new major versions.

For us, pyobs is mainly a tool for operating our four telescopes (see **Section 6**). Therefore, keeping it in a state that is useful for us is our top priority. However, as we already showed with our solar telescope, we are willing to adapt existing code to work in new environments. That said, our time is limited, so we will not be able to give full-time support, but we continuously work on the documentation and try reply to emails and GitHub issues as quickly as possible. We would love to see a little community growing around pyobs that actively develops and supports it.

One group of observational astronomers that we have skipped over completely in this paper are the amateurs. Over the last decades they have built an amazing foundation of tools to build on, be it ASCOM, INDI, N.I.N.A.[18], and so many others. Nowadays, amateur astronomers do some scientific work that many professional observatories cannot do anymore, e.g., long-time monitoring of variables. One great example as of late was the dimming of Betelgeuse, which is far too bright for larger telescopes.

However, at least in its current state, we do not believe that pyobs is a good tool for amateur astronomers. It was designed mainly for robotic observations at professional observatories. For instance, the GUI, which is an essential part for every remote setup, is mainly a maintenance tool for us and not used during regular observations. With pyobs being as open as it is, though, there is no reason, why it can not be developed into a direction that would make it more useful for amateurs as well. Therefore, any interested amateur astronomers are welcome to play around with pyobs, improve it, and contact us with any question or comment.

8 SUMMARY AND OUTLOOK

In this paper we presented the observation control system pyobs. While pyobs itself is written in *Python* and highly depends on third-party packages, it can easily be extended by any programming language that supports the communication protocol XMPP. We showed that pyobs is highly customizable due to its configuration files, and provides a lot of functionality for robotic telescope operations out of the box: it has support for common tasks like flat-fielding and auto-focus series and connects to the open-source LCO observation portal for organizing tasks.

At the time of writing this paper, pyobs is available in version 0.17. As the leading zero suggests, we do not believe that it has reached a stable state, in which no major changes to any of its system will happen in the near future. However, at least the currently planned modifications are mostly minor, and we expect to publish a first release this year or soon thereafter–so this should not keep anyone from using pyobs before that.

The list of planned improvements for the core of pyobs is long, but contains mostly minor items, which probably will not affect running systems. Some of the more major ones are:

- The error handling (see **Section 2.6**) is quite new and not used everywhere. It is missing, especially, in the robotic modules.
- Some access control will be added, so that a module can allow some of its methods to be called only by authorized clients.
- There already exist a few unit tests for the core package, but they are not covering everything, not even the most important parts.
- New interfaces (see **Section 2.2.2**) will be added–e.g., for supporting to track non-sidereal targets –, which might make it necessary to change existing ones.

While these items are for the core system, future development will mainly concentrate on additional modules. For instance, we would like to guide using a guiding telescope, which would require applying some pointing model to the offsets in order to compensate for different movements of the telescopes like bending and (thermal) stretching. We would also like to add a wrapper to the PHD2 guiding software, which would allow us to use this well-tested package in addition to our own guiding modules. Furthermore, as mentioned in **Section 5.1**, a new (non-greedy) scheduler is high up on the wish list.

Using existing software was the goal for pyobs from the beginning. Instead of developing code from scratch it was built

[18]https://nighttime-imaging.eu/

on top of widely used *Python* packages from the astronomical community and beyond. With pyobs-alpaca we already showed that we can bridge towards other protocols, and there probably will be a wrapper for INDI as well, which we can use to add devices for which an INDI driver already exists. There is also a plan to add wrappers for client software like Stellarium[19] or KStars, [20] that will make accessing a remote system easier, e.g. for students.

We will continue developing pyobs mainly for our own telescopes, but always trying to be as general as possible, so that it can be used by other observatories. The documentation is a good place to start playing around with pyobs and will be extended continuously. The author of this paper is looking forward to any contribution to pyobs, any comment and suggestion for improvement, and any question via email or GitHub issue tracker.

DATA AVAILABILITY STATEMENT

The software presented in this article can be found here: https://github.com/pyobs https://www.pyobs.org/.

AUTHOR CONTRIBUTIONS

T-OH is the main developer of pyobs. FH, the former PI of MONET, developed a few of the device modules and gave helpful input for the big picture. KR and SM worked on adapting pyobs for the solar telescope. TM implemented the auto-guiding for single stars. SS is the PI of the FTS and responsible for a long list of feature requests and suggestions for improvements.

ACKNOWLEDGMENTS

The development of pyobs and its modules was only possible by using several *Python* packages (in alphabetical order): Aiohttp, an asynchronous HTTP Client/Server for asyncio and *Python*.[21]. Astroplan, an open source *Python* package to help astronomers plan observations (Morris et al., 2018). Astropy,[22] a community-developed core *Python* package for Astronomy (Astropy Collaboration et al., 2013; Astropy Collaboration et al., 2018). Astroquery, a set of tools for querying astronomical web forms and databases (Ginsburg et al., 2019). asyncinotify, an async python inotify package.[23]. ccdproc, an Astropy package for image reduction (Craig et al., 2017). Cython, an optimising static compiler for the *Python* programming language.[24]. lmfit, Non-Linear Least-Squares Minimization and Curve-Fitting for *Python*.(Newville et al., 2021). Matplotlib, a comprehensive library for creating static, animated, and interactive visualizations in *Python*.[25]. Numpy, a fundamental package for scientific computing with *Python* (Harris et al., 2020). Pandas, an open source data analysis and manipulation tool (McKinney, 2010; Pandas Development Team, 2020). Paramiko, a pure-Python implementation of the SSHv2 protocol.[26]. Photutils, an Astropy package for detection and photometry of astronomical sources (Bradley et al., 2020). py-expression-eval, a *Python* mathematical expression evaluator.[27]. PyQt5, a set of *Python* bindings for Qt application framework.[28]. Python-aravis, a Pythonic interface to the auto-generated aravis bindings.[29]. Python-daemon, *Python* library to implement a well-behaved Unix daemon process.[30]. Python-telegram-bot, a *Python* wrapper for using Telegram.[31]. Python-zwoasi, a *Python* binding to the ZWO ASI version two library.[32]. PyYAML, a full-featured YAML framework for the *Python* programming language.[33]. Qasync, an implementation of the PEP 3156 event-loop to be used in PyQt applications.[34]. Scipy, a package for fundamental algorithms for scientific computing in *Python* (Virtanen et al., 2020). SEP, a *Python* and C library for Source Extraction and Photometry (Barbary, 2016; Barbary et al., 2017), based on Source Extractor (Bertin and Arnouts, 1996). Single-source, a single source of truth for version and name of a project.[35]. Slixmpp, an XMPP library for *Python* 3.7 + .[36]. Some more packages are currently used by pyobs but not mentioned here, since they are going to be replaced soon. The GUI uses icons from the "Crystal Clear" set.[37]. Running our own instance of the LCO Observation Portal as well as connecting it to pyobs was made possible with the help of the great team at Las Cumbres Observatory (LCO). We also use parts of the frontend of their science archive. Both are parts of the LCO Observatory Control System (OCS).[38]

[19]http://stellarium.org
[20]https://edu.kde.org/kstars/
[21]https://docs.aiohttp.org/
[22]http://www.astropy.org
[23]https://asyncinotify.readthedocs.io/
[24]https://cython.org
[25]https://matplotlib.org
[26]https://www.paramiko.org/
[27]https://github.com/AxiaCore/py-expression-eval/
[28]https://www.riverbankcomputing.com/software/pyqt/
[29]https://github.com/SintefManufacturing/python-aravis
[30]https://pagure.io/python-daemon/
[31]https://github.com/python-telegram-bot/python-telegram-bot
[32]https://github.com/python-zwoasi/python-zwoasi
[33]https://pyyaml.org
[34]https://github.com/CabbageDevelopment/qasync
[35]https://github.com/rabbit72/single-source
[36]https://slixmpp.readthedocs.io/
[37]https://commons.wikimedia.org/wiki/Crystal_Clear
[38]https://observatorycontrolsystem.github.io/

REFERENCES

Astropy CollaborationPrice-Whelan, A. M., Price-Whelan, A. M., Sipőcz, B. M., Günther, H. M., Lim, P. L., Crawford, S. M., et al. (2018). The Astropy Project: Building an Open-Science Project and Status of the v2.0 Core Package. *AJ* 156, 123. doi:10.3847/1538-3881/aabc4f

Astropy CollaborationRobitaille, T. P., Robitaille, T. P., Tollerud, E. J., Greenfield, P., Droettboom, M., Bray, E., et al. (2013). Astropy: A Community Python Package for Astronomy. *A&A* 558, A33. doi:10.1051/0004-6361/201322068

Akerlof, C. W., Kehoe, R. L., McKay, T. A., Rykoff, E. S., Smith, D. A., Casperson, D. E., et al. (2003). The ROTSE-III Robotic Telescope System. *Publ. Astron Soc. Pac* 115, 132–140. doi:10.1086/345490

Alcock, C., Axelrod, T. S., Bennett, D. P., Cook, K. H., Park, H. S., Griest, K., et al. (1992). "The Search for Massive Compact Halo Objects with a (Semi) Robotic Telescope," in *Robotic Telescopes in the 1990s. Vol. 103 of Astronomical Society of the Pacific Conference Series*. Editor A. V. Filippenko, 193–202.

Antonelli, L. A., Zerbi, F. M., Chincarini, G., Ghisellini, G., Rodonò, M., Tosti, G., et al. (2003). The REM Telescope: a Robotic Facility to Monitor the Prompt Afterglow of Gamma Ray Bursts. *Mem. Soc. Astron. Ital.* 74, 304.

Barbary, K. (2016). Sep: Source Extractor as a Library. *Joss* 1, 58. doi:10.21105/joss.00058

Bellm, E. C., Kulkarni, S. R., Barlow, T., Feindt, U., Graham, M. J., Goobar, A., et al. (2019). The Zwicky Transient Facility: Surveys and Scheduler. *PASP* 131, 068003. doi:10.1088/1538-3873/ab0c2a

Bertin, E., and Arnouts, S. (1996). SExtractor: Software for Source Extraction. *Astron. Astrophys. Suppl. Ser.* 117, 393–404. doi:10.1051/aas:1996164

Bischoff, K., Tuparev, G., Hessman, F. V., and Nikolova, I. (2006). "MONET/North: a Very Fast 1.2m Robotic Telescope," in *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series. Vol. 6270 of Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*. Editors D. R. Silva and R. E. Doxsey, 62701Q. doi:10.1117/12.671433

Castro-Tirado, A. J., Jelínek, M., Mateo Sanguino, T. J., and de Ugarte Postigo, A.the BOOTES Team (2004). BOOTES: A Stereoscopic Robotic Ground Support Facility. *Astron. Nachr.* 325, 679. doi:10.1002/asna.200410333

Chromey, F. R., and Hasselbacher, D. A. (1996). The Flat Sky: Calibration and Background Uniformity in Wide Field Astronomical Images. *PASP* 108, 944. doi:10.1086/133817

Barbary, K., Boone, K., Craig, M., Deil, C., and Rose, B. (2017). *Kbarbary/Sep: v1.0.2*. doi:10.5281/zenodo.896928

Bradley, L., Sipőcz, B., Robitaille, T., Tollerud, E., Vinícius, Z., Deil, C., et al. (2020). *Astropy/Photutils: 1.0.0*. doi:10.5281/zenodo.4044744

Craig, M., Crawford, S., Seifert, M., Robitaille, T., Sipőcz, B., Walawender, J., et al. (2017). *Astropy/Ccdproc: v1.3.0.Post1*. doi:10.5281/zenodo.1069648

Newville, M., Otten, R., Nelson, A., Ingargiola, A., Stensitzki, T., Allan, D., et al. (2021). *Lmfit/Lmfit-Py: 1.0.3*. doi:10.5281/zenodo.5570790

Pandas Development Team, T. (2020). *Pandas-Dev/Pandas: Pandas*. doi:10.5281/zenodo.3509134

Filippenko, A. V., Li, W. D., Treffers, R. R., and Modjaz, M. (2001). "The Lick Observatory Supernova Search with the Katzman Automatic Imaging Telescope," in *IAU Colloq. 183: Small Telescope Astronomy on Global Scales. Vol. 246 of Astronomical Society of the Pacific Conference Series*. Editors B. Paczynski, W. P. Chen, and C. Lemme, 183, 121–130. doi:10.1017/s0252921100078738

Ginsburg, A., Sipőcz, B. M., Brasseur, C. E., Cowperthwaite, P. S., Craig, M. W., Deil, C., et al. (2019). Astroquery: An Astronomical Web-Querying Package in Python. *AJ* 157, 98. doi:10.3847/1538-3881/aafc33

Granzer, T. (2006). STELLA and RoboTel - a Prototype for a Robotic Network? *Astron. Nachr.* 327, 792–795. doi:10.1002/asna.200610635

Granzer, T., Weber, M., and Strassmeier, K. G. (2012). "The STELLA Control System," in *Astronomical Society of India Conference Series. Vol. 7 of Astronomical Society of India Conference Series*, 247.

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., et al. (2020). Array Programming with NumPy. *Nature* 585, 357–362. doi:10.1038/s41586-020-2649-2

Henry, G. W., Fekel, F. C., and Hall, D. S. (1995). An Automated Search for Variability in Chromospherically Active Stars. *AJ* 110, 2926. doi:10.1086/117740

Hessman, F. V. (2001). "MONET: a MOnitoring NEtwork of Telescopes," in *IAU Colloq. 183: Small Telescope Astronomy on Global Scales. Vol. 246 of Astronomical Society of the Pacific Conference Series*. Editors B. Paczynski, W. P. Chen, and C. Lemme, 183, 13–21. doi:10.1017/s0252921100078544

Hessman, F. V. (2004). The MONET Project and beyond. *Astron. Nachr.* 325, 533–536. doi:10.1002/asna.200410274

Hidas, M. G., Hawkins, E., Walker, Z., Brown, T. M., and Rosing, W. E. (2008). Las Cumbres Observatory Global Telescope: A Homogeneous Telescope Network. *Astron. Nachr.* 329, 269–270. doi:10.1002/asna.200710950

Ivezić, Ž., Kahn, S. M., Tyson, J. A., Abel, B., Acosta, E., Allsman, R., et al. (2019). LSST: From Science Drivers to Reference Design and Anticipated Data Products. *ApJ* 873, 111. doi:10.3847/1538-4357/ab042c

Kuba´nek, P., Jelínek, M., Nekola, M., Topinka, M., Štrobl, J., Hudec, R., et al. (2004). "RTS2 - Remote Telescope System, 2nd Version," in *Gamma-Ray Bursts: 30 Years of Discovery. Vol. 727 of American Institute of Physics Conference Series*. Editors E. Fenimore and M. Galassi, 753–756. doi:10.1063/1.1810951

Lang, D., Hogg, D. W., Mierle, K., Blanton, M., and Roweis, S. (2010). Astrometry.net: Blind Astrometric Calibration of Arbitrary Astronomical Images. *Astronomical J.* 139, 1782–1800. doi:10.1088/0004-6256/139/5/1782

Law, N. M., Kulkarni, S. R., Dekany, R. G., Ofek, E. O., Quimby, R. M., Nugent, P. E., et al. (2009). The Palomar Transient Factory: System Overview, Performance, and First Results. *Publ. Astron Soc. Pac* 121, 1395–1408. doi:10.1086/648598

Lipunov, V. M., Kornilov, V. G., Krylov, A. V., Kuvshinov, D. A., Gorbovskoy, E. S., Tyurina, N. V., et al. (2007). Observations of Gamma-Ray Bursts and a Supernovae Search at the Robotic Telescope MASTER. *Astronomical Astrophysical Trans.* 26, 79–86. doi:10.1080/10556790701300462

McKinney, W. (2010). "Data Structures for Statistical Computing in Python," in Proceedings of the 9th Python in Science Conference. Editors S. van der Walt and J. Millman, 56–61. doi:10.25080/Majora-92bf1922-00a

Morris, B. M., Tollerud, E., Sipőcz, B., Deil, C., Douglas, S. T., Medina, J. B., et al. (2018). Astroplan: An Open Source Observation Planning Package in Python. *AJ* 155, 128. doi:10.3847/1538-3881/aaa47e

Perlmutter, S., Muller, R. A., Newberg, H. J. M., Pennypacker, C. R., Sasseen, T. P., and Smith, C. K. (1992). "A Doubly Robotic Telescope: the Berkeley Automated Supernova Search," in *Robotic Telescopes in the 1990s. Vol. 103 of Astronomical Society of the Pacific Conference Series*. Editor A. V. Filippenko, 67–71.

Riddle, R., Cromer, J., Hale, D., Henning, J., Baker, J., Milburn, J., et al. (2018). "The Zwicky Transient Facility Robotic Observing System (Conference Presentation)," in *Observatory Operations: Strategies, Processes, and Systems VII*. Editors A. B. Peck, C. R. Benn, and R. L. Seaman (Bellingham, WA, USA: SPIE). doi:10.1117/12.2312702

P. Saint-Andre (Editor) (2004). "Extensible Messaging and Presence Protocol (XMPP): Core," *RFC 3920, RFC*.

Schäfer, S., Huke, P., Meyer, D., and Reiners, A. (2020a). "Fiber-coupling of Fourier Transform Spectrographs," in *Ground-based and Airborne Instrumentation for Astronomy VIII*. Editors C. J. Evans, J. J. Bryant, and K. Motohara (Bellingham, WA, USA: International Society for Optics and Photonics (SPIE)), Vol. 11447, 784–795. doi:10.1117/12.2561599

Schäfer, S., Royen, K., Huster Zapke, A., Ellwarth, M., and Reiners, A. (2020b). "Observing the Integrated and Spatially Resolved Sun with Ultra-high Spectral Resolution," in *Ground-based and Airborne Instrumentation for Astronomy VIII*. Editors C. J. Evans, J. J. Bryant, and K. Motohara (Bellingham, WA, USA: International Society for Optics and Photonics (SPIE)), Vol. 11447, 2187–2208. doi:10.1117/12.2560156

Strassmeier, K. G., Bartus, J., Cutispoto, G., and Rodonó, M. (1997). Starspot Photometry with Robotic Telescopes. *Astron. Astrophys. Suppl. Ser.* 125, 11–63. doi:10.1051/aas:1997369

Street, R. A., Pollaco, D. L., Fitzsimmons, A., Keenan, F. P., Horne, K., Kane, S., et al. (2003). "SuperWASP: Wide Angle Search for Planets," in *Scientific*

Frontiers in Research on Extrasolar Planets. *Vol. 294 of Astronomical Society of the Pacific Conference Series*. Editors D. Deming and S. Seager, 405–408.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., et al. (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nat. Methods* 17, 261–272. doi:10.1038/s41592-019-0686-2

Conflict of Interest: Author TM was employed by company TNG Technology Consulting GmbH.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

frontiers | Frontiers in Astronomy and Space Sciences

TYPE Methods
PUBLISHED 19 August 2022
DOI 10.3389/fspas.2022.924871

OPEN ACCESS

EDITED BY
Shilong Liao,
Shanghai Astronomical Observatory (CAS), China

REVIEWED BY
Edwin Quintero,
Technological University of Pereira, Colombia
Vladimir Lipunov,
Lomonosov Moscow State University, Russia

*CORRESPONDENCE
A. J. Castro-Tirado,
ajct@iaa.es

SPECIALTY SECTION
This article was submitted to Astronomical Instrumentation, a section of the journal Frontiers in Astronomy and Space Sciences

RECEIVED 20 April 2022
ACCEPTED 14 July 2022
PUBLISHED 19 August 2022

CITATION
Castellón A, Larrubia G, Castillo R, Castro-Tirado AJ, Fernández-García E, Pérez del Pulgar C, Reina A and Castillo-Carrión S (2022), Dome control with fuzzy logic.
Front. Astron. Space Sci. 9:924871.
doi: 10.3389/fspas.2022.924871

Dome control with fuzzy logic

A. Castellón[1,2], G. Larrubia[3], R. Castillo[3], A. J. Castro-Tirado[4,2]*, E. Fernández-García[4], C. Pérez del Pulgar[2], A. Reina[2] and S. Castillo-Carrión[5]

[1]Department of Álgebra, Geometría y Topología, University of Málaga, Málaga, Spain, [2]Departamento de Ingeniería de Sistemas y Automática, Escuela de Ingeniería Industrial, Unidad Asociada al CSIC, University of Málaga, Málaga, Spain, [3]Alumni of the University of Málaga, Málaga, Spain, [4]Instituto de Astrofísica de Andalucía (IAA) CSIC, Granada, Spain, [5]Enseñanza Virtual y Laboratorios Tecnológicos, University of Málaga, Málaga, Spain

This article describes a fuzzy logic-based method optimized for the dome control of a robotic astronomical observatory. A Mamdani inference has been developed in order to make the decision to open or close the dome. The input variables are obtained from data received from a weather station besides the percentage of cloudy sky as derived by two allsky cameras. The software has been tested at the BOOTES-1 observatory as part of the BOOTES Global Network of Robotic Telescopes led by IAA-CSIC.

KEYWORDS

fuzzy logic, Mamdani, inference model, dome control, robotic observatory

1 Introduction

The BOOTES Global Network (Burst Observer and Optical Transient Exploring System) have so far five observatories located in Algarrobo Costa and El Arenosillo (Spain), Lauder (New Zealand), San Pedro Mártir (México), and Yunnan (China). Their main goal is to find and study optical counterparts from transient phenomena such as gamma-ray bursts, gravitational wave bursts, and others. Most information about GRB and GW triggers are distributed through GCN (Barthelmy, 1998) between subscribers. The first robotic telescope network was ROTSE (Akerlof et al., 1999), and now between working are MASTER (Lipunov, 2019) and BOOTES (Castro-Tirado et al., 2012).

Therefore, they do not have semi-spherical dome, instead they do have a gabled dome that opens completely allowing a fast movement from the telescope toward any point in the sky as response to any astronomical alert deserving fast reaction.

The observatories are robotized. Therefore, the observation time is set depending on the height of the Sun above the horizon, and they respond to alerts they receive from different sources. As response time needs to be in a very short space of time, the dome has to stay open as much time as possible, that is to say, every time sky conditions allow it and meteorological situation does not signify a danger to the set of instruments.

As the two relevant subsystems are a meteorological station and two allsky cameras, one inside the dome and other outside, a software program inspects in real-time information from the meteorological station and the cameras, and it decides whether or not to open the dome. However, this decision based on classical logic has turned out to be too rigid. Sometimes, the software, when the thresholds set for some parameters were exceeded, decided to closed the dome for a period of time, and it was found a posteriori

that it could have been open, losing the opportunity to observe the transient phenomenon that happened precisely at that instant. (Murphy's law usually works).

That is why we thought about fuzzy logic because in these kind of situations, it adopts human behavior, is more flexible, and it does not keep too harsh yes or no decisions.

The chosen method has been developed by Mamdani and Assilian (1975) and it is widely used in decision making problems. It is stated in four steps:

1. **Fuzzification of the input variables.** Input variables are taken, and their memberships to associate fuzzy sets are calculated.
2. **Rule evaluation.** Fuzzy rules are applied to the input variables. If a given fuzzy rule has multiple antecedents, the operators are used to obtain a single evaluation number. This value is applied to the consequent using the minimum method (truncates the consequent with the antecedent's truth value) or product method (multiply all values by antecedent's truth value).
3. **Aggregation of the rule outputs.** A single fuzzy set is obtained by unifying the outputs from the previous fuzzy rules and by combining membership functions of all rule consequents.
4. **De-fuzzification.** The result is shown as a classic membership value taking as input the fuzzy set from the previous step. The method used is the centroid, which returns the center of the area under the associated membership functions:

$$\text{centroide} = \frac{\int \mu_A(x)x}{\int \mu_A(x)}, \tag{1}$$

where x is the input variable and μ_A is the membership function.

The Mamdani fuzzy inference method has been widely used in a variety of fields, including those related to meteorology, as can be seen in Manish et al. (2019) and Agboola et al. (2013).

2 Problem's data

2.1 Input data collection

The BOOTES observatories have two allsky cameras named CASANDRA (Compact All-Sky Array of Night Devices for Rapid Alerts) and Starcam. CASANDRA is composed by a Moravian G4-16000 with Nikon equipped with a fish-eye 16 mm lens, and it is located inside the observatory. The Starcam is a ZWO120MM wide-angle lens situated outside the observatory, and it is protected by a carbon fiber transparent dome. Astrometry is obtained from both cameras. This allows calculating in real time the chip coordinates where the stars should be projected.

The software looks in the CCD or CMOS detector for the stars among those from the Hipparcos catalog with magnitude below 3 and 5, and it fills files (one per camera) with the following structure:

1. Date/time in FITS format.
2. Zenithal distance of the Moon in degree format.
3. Illuminated Moon fraction (percentage).
4. Percentage of stars with magnitude below 3 found.
5. Percentage of stars with magnitude below 5 found.

The following is a sample line from these files:
2022-01-20T06:30:34___56.13___95.00%___86.11%___53.85%

This mean that at 06:30:34 time on 20 January 2022, the Moon set at 56.13 deg of zenithal distance had 95% illuminated Moon surface, 86.11% stars in range with magnitude below 3 were found, and 53.85% stars in range with magnitude below 5 were found.

The fact that zenithal distance of the Moon percentage of illuminated Moon surface is considering, in moonless sky or when the Moon is low above the horizon, cloud absence will turn into a high percentage of stars found, but the Moon presence decreases these values due to three main causes:

- The Moon saturates a significative fraction of the CCD or CMOS detector depending on its brightness, reducing useful exploration area.
- Sky background increases its brightness preventing fainter stars to be detected.
- Strong moonlight ensures, at the outside camera dome (Starcam), all the imperfections that complicate star detection such as scratches, bird or insect excrements, and rain or dew drops.

Software used in both cameras removes a circular area centered at the Moon location, and the given radius is adapted to the percentage of the illuminated Moon and at the zenithal distance of the Moon. As it shall be seen later, it is required that modified functions to input variable membership values are introduced.

In any case, Moon's influence on star count is higher in Starcam (outside) than in CASANDRA (inside), which makes sense due to the sensitivity and optics difference between them.

Another software process runs in real time, reads information from meteorological station, and produces files with the following structure:

1. Date in YYYY-MM-DD format
2. Universal time in hh:mm:ss format
3. Atmospheric pressure in inch of mercury column
4. Inside temperature in Fahrenheit degrees
5. Outside temperature in Fahrenheit degrees
6. Inside relative humidity
7. Outside relative humidity

10.3389/fspas.2022.924871

TABLE 1 Correlation between pairs of variables.

Variables to contrast	Correlation coefficient
Stars with magnitude 3 and 5	0.9325
Inside and outside temperature	0.9185
Inside and outside humidity	0.7623
Inside and outside dew point	0.9502

8. Inside dew point in Fahrenheit degrees
9. Outside dew point in Fahrenheit degrees
10. Current wind speed in miles per hour
11. Average wind speed during last 10 min in miles per hour
12. Wind direction in azimuth degrees from North and clockwise, that is to say, 0 for North, 90 for East, etc.
13. Day rain until current time in inches
14. Inch per hour from daily rain
15. Dome status (negative value if dome is closed or there is no signal, and 1 if dome is open).

Moonless nights are considered when the zenithal distance of the Moon is higher than 85 deg (or below the horizon) of the Moon illuminated surface is lower than 10%. In both cases, Moon's influence is not significant in percentage of stars found by Starcam.

A previous statistics with data from a complete year was made to check relationship between the fields (3, 4), (5, 6), and (7, 8) and between the percentages of stars found and stars with magnitude below 3 and 5. The results are given in Table 1.

Certainly, there is a correlation between these field pairs, and for this particular reason, the number of variables is simplified, and both the percentage of stars with 3rd magnitude and the outside meteorological data will be used.

2.2 Linguistics variables

The following variables will be used:

- VL: very low
- L: low
- N: normal
- H: high
- VH: very high

To infer the corresponding values, all the values from each field in the yearly statistic were ordered from low to high until they reach

- For VL: 7% of the sample
- For L: 16% of the sample
- For N: 50% of the sample (median)
- For H: 66% of the sample
- For VH: 93% of the sample

This method will be applied in most variables, except rain, humidity, and CASANDRA and Starcam values, which will have a designed process.

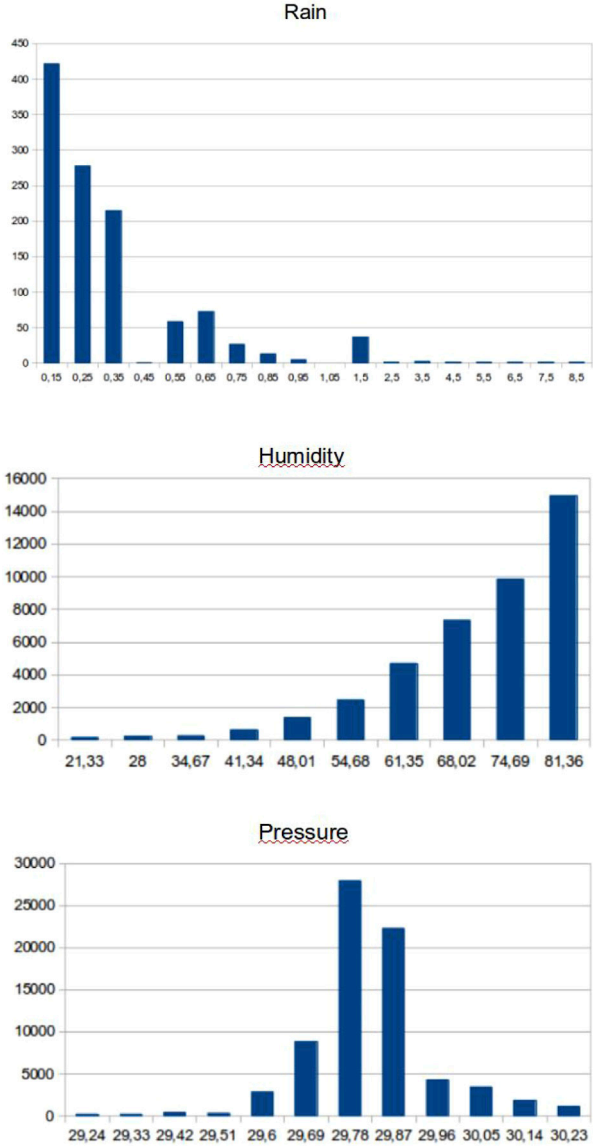


FIGURE 1
Rain, humidity, and pressure frequency.

2.3 Rain

The membership function from this variable will take only value 1 for rain absence, and value 0 for rain (field 13). Obviously,

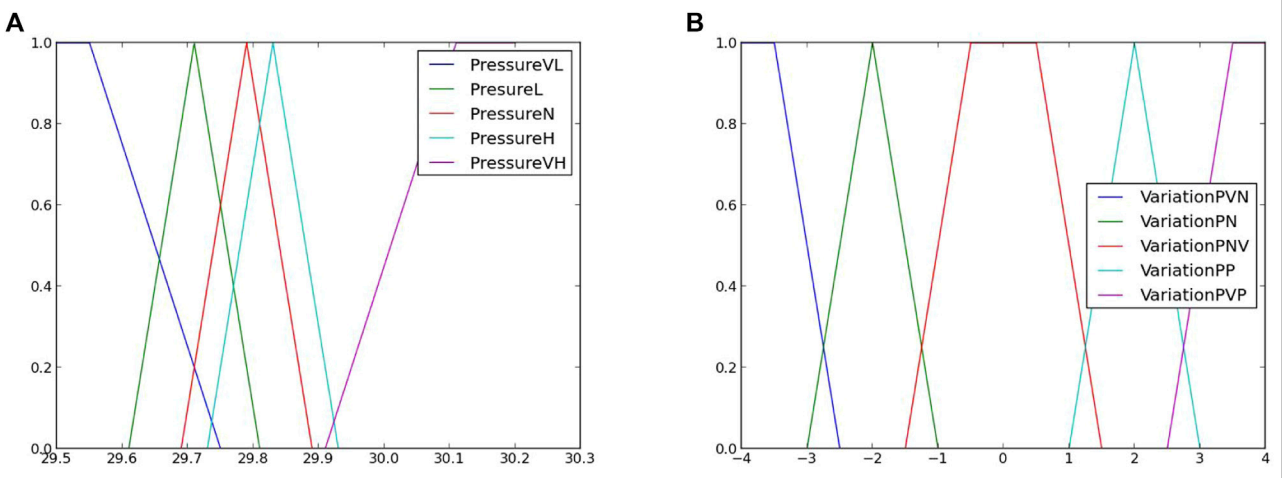


FIGURE 2
Pressure variable membership functions **(A)**. Pressure variation membership functions **(B)**.

the dome must stay closed during any rain. This is in fact a classic set and not a fuzzy one. Due to local climatology, only in 1,145 (out of 37,981) were rain recorded, and this is covered in Figure 1 left.

2.4 Humidity

Once again, a classic set will be used. The membership function will take value 0 if relative humidity is greater or equal to 95%, and value 1 otherwise. During high humidity, optic, mechanic, and electronic instrumentation are in danger; moreover, the stars found collection is lower due to water condensation on top of outside dome (Figure 1 center).

2.5 Dome

This is the last no fuzzy set. It will use field 15 (open or closed dome). If dome is closed or there is no signal, membership function will return value 0. Otherwise (open dome), it will return value 1. If dome is open, records from inside CASANDRA camera (inside) will be used instead of outside Starcam camera (outside).

2.6 Pressure

Low pressure predicts storm, cyclone or torrential rain appearance, very pernicious situations for the telescope, and assembling and electronics. In this circumstance, the dome must stay closed. High pressure means steady weather with a high probability of clear sky, which is positive for opening the dome. In the year, statistic sample considered the atmospheric pressure has a median of 29,805 inches of mercury. That is the normal value (N) for the membership functions. From that to the maximum, there will be set values high (H) or very high (VH), and from median to the minimum, values low (L) or very low (VL) (Figure 1 right).

Pressure values on their own do not give more information than possible bad weather. More clarifying is pressure variation during a time interval before decision making, which will be described in next subsection. Very low (MB) pressure reaches at 29.65, low (L) at 29.71, normal (N) at 29.79, high (H) at 29.83, and very high (VH) at 30.01 (Figure 2A).

2.7 Pressure variation

Pressure variation values are obtained by reading pressure field 20 min before making the decision to open or close the dome. Let set a counter at 0, each pressure values rise between one record and the next increases counter by 1, and each descent decreases it by 1. If there is no pressure change, counter stays invariable. A study was developed with the relationship between pressure variation and rain records and stars collections from both cameras to analyze pressure variation influence on clouds or rain appearance. It was noticed that a very negative pressure variation leads to rain and percentage of stars found is very low. With positive variations, it can rain, but only with a high standard deviation; for this reason, perhaps the weather gets better.

It is notified that when pressure decreases slowly, there is practically no rain. Rain appears when pressure decreases roughly, and as cumulative rain increases, pressure also does it slowly. To summarize, very negative pressure variation leads to

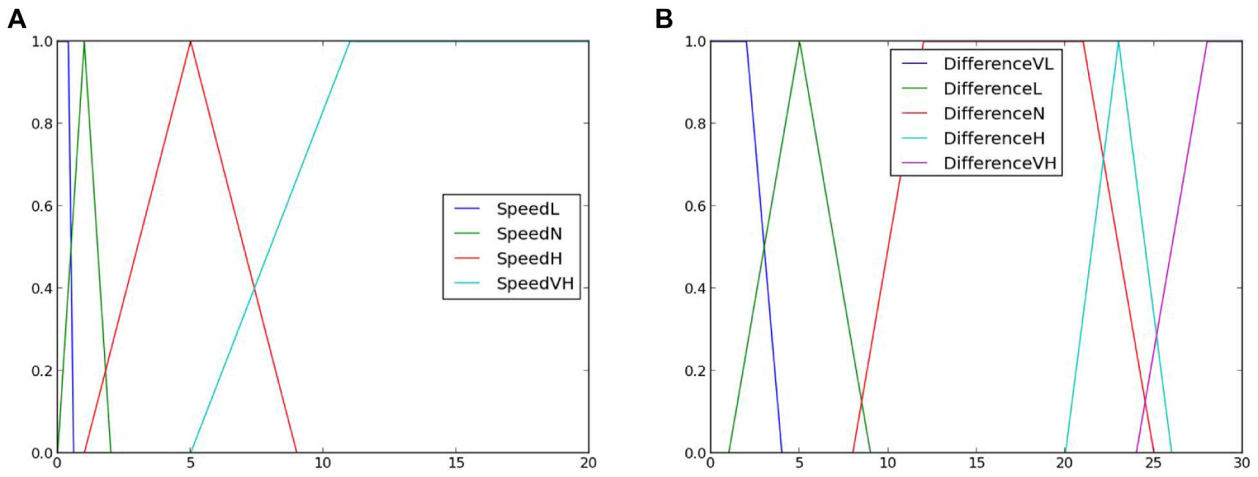


FIGURE 3
Wind speed membership functions **(A)**. Temperature and dew point difference membership functions **(B)**.

clouds and rain, low pressure variation shows steady weather, and positive pressure variation shows that weather gets better or worse (it not only depends on pressure absolute value and but also on variation value).

The membership functions are made by considering that most relevant data are obtained with negative, very negative, positive, and very positive pressure variations. Triangular functions are used for these variables and a trapezoidal one when there is no variation. Very negative (VN) pressure variation reaches at -3, negative (N) at -2, no variation (WV) at 0, positive (P) at 2, and very positive (VP) at 3 (Figure 2B).

2.8 Wind speed

This variable excludes itself from Gaussian function because there are only a few days with significant wind preventing the observations. Indeed, these are the most dangerous days for the telescope assembling and the dome that when it is open acts like a sail.

Reference values are 0.5 for low speed (L), 1 for normal speed (N), 5 for high speed (H), and 11 for very high speed (VH). These membership functions are reflected in Figure 3A.

2.9 Wind direction

For analyzing this variable, low wind speed has been discarded because, in this circumstance, wind direction is irrelevant. The analysis of percentage of stars found with CASANDRA and Starcam depending on wind direction shows that percentage is very low with South/Southeast wind and relatively high with North/Northwest wind, as can be seen in the circular graphics in Figure 4.

The distance to the center represents the average percentage of right stars searched. It can be seen in both graphics that the southeastern wind lowers the percentage, whereas the northern wind improves the situation.

2.10 Temperature and dew point difference

As expected in the Starcam, when temperature is close to dew point, the average percentage of stars found is very low. The lowest percentage reaches at a difference of 2.06 Fahrenheit degrees. On the other hand, when this difference is very high (from 19.64 Fahrenheit degrees), the average percentage is very high, and standard deviation is very low, which indicates not only a lot of accuracy but also what is mostly happening.

All this considered membership function takes the following values: 3 for very low difference (VL), 5 for low difference (L), 20 for normal difference (N), 23 for high difference (H), and 26 for very high difference (VH). This is shown in Figure 3B.

2.11 CASANDRA and Starcam records

These values will be added to meteorological variables. They will be useful when Starcam shows high or very high percentage of accuracy because this means good sky, and it is convenient to open the dome. While dome is open, CASANDRA records will be considered. This is because CASANDRA is more efficient than Starcam in star search, even in unfavorable Moon conditions.

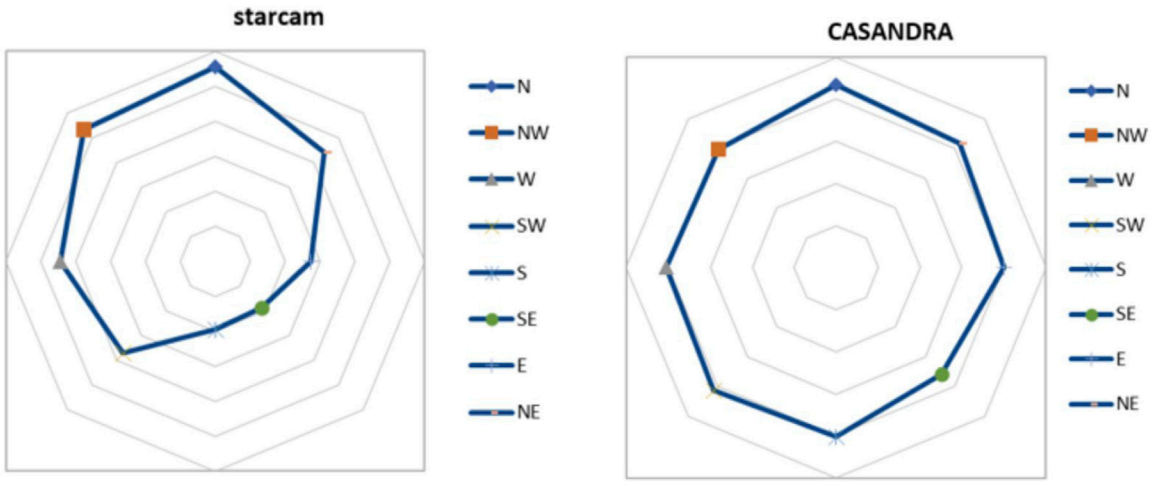


FIGURE 4
Percentage of hits according to wind direction frequencies.

This fact has been inferred by performing statistics on a sample of 12,975 moonless records and 20,238 Moon night records. Usually, percentage of accuracy is high because the robotized classic logic-based system that was in charge during statistic opened the dome when meteorological conditions were good. Not surprisingly is that percentage of accuracy is higher in moonless nights than in nights with Moon. Either way, while CASANDRA records are not bad, there is no problem with keeping the dome open. This is the reason why CASANDRA membership function will return 1 at percentage of accuracy higher than 50% and 0 otherwise, that is, like a no fuzzy set.

Starcam situation is very different. As it is outside the dome, it captures pictures in all circumstances: rainy, clear sky, cloudy, dew, and with Moon or moonless. It is less sensitive, and it field of view is smaller than of the CASANDRA one. Due to being protected by a carbon fiber transparent dome, either accumulative dirt or condensation deteriorates the capacity of reinforce star position in the field of vision. After dividing the sample of 21,775 Starcam records without Moon into quintiles, it is considered a value of 12.24 for variable wrong reading (W), 75 for regular reading (R), 88.57 for good reading (G), and 91.67 for very good reading (VG).

2.12 Moon's influence on Starcam

As mentioned earlier, bright Moon's presence in the sky causes a success reduction at searching stars in both cameras and more emphasized in Starcam than in CASANDRA. It is concluded (from other logical way) for an overall sample of 74,801 records achieved in the corresponding Moon nights. The idea is to modify ends from the membership functions of the variable wrong record, regular record, good record, and very good record depending on zenithal distance of the Moon and percentage of illuminated Moon fraction. For that purpose, coefficients to multiply ends functions have been searched. Keep in mind that a percentage of stars found of, for example, 56%, would be bad for a moonless night but very acceptable with bright and high Moon in the sky.

TABLE 2 Percentage of stars.

Illuminated Moon disc	70th percentile
10%	86.02
20%	82.05
30%	79.31
40%	79.47
50%	75.61
60%	77.27
70%	66.67
80%	65.71
90%	60.53
100%	29.41

On a first statistical analysis, percentile 70 is taken as reference. Table 2 displays the average percentage of success at percentile 70 for different values of illuminated Moon fraction.

It can be seen, as expected, that success index decreases as moon brightness increases (all possible zenithal distances for Moon nights are included in the table). The reason was shown earlier: As illuminated Moon fraction increases, there is less sky area to explore, and it is more difficult to find stars with a brighter

TABLE 3 Percentages of stars found as a function of the lunar disk and zenith distance from the Moon.

Disc between 70% and 80%		Disc between 80% and 90%		Disc between 90% and 100%	
Zenithal distance	**Stars**	**Zenithal distance**	**Stars**	**Zenithal distance**	**Stars**
85	81.53%	85	80.39%	85	70.55%
80	65.79%	80	63.61%	80	50.00%
70	63.46%	70	48.65%	70	50.00%
60	69.70%	60	64.86%	60	24.39%
50	70.27%	50	73.44%	50	25.71%
40	58.04%	40	57.69%	40	31.54%
30	54.24%	30	42.86%	30	22.22%
20	42.31%	20	43.75%	20	08.78%

TABLE 4 Coefficients obtained by statistical methods.

D	*z*	Coefficient
70	25	0.8710
70	30	0.8710
70	35	0.8710
70	40	0.8798
70	45	0.8710
70	50	0.8094
70	55	0.8064
70	60	0.8710
70	65	0.8627
70	70	0.8710
70	75	0.9231
70	80	0.9355

sky. Table 3 shows average percentage of accuracy for zenithal distances between 20 and 85 deg.

It is noticed an expected behavior: percentage of success increases as Moon is far from zenith. However, there will be nights with high percentage of illuminated Moon fraction, but with the others, conditions favorable for opening the dome. As CASANDRA is more efficient in recognizing stars than Starcam, the inference method will keep the dome open even though Starcam records have very low percentage of success.

The collection of multiplier coefficient to endpoints of the membership functions have been made after comparing Starcam records from all summer nights from 2019, 2020, and 2021. Summer was chosen because of the usual lack of clouds during the season, and then it has almost no influence in the comparison of efficiency in star recognition with or without Moon.

In moonless nights, it has been calculated a value of 86.11% success at percentile 80. In nights with Moon, the corresponding values of percentile 80 were located at zenithal distances between 25 and 85 deg and between 50 and 100% of percentage of illuminated Moon fraction. Percentages of illuminated Moon fraction lower than 50% were discarded because efficiency in star counting was very similar to the one at moonless nights. Remember that near new Moon, the satellite has low height over the horizon, and it stays at the sky for a short time, then it barely interferes with the count. See Table 4 for an example.

The coefficient has been obtained by dividing the percentile 80 of moonless nights by the corresponding percentile of nights with 70% $\leq D <$ 80% and zenithal distances range $z \in [25, 80]$. For each value $D \in \{50, 60, 70, 80, 100\%\}$, a quadratic function has been calculated by least-squares adjustment function:

$$\text{Coefficient}\,(D, z) = a + bD + cz + dD^2 + ez^2 + fDz. \quad (2)$$

So, if membership function for record VH at moonless night was

$$\text{recordVH}\,(\text{x}) = \begin{cases} 0 & \text{if } x < 89.67, \\ \dfrac{x - 89.67}{93.67 - 89.67} & \text{if } 89.67 \leq x < 93.67, \\ 1 & \text{if } x \geq 93.67, \end{cases}$$

including Moon's influence, it would be expressed as:

$$\text{recordVH}\,(x, D, z) = \begin{cases} 0 & \text{if } x < 89.67 \times k, \\ \dfrac{x - 89.67 \times k}{(93.67 - 89.67) \times k} & \text{if } 89.67 \times k \leq x < 93.67 \times k \\ 1 & \text{if } x \geq 93.67 \times k, \end{cases}$$

with k = Coefficient (D, z). See Figure 5A.

3 Mamdani inference method

By using the variables described in the previous section, the conditional inference rules if. . .then. . .kind were written. For inference rule consequent, it has determined output values very inconvenient, slight convenient, convenient, and very convenient, with membership functions in Figure 5B.

After variable analysis, the following rules were written:

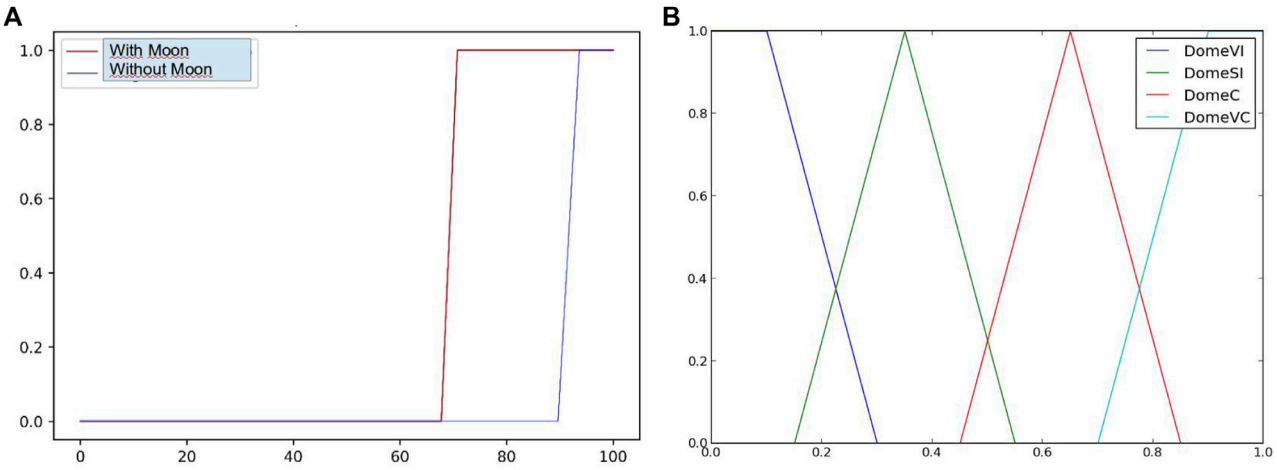


FIGURE 5
Variation of the membership function due to the influence of the Moon **(A)**. Degree of convenience of opening dome **(B)**.

1. If rains, then the dome has to remain closed.
2. If humidity, then the dome has to remain closed.
3. If average wind speed is high, then it is slightly convenient to open dome.
4. If average wind speed is very high, then it is very inconvenient to open dome.
5. If pressure variation is very negative, then it is very inconvenient to open dome.
6. If pressure variation is negative, then it is slightly convenient to open dome.
7. If temperature-dew point difference is very low, then it is slightly convenient to open dome.
8. If temperature-dew point difference is high, then it is convenient to open dome.
9. If temperature-dew point difference is very high, then it is very convenient to open dome.
10. If pressure is very low, then it is very inconvenient to open dome.
11. If pressure is high, then it is convenient to open dome.
12. If pressure is very high, then it is very convenient to open dome.
13. If wind direction is close to Southeast and average speed is not low, then it is slightly convenient to open dome.
14. If wind direction is close to North, then it is convenient to open dome.
15. If closed dome and Starcam record is not wrong, then it is convenient to open dome.
16. If Starcam record is very wrong, then it is very inconvenient to open dome.
17. If open dome and CASANDRA record are not wrong, then it is very convenient to open dome.

TABLE 5 Statistics of the degree of convenience of opening dome.

Dome	Total	Media	Sigma
Open	29,411	0.654	0.047
Closed	19,273	0.390	0.137

The non-fuzzy character of rules 1 and 2 implies to modifying the centroid formula in the following sense:

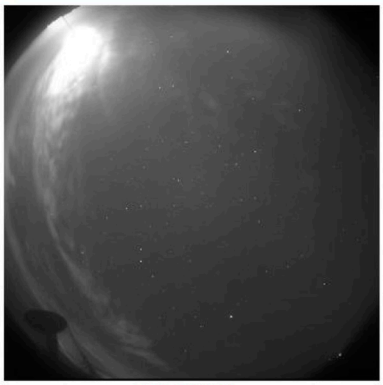
$$\text{centroide} = (1-r)(1-h)\frac{\int \mu_A(x)x}{\int \mu_A(x)}, \tag{3}$$

where r and h are the corresponding values of the membership functions rain and humidity.

The process begins with fuzzification of all input variables, taking their value and establishing membership value to the corresponding fuzzy set. These inputs are applied to the previous 17 inference rules by using, if necessary, a modifier or, if there is more than one antecedent, by using the appropriate operator $\wedge$ (minimum) or $\vee$ (maximum). The result is applied to the consequent by using the minimum method, and all outputs are unified by using the maximum operator to obtain a single set for each output variable. As interest is in a numerical value and not in a set, the centroid method at interval [0, 1] will be applied to the set, and the membership value to open the dome will be obtained.

For inference rule consequent, output values will be determined, which will be different types of convenience to open dome, very inconvenient, slight convenient, convenient, and very convenient.

Date/hour: 2020-05-01 02:20:11
State of the dome: open Degree of convenience: 0.626

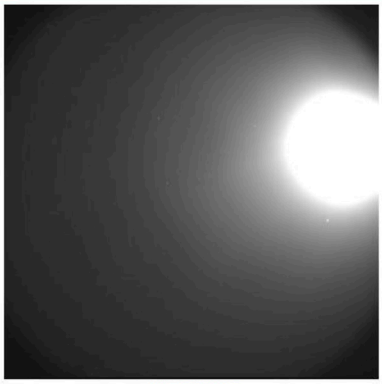


CASANDRA

Starcam

Presure	29,98	Rain	0
Temperature	60,3F	Z Moon	93º,06
Humidity	77,00%	Percentage moon disc	52,7%
Dew point temp. difference	7,20F	Starcam reading	88,89%
Wind speed	5	CASANDRA reading	100%
Wind direction	284º		

Date/hour: 2021-06-27 03:15:18
State of the dome: open Degree of convenience: 0.771



CASANDRA

Starcam

Presure	29,76	Rain	0
Temperature	60,0F	Z Moon	60º,04
Humidity	75,00%	Percentage moon disc	92,3%
Dew point temp. difference	7,90F	Starcam reading	56,41%
Wind speed	1	CASANDRA reading	91,43%
Wind direction	260º		

FIGURE 6
Degree of suitability indicates open dome, and the dome was open.

Date/hour: 2020-10-21 20:25:07
State of the dome: closed Degree of convenience: 0.000

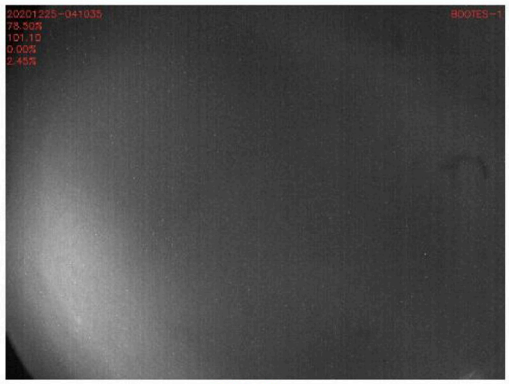
N/A

CASANDRA Starcam

Presure	29,68	Rain	0,42
Temperature	69,6F	Z Moon	78°,69
Humidity	77,00%	Percentage moon disc	32,23%
Dew point temp. difference	7,40F	Starcam reading	13,89%
Wind speed	4	CASANDRA reading	-2
Wind direction	233°		

Date/hour: 2020-12-25 04:10:35
State of the dome: closed Degree of convenience: 0.342

N/A



CASANDRA Starcam

Presure	29,92	Rain	0
Temperature	60,2F	Z Moon	101°,01
Humidity	71,00%	Percentage moon disc	78,50%
Dew point temp. difference	9,30F	Starcam reading	0,00%
Wind speed	6	CASANDRA reading	-2
Wind direction	15°		

FIGURE 7
Degree of suitability indicates close dome, and the dome was closed.

Date/hour: 2021-01-14 02:25:34
State of the dome: closed Degree of convenience: 0.696

N/A

CASANDRA

Starcam

Presure	30,18	Rain	0
Temperature	41,7F	Z Moon	160º,63
Humidity	72,00%	Percentage moon disc	1,10%
Dew point temp. difference	8,30F	Starcam reading	79,07%
Wind speed	2	CASANDRA reading	-2
Wind direction	265º		

Date/hour: 2020-02-18 20:50:29
State of the dome: closed Degree of convenience: 0.633

N/A

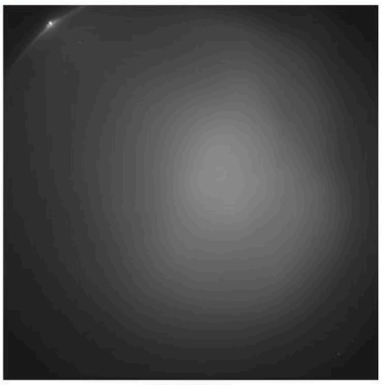
CASANDRA

Starcam

Presure	30,16	Rain	0
Temperature	57,9F	Z Moon	166º,01
Humidity	66,00%	Percentage moon disc	20,80%
Dew point temp. difference	11,20F	Starcam reading	82,93%
Wind speed	4	CASANDRA reading	-2
Wind direction	278º		

FIGURE 8
Dome was closed, but the degree of convenience indicated that it could have remained open.

Date/hour: 2020-12-23 02:30:45
State of the dome: open Degree of convenience: 0.595

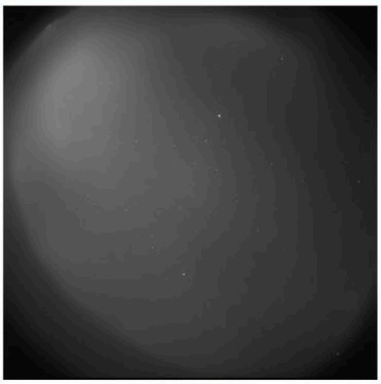


CASANDRA

Starcam

Presure	30,04	Rain	0
Temperature	56,7F	Z Moon	104º,13
Humidity	79,00%	Percentage moon disc	60,70%
Dew point temp. difference	6,4F	Starcam reading	0,00%
Wind speed	6	CASANDRA reading	3,45%
Wind direction	43º		

Date/hour: 2021-04-12 03:10:19
State of the dome: open Degree of convenience: 0.460



CASANDRA

Starcam

Presure	29,87	Rain	0
Temperature	58,4F	Z Moon	124º,16
Humidity	79,00%	Percentage moon disc	0,10%
Dew point temp. difference	6,4F	Starcam reading	5,56%
Wind speed	4	CASANDRA reading	4,76%
Wind direction	344º		

FIGURE 9
Dome was open, but the degree of convenience indicated that it could have remained closed.

4 Testing the fuzzy inference model

For the testing of the inference model, we have built a database of 48,684 records containing the information described in Section 2 regarding the weather station data and the readings from the CASANDRA and Starcam cameras. These data cover a time period from 1 January 2020 to 29 December 2021. Of these 48,684 measures, the dome was open in 29,411 (60.41%) and closed in 19,273 (39.59%).

Table 5 shows statistics on the degrees of convenience of opening the dome provided by the modified Mamdani method of Eq. 3.

As expected, we observed a clear difference between the degrees of convenience obtained with open dome and closed dome. Moreover, the standard deviations are small, indicating a logical clustering.

It should be noted that in the almost 2 years for which we have used data, the dome has been closed at times due to breakdowns or maintenance work. However, even in these circumstances, it is interesting to know how the inference engine would have responded from the meteorological data and the readings from the external camera (Starcam).

For this study and in view of the statistics, we have considered that the dome could be opened from a degree of convenience greater than or equal to the mean minus one standard deviation, that is, $0.607 = 0.654 - 0.047$. By this criterion, on 2,059 occasions, the dome was closed when the model indicates that it could have been opened.

Figure 6 shows two situations (one with Moon and one without Moon) where the model indicated to open the dome, and it was indeed open. The bright light on the left edge of the images is an aerial beacon that was subsequently shielded at our request to avoid stray lights.

Figure 7 shows two cases where the degree of convenience advises to close the dome with the dome being closed. In the first one, there was a cumulative rainfall in the last hour of 0.42 inches, which forces the inference engine to return a 0 degree of convenience to open dome. The second was completely cloudy with a Starcam reading of 0% stars found.

Figure 8 shows two examples where the dome was closed, but the Mamdani engine indicates that it should have been opened.

Finally, Figure 9 shows two cases where the dome was open, yet the method of inference indicates that it should have been closed.

5 Improvements for the future and pending work

The tests described in the previous section have been performed with data collected from the past and applying the inference method to it. Now, it is time to perform the tests with data obtained in real time. It has not been possible to tackle this task at the time of writing this article due to a recent malfunction of the weather station, which returns absurd data for humidity and outside temperature to the dome and has yet to be corrected.

Our intention is to export this method tested at the BOOTES-1 observatory to the rest of the BOOTES observatories, which are equipped with the same instrumentation. In each of them, we must then carry out the corresponding statistical studies in order to determine the corresponding membership functions.

We assume that some adjustments to the endpoints of the membership functions of the variables will result from real-time experimentation.

6 Conclusion

The very first task of an astronomical observatory before starting its observing work is to check whether the conditions are right for opening the dome. In networks of robotic observatories around the world, dusk occurs over a too wide a range of times to be monitored by a human. Thus, an algorithm has to be developed to make that decision. In this work, we have experimented on the suitability of the tools that fuzzy logic offers in this sense, which has been proven in the absence of completing this experimentation in real time.

Data availability statement

The raw data supporting the conclusion of this article will be made available by the authors upon request, without undue reservation.

Author contributions

All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication. Under the leadership ship of AC, both GL and RC, optimized the method, whereas AC-T, EF-G, CP, AR, and SC-C, provided the data and tested the method.

Acknowledgments

AC-T acknowledges support from the Spanish Ministry Project PID 2020-118491GB-I00, Junta de Andalucía Project P20-01068, and the "Center of Excellence Severo Ochoa" award for the Instituto de Astrofísica de Andalucía (SEV-2017-0709).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

References

Agboola, A. H., Gabriel, A. J., Aiyu, E. O., and Alese, B. K. (2013). Development of a fuzzy logic based rainfall prediction model. *Fed. Univ. Technol.* 3 (4), 427–433. P.M.B. 704, Akure, Ondo State, Nigeria.

Akerlof, C. W., Balsano, R., Barthelmy, S., Bloch, J., Butterworth, P., Casperson, D., et al. (1999). Observation of contemporaneous optical radiation from a γ-ray burst *Nature*, 398, 400–402. doi:10.1038/18837

Barthelmy, S. D., Cline, T. L., Butterworth, P., Kippen, R. M., Briggs, M. S., and Connaughton, V. (1998). The grb coordinates network (gcn): A status report, fourth huntsville gamma-ray burst symposium. *AIP Conf. Proc.* 428, 99–103. doi:10.1063/1.1361631

Castro-Tirado, A. J., Jelínek, M., Gorosabel, J., Kubánek, P., Cunniffe, R., and Guziy, S. (2012). Building the BOOTES world-wide network of robotic telescopes. *Astronomical Soc. India Conf. Ser.* 7, 313–320.

Lipunov, V. M., Vladimirov, V. V., Gorbovskoi, E. S., Kuznetsov, A. S., Zimnukhov, D. S., Balanutsa, P. V., et al. (2019). The concept of a multi-functional astronomy complex and dynamically integrated database applied to multi-channel observations with the MASTER global network. *Astron. Rep.* 63, 293–309. doi:10.1134/s1063772919040073and others

Mamdani, E. H., and Assilian, S. (1975). An experiment in linguistic synthesis with a fuzzy logic controller. *Int. J. Man-Machine Stud.* 7, 1–13. doi:10.1016/s0020-7373(75)80002-2

Manish, K. S., Harkamal, D. K., and Parag, N. (2019). Rain prediction using fuzzy logic. *Int. J. Eng. Adv. Technol. (IJEAT)* 9 (1), 2796–2799.

frontiers | Frontiers in Astronomy and Space Sciences

ORIGINAL RESEARCH
published: 22 August 2022
doi: 10.3389/fspas.2022.896040

The Tian Ma 65-m Telescope Automatic Early Warning System

Shang-Guan Wei-Hua[1,2], *Zhao Rong-Bing*[1]*, *Zhang Dong*[1,2] *and Zhang Chu Yuan*[3]

[1]*Shanghai Astronomical Observatory, Chinese Academy of Sciences, Shanghai, China,* [2]*University of Chinese Academy of Sciences, Beijing, China,* [3]*Xi'an University of Posts and Telecommunications, Xi'an, China*

The Tian Ma Radio Telescope (TMRT), which is mainly used for deep space exploration and radio astronomy observations, is the largest fully steerable radio telescope in Asia. For promoting the automation of the telescope, an automatic early warning system is designed and implemented. The system can conveniently aggregate heterogeneous sensor data, make use of established strategies to implement an alert system, and send real-time alarms through multiple channels, which is helpful to promote unmanned operation. In addition, we adopt DevOps (a compound of development (Dev) and operations (Ops) which means end-to-end automation in software development and delivery) to simplify the development, upgrading, and maintenance of the telescope automatic early warning system.

Keywords: TMRT, automation, real-time alarms, DevOps, automatic early-warning system

OPEN ACCESS

Edited by:
Frederic Victor Hessman, Georg-August-Universität Göttingen, Germany

Reviewed by:
Aletha De Witt, Hartebeesthoek Radio Astronomy Observatory (HartRAO), South Africa
Tim-Oliver Husser, University of Göttingen, Germany

***Correspondence:**
Zhao Rong-Bing
zhaorb@shao.ac.cn

Specialty section:
This article was submitted to Astronomical Instrumentation, a section of the journal Frontiers in Astronomy and Space Sciences

Received: *14 March 2022*
Accepted: *23 June 2022*
Published: *22 August 2022*

Citation:
Wei-Hua S-G, Rong-Bing Z, Dong Z and Chu Yuan Z (2022) The Tian Ma 65-m Telescope Automatic Early Warning System. Front. Astron. Space Sci. 9:896040. doi: 10.3389/fspas.2022.896040

1 INTRODUCTION

The Tian Ma Radio Telescope (TMRT) in Shanghai has a primary reflector of 65 m diameter with a shaped Cassegrain configuration. It has advanced equipment such as an active surface control system, an 8-band low-noise receiver system covering 1–50 GHz, VLBI high-speed data acquisition systems, and high-stability hydrogen atomic clocks. It is available for deep space exploration tasks such as orbit determination, for radio astronomy single-dish and Very Long Baseline Interferometry (VLBI) observations, and for geodetic VLBI observations. The TMRT has already achieved many of its engineering goals and, as a result, has provided many world-class scientific results. First, the TMRT, as a powerful unit of the Chinese VLBI Network (CVN), has undertaken many deep space exploration tasks, such as VLBI observations in the Chang'e-3 orbit determination (Liu et al., 2015). Second, the TMRT conducts many astronomical observations for scientific research purposes. Its operating frequency range covers many important astronomical spectrum lines, such as NH3, CCS, and HC3N (Xie J et al., 2021; Zhang et al., 2021). Pulsars are also important targets of the TMRT, such as the comprehensive pulse profile study of 71 pulsars (Zhao R.-S et al., 2019). Third, the TMRT also carries out many geodetic VLBI observations. For example, the TMRT, as one component of the International VLBI Service for Geodesy and Astrometry (IVS) network stations, participated in six IVS sessions in 2019 and 2020 (Xie B et al., 2021).

To meet the above mentioned scientific and engineering goals, the TMRT has been operating for almost 24 h a day. Manpower on 24-h shifts is a direct way to monitor the operation of the TMRT, but with the progress of science and technology, more and more stations have researched and applied remote and autonomous operations and status monitoring (Neidhardt, 2017). More remote control software are also being developed for telescopes, such as Client Graphics User Interface Library Tools (CGLT), E-control, and Jmonan (Neidhardt et al., 2010; Ruztort et al., 2012; Zhao D et al., 2019). The remote and autonomous control software is also applied to the TMRT.

The Tian Ma 65-m Telescope Automatic Early Warning System (TAEWS) can provide a safety guarantee for the TMRT during autonomous operations through real-time operational status early

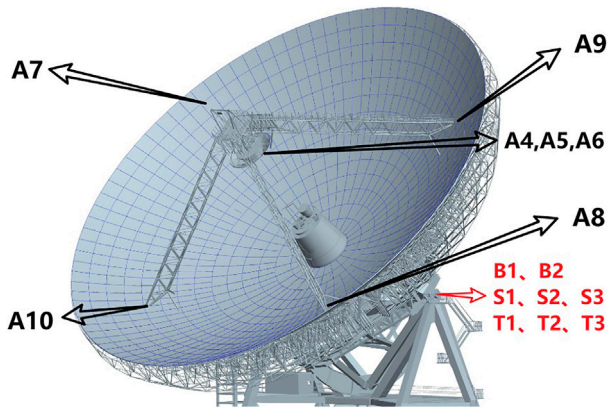


FIGURE 1 | Figure shows the model diagram of the TMRT. The black text "A" marks the vibration sensor position, the red "B" marks the range sensor position, the red "S" marks the stress sensor position, the red "T" marks the temperature sensor position. And the numbers are their numbers, such as "A1" means vibration sensor number 1.

warning detection for the TMRT. Currently, we have placed some sensors in different locations of the telescope. The TAEWS aggregates these heterogeneous sensor data and then detects in real time whether the TMRT is operating beyond the preset alarm values. It uses push notifications and alert notifications will be sent through multiple channels in case of abnormalities. Engineers can also download historical data from the system after the event to analyze the cause of the error in the TMRT.

Next, we will introduce the four parts of the TAEWS: **Section 2**) The monitoring items of the TMRT; **Section 3**) Aggregation of heterogeneous data; **Section 4**) Data visualization, detection and alarm; **Section 5**) Application of DevOps (a compound of development (Dev), and operations (Ops) meaning end-to-end automation in software development and delivery) in the system.

2 THE MONITORING ITEMS OF THE TMRT

At present, the monitoring items supported by the system mainly include motor current detection, antenna rotation detection, bearing stress detection, and antenna mechanical structure problem detection. As shown in **Figure 1**, sensors, which include vibration sensors, temperature sensors, stress sensors, and ranging sensors, are distributed in different locations on the antenna. Each of them will be described next.

The TMRT weighs about 2,700 tons and the weight of its reflector is mainly borne by bearings. As shown in **Figure 1**, stress sensors, temperature sensors, and distance sensors are added to the bearing end. These sensors detect stress changes in the reflector to the bearing. The TMRT has eight azimuth (AZ) rotation motors and four elevation axis rotation motors. The rotation of the antenna is driven by a motor, which detects changes in motor current when the motor is driving to determine if there is any abnormal resistance during operation. Each motor of the TMRT is equipped with at least one current sensor. Besides this, the rotation speed and acceleration of the AZ and pitch axes can also be detected. When observing at higher frequencies, the vibration caused by the servo drive systems will be significant, so we have placed vibration sensors at the primary reflector, sub-reflector, and feeder (Brandt, 2000). We can detect whether there is any problem with the mechanical structure of the antenna and if the servo control is normal.

To enable an early warning for the above-mentioned detection items (sensors), we have designed the TAEWS. The structure diagram of this system is shown in **Figure 2**. As shown in the structure diagram, the TAEWS is divided into three parts: the environment, the services, and the user interface. We deploy the services part on the Docker Engine[1] (environment part), which facilitates our development using DevOps. The details of DevOps will be introduced in **Section 5**. The services in the structure diagram include Data Sampler, Influxdb[2], Telegraf[3], Grafana[4], Grafana-Images-Renderer[5], Portainer[6], and source code management software.

Data Sampler is used to collect data from the sensor software and hardware and print it out on the system console in a comma separated value (CSV) format.

Influxdb is a time-series database for storing sensor data. The use of a time-series database will also optimize our storage of time-stamped sensor data.

Telegraf is used to store Data Sampler output sensor data to the Influxdb database and to keep the Data Sampler service running. For example, when the sensor network is disconnected, the Data Sampler may crash and exit. But the Telegraf will restart the Data Sampler by timed attempts, and we set the time interval to 1 min. So, after the sensor network is restored, the Data Sampler will also be restored after 1 minute.

Grafana provides visualization, monitoring, and alerting capabilities primarily for data in Influxdb. We have designed panels to visualize the values of the sensors on the TMRT, as detailed in **Section 4**.

Grafana-Images-Renderer is used to help Grafana improve the alert messages by replacing the values in the alert emails with screenshots of the visualization panels. The pictures of the visualization panel will be more useful for engineers and observers to understand the situation than the alarm values.

Portainer is the service management platform of the TAEWS. As a service management tool, Portainer makes it convenient to deploy services, create services, edit service configurations, and view service status.

[1]Docker Inc. (2020). Docker Engine overview. https://docs.docker.com/engine/ [Accessed 29 May 2022].

[2]InfluxData. (2022). InfluxDB. https://github.com/influxdata/influxdb [Accessed 29 May 2022].

[3]InfluxData. (2022). Telegraf. https://github.com/influxdata/telegraf [Accessed 29 May 2022].

[4]Grafana Labs. (2022). Grafana. https://github.com/grafana/grafana [Accessed 29 May 2022].

[5]Grafana Labs. (2022). grafana-image-renderer. https://github.com/grafana/grafana-image-renderer [Accessed 29 May 2022].

[6]Portainer.io. (2022). Portainer. https://github.com/portainer/portainer [Accessed 29 May 2022].

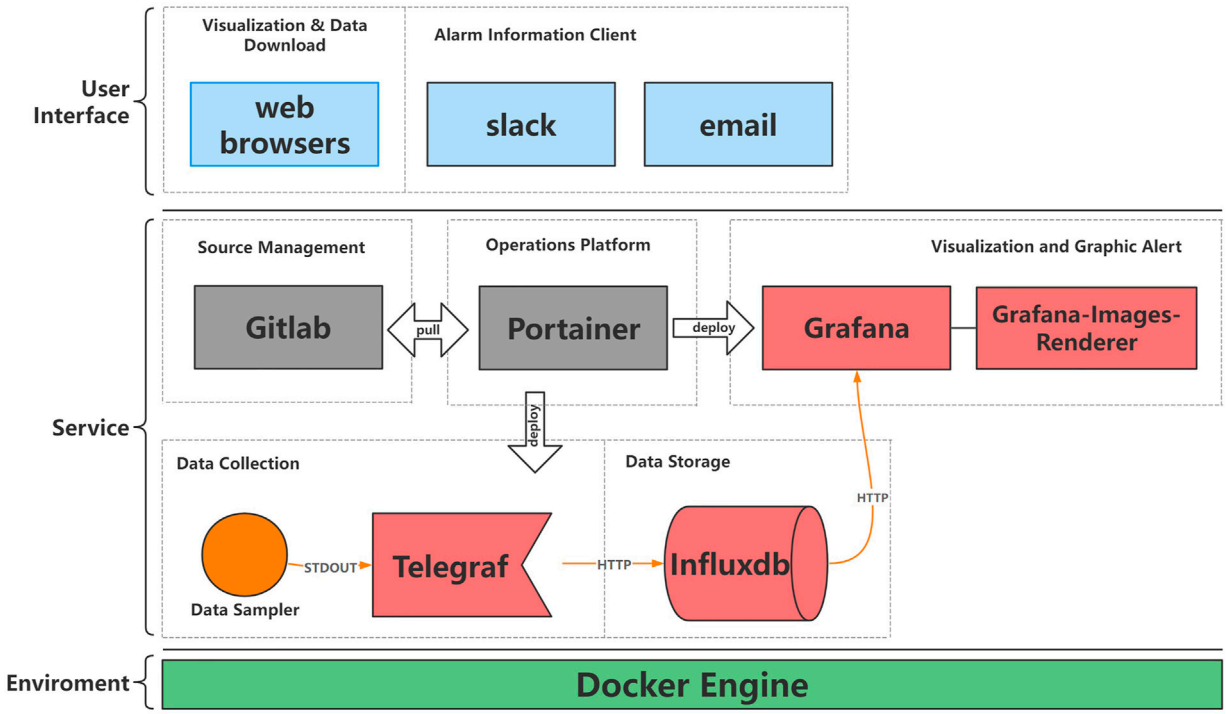


FIGURE 2 | Figure shows that the architecture of the TAEWS is divided into three parts: environment, services, and user interface. All services of the TAEWS are deployed on the Docker Engine. Services mainly include Data Sampler, Telegraf, Influxdb, Grafana, Grafana-Images-Renderer, Portainer, and GitLab. Data Sampler and Telegraf are used for data collection, Influxdb for the time-series database, Portainer for container deployment, GitLab for source code management, Grafana and Grafana Render for visualization and graphic alert. User interfaces include web browsers, Slack, and email. Users can access visualization pages through browsers and push alerts through email, Slack, etc.

GitLab[7] is used to manage service orchestration code and the data sampler code, and to implement DevOps. More details are provided in **Section 5**.

The main TAEWS user interfaces are through Slack[8], email, and a web browser. Engineers and observers can receive alert messages from Grafana via Slack and email. They have cross-platform access to real-time graphs of sensor data on Grafana via a web browser.

3 AGGREGATION OF HETEROGENEOUS DATA

Next, we need to aggregate the data from these sensors. Each sensor is equipped with a Data Sampler, a Telegraf, and an Influxdb service called the Telegraf-Influxdb data stream. The Telegraf-Influxdb data stream structure is used to implement persistent sensor data collection: the Data Sampler collects sensor data and prints it to Telegraf in CSV format, and Telegraf writes the collected data to Influxdb. The data transfer differences of the heterogeneous sensors are solved by different Data Samplers. Since the Telegraf-Influxdb data stream structure is based on Telegraf and Influxdb, only the Data Sampler is programmed.

First, the communication interface, data sampling rate, and data format of heterogeneous sensors may be different. Some manufacturers of sensors on the TMRT have customized communication interface standards specific to their sensors, which may require us to use the manufacturer's software, protocols, development languages, or dependency packages. For example, the bearing sensors on the TMRT use a customized Modbus protocol that cannot be communicated with directly by existing open-source tools. The data storage format of the vibration sensor REFTEK130 on the TMRT is PAS. The python library ObsPy (Beyreuther et al., 2010) and the commercial software of Reftek Systems Inc. are the only tools found that can decode the PAS format.

Transferring data to Telegraf *via* a standard output stream is a convenient way to solve the problem. First, the Data Sampler outputs data to the system in CSV format. Second, Telegraf calls the Data Sampler program through a child process and gathers the system output stream from the child process. Since the programming language can be printed, there are many programming language options for The Data Sampler. This helps to cope with sensor hardware and software interfaces from various vendors compared to a single programming language. And because Telegraf can cache data and maintain

[7]GitLab B.V. (2021). GitLab. https://gitlab.com/gitlab-org/gitlab [Accessed 29 May 2022].

[8]Slack Technologies, LLC, Salesforce. (2022). Where Work Happens. https://slack.com/[Accessed 29 May 2022].

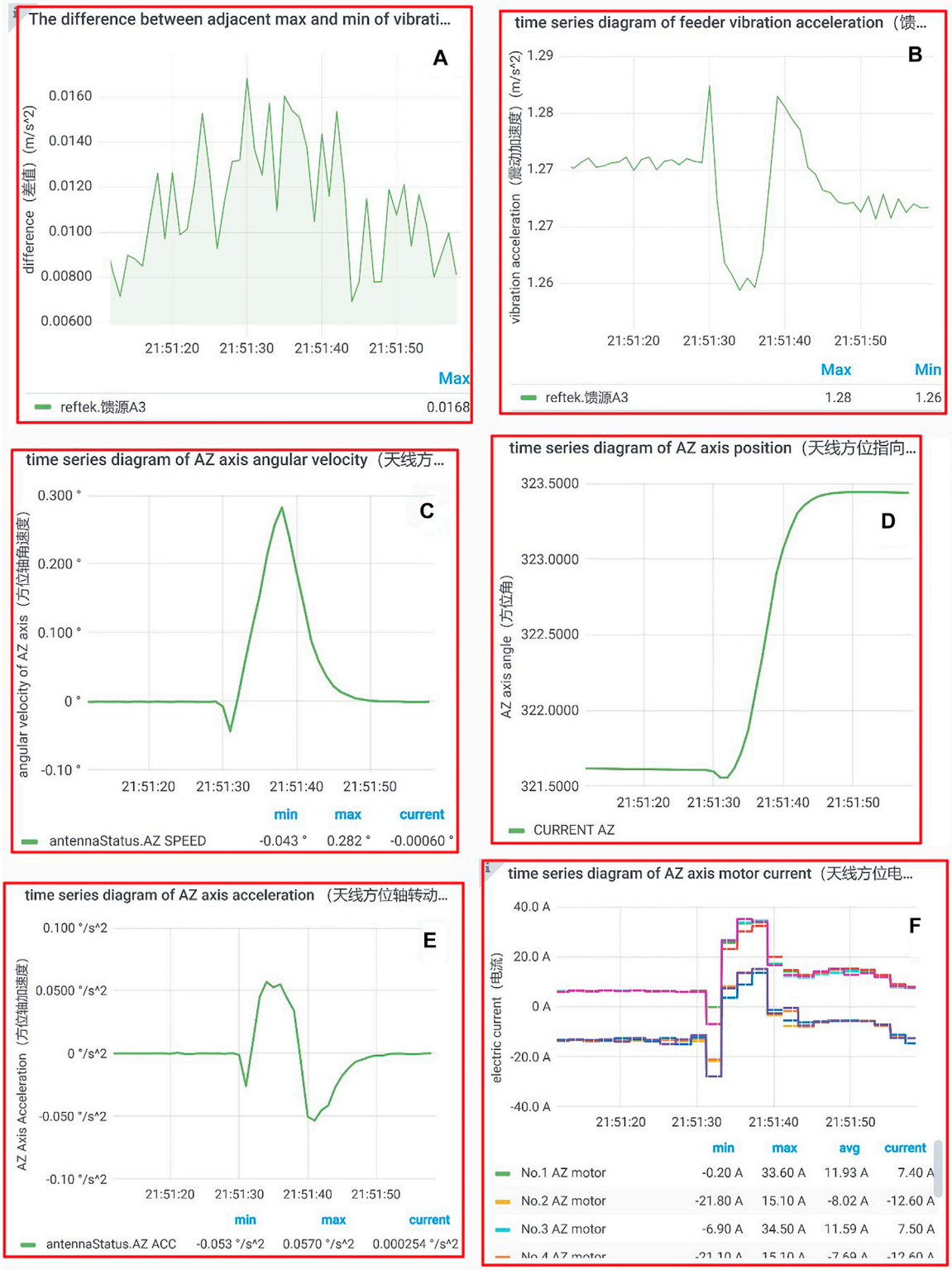


FIGURE 3 | This is a visualization page in TAEWS. **(A)** is the difference between adjacent max and min of vibration acceleration, **(B)** is the time-series diagram of feeder vibration acceleration, **(C)** is the time-series diagram of AZ axis angular velocity, **(D)** is the time-series diagram of AZ axis position, **(E)** is the time-series diagram of AZ axis acceleration, **(F)** is the time-series diagram of AZ axis motor current.

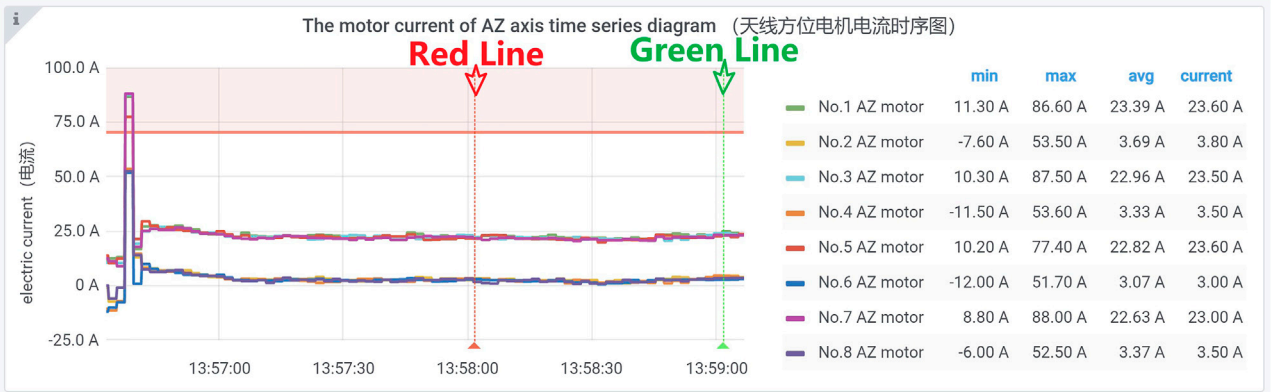


FIGURE 4 | This is a sample timing diagram for an antenna AZ motor current alarm. In the figure, the motor current curve has broken through the red alarm area above, and then Grafana will mark 13:58:00 with a red line, which shows that there was a voltage anomaly within 2 minutes. The green line at 13:59:00 shows that the abnormality did not appear after the time point marked by the red line.

the data sampler, the data sampler basically just converts the data transferred from the sensor hardware or software into CSV format and prints it to the system console.

Second, since telescope anomaly monitoring is still under research, we need to ensure that the system is scalable, i.e., that it is compatible with adding new detections but also with removing the old ones. Therefore, we make each sensor's Data Sampler and Telegraf into a container, which makes the Data Sampler highly cohesive with the sensor, and coupling with the database is resolved through Telegraf. The data collection container is connected to the database, ensuring that there is no dependency between Data Samplers.

4 DATA VISUALIZATION, DETECTION, AND ALERTS

The user interface of Grafana allows for customized visual charts, customized alert rules, and customized message channels. **Figure 3** shows some graphs we designed. For example (E) in **Figure 3** is used to visualize the AZ acceleration of the TMRT. The maximum and minimum values of the AZ acceleration are shown at the bottom of the graph. While sensor data is continuously recorded through the Telegraf-Influxdb data stream structure, Grafana can provide real-time visualization capabilities and real-time alerting capabilities for telescope status. For example, we visualize the elevation angle of the antenna, and then set the alert rule to notify the TMRT observer when the antenna elevation angle is below 20° for 10 min.

In **Figure 3**, we can see that the values include antenna rotation speed, antenna pointing position, antenna rotation acceleration, antenna motor current, antenna bearing stress, and antenna vibration acceleration. Changing the step size to adjust the time interval at which the data is displayed makes it easier to understand long-term changes and short-term changes in the data. Note that the minimum time interval of the data displayed by the Grafana is limited by the sampling rate of the data print from the data sampler. Grafana also supports downloading data directly from charts.

If an alarm occurs, a red alarm line will appear on its chart, as shown in **Figure 4**. At present, the alarm of TMRT mainly uses simple and complex threshold values to set a reasonable interval for necessary data items for real-time monitoring. When the value exceeds the interval, the system will alert the message. The simple thresholds are set based on unprocessed data. For example, through the experience of daily maintenance, we set the upper limit of the AZ voltage value as 70A, and when the voltage exceeds the range, abnormal marks will appear, as shown in **Figure 4**. The complex thresholds are set by performing some simple operations on the data. For example, the threshold for vibration monitoring is the adjacent difference and standard deviation of normalized vibration data. The mail alert, shown in **Figure 5**, is a practical example of adjacent differences.

As shown in **Figure 5**, if an alert subscription is set, a push message will be sent through Slack and email. First, a time-series graph is added to the email alert message. The original alert email message has only anomalous data values, which is what the email section in **Figure 5** looks like without the time-series graph in the blue box. The Grafana-Images-Renderer service is used to draw Grafana graphs and export them as PNG format images, which Grafana then uses to draw time-series data charts to add to the emails. Next, in order for the time-series graph to show the situation before and after the occurrence of the alarm, we postpone the alarm time backward with an acceptable delay. As in **Figure 5**, the values that exceed the red alert area are located 1 min before.

5 DEVOPS IN TAEWS

DevOps speeds the delivery of higher quality software by combining and automating the work of software development and IT operations teams. As shown in **Figure 6**, We apply the DevOps method to development in the Docker environment to further improve the speed of software iteration and the convenience of service deployment.

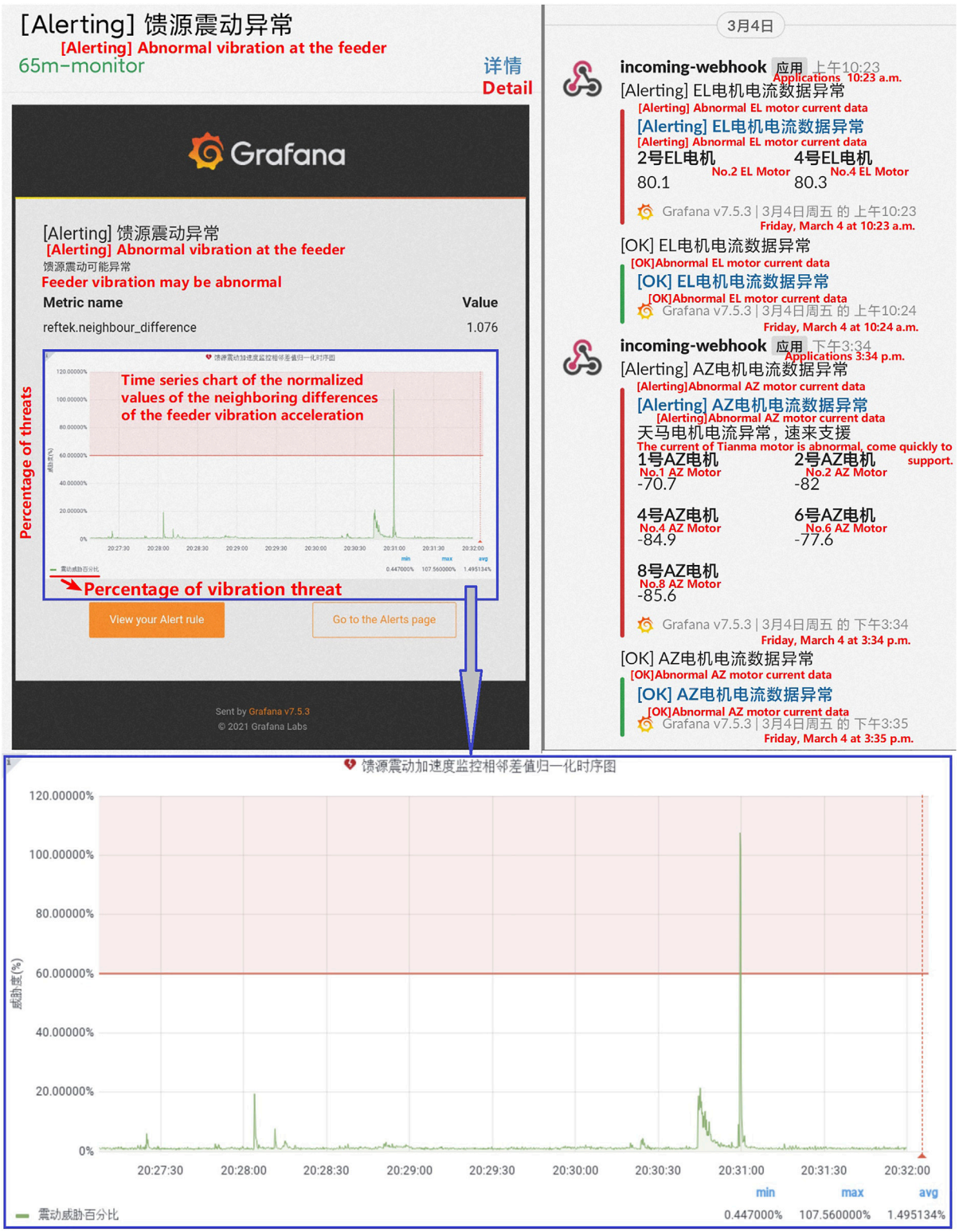


FIGURE 5 | This is a screenshot of the system alert push message. The top-left part is the email message interface, which includes alert values, alert items, value charts, and some alert messages. The bottom half of the figure, which is marked 'C', is a chart detailing the data for this alert. The top-right part is the message interface on the Slack app, which includes alert values, alert items, and some alert messages. The marker "A" and marker "B" are examples of TMRT alerts.

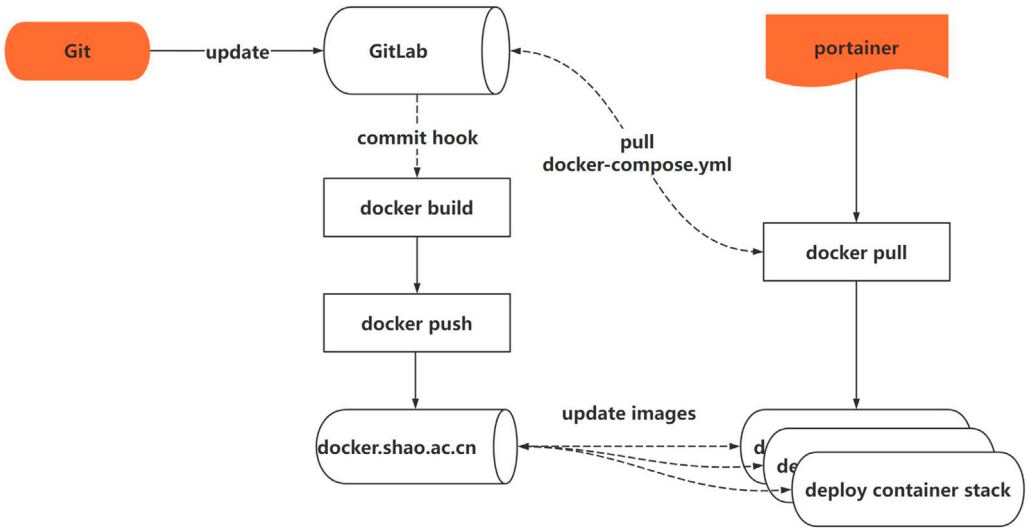


FIGURE 6 | We use Git to update code to the source code management platform such as GitLab, then the platform performs container building, container pushing, and so on through hooks. Later, Portainer automatically pulls the latest service image through the container registry.

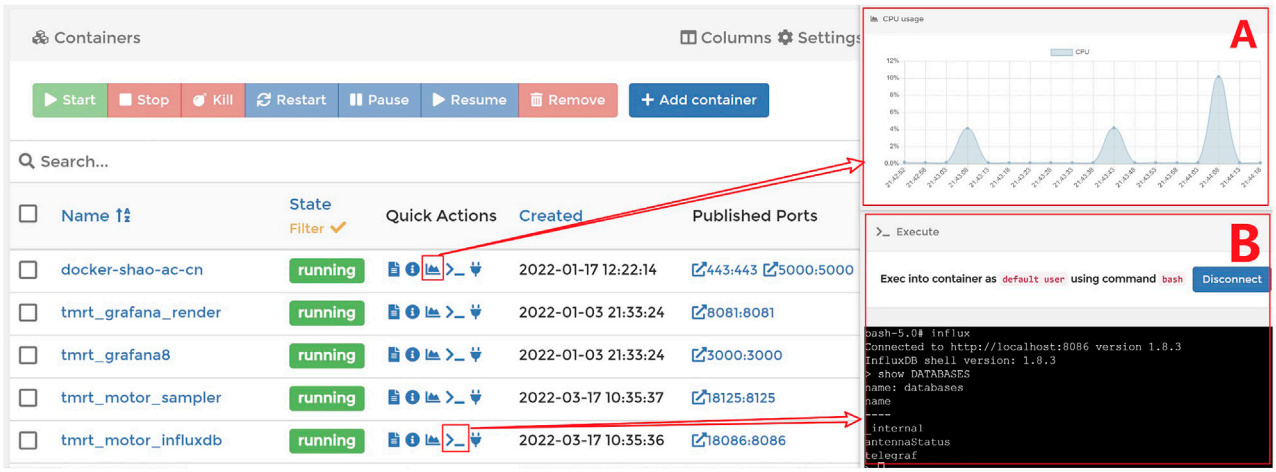


FIGURE 7 | This is an example of using the Portainer to manage the TAEWS system. The information of all the services, such as current running status, creation time, and port occupancy, is clearly visible on the page, and operations such as logging, terminal, and status view are available in the Quick Actions column.

During the development of TAEWS, GitLab manages two types of projects. The first type is the Data Sampler code. The Data Sampler projects include the dockerfile code used to build containers. When the code is submitted to the code management platform, the Data Sampler container is automatically built and submitted to shao. docker.ac.cn, a private container image repository on the intranet. The second type is the TAEWS container orchestration code. This code is mainly the "docker-compose.yml" file and some environment variable files. The environment variable files record the environment variable settings needed for each service. For example, the name of the database such as "vibration_data". The "docker-compose.yml" file is used for the Portainer to deploy containers. For example, the Grafana container maps port 3,000 inside the container to port 80 on the host. The Portainer automatically pulls the required images from docker. shao.ac.cn and deploys the TAEWS on the Docker engine. These approaches enable the continuous integration of the system, which facilitates the development of the system.

In addition to using the Portainer to create and deploy services, we also use the Portainer to complete the maintenance of the services. In **Figure 7**, we can clearly see the port status, creation time, and running state of each container. The Quick Actions allow the user to debug and monitor services. For example, we check the current CPU usage of the container image repository, as in part A of

Figure 7. We may also need to open the container's terminal to execute some commands. For example, part B of **Figure 7** to see if the name configuration of the database is normal.

6 CONCLUSION

As an automatic and real-time early warning detection system for the TMRT, the TAEWS achieves the abnormal condition detection of electromechanical current, antenna rotation, bearing stress, and antenna mechanical structure. It implements persistent storage of heterogeneous sensor data, monitoring of threshold rules for the sensor data, and notification of exception messages. We also briefly describe the containerization and DevOps applications of this system, describing the ability to iterate quickly with a small amount of coding. In the future, with the increase of detection items and the improvement of anomaly detection methods, the automatic warning system of the TMRT will be gradually improved.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Material; further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

SW-H: conceptualization, methodology, software, project administration, investigation, formal analysis, validation, and writing—original draft; ZR-B: conceptualization, funding acquisition, resources, supervision, and writing—review and editing; ZD: software and writing—original draft; and ZC-Y: visualization and investigation.

FUNDING

This work was supported by the National SKA Program of China, No. 2020SKA0120104.

REFERENCES

Beyreuther, M., Barsch, R., Krischer, L., Megies, T., Behr, Y., and Wassermann, J. (2010). ObsPy: A Python Toolbox for Seismology. *Seismol. Res. Lett.* 81 (3), 530–533. doi:10.1785/gssrl.81.3.530

Brandt, J. J. (2000). "Controlling the Green Bank Telescope," in *Advanced Telescope and Instrumentation Control Software. Advanced Telescope and Instrumentation Control Software, Proc* (Munich, Germany: SPIE 4009), 96–108. doi:10.1117/12.388381

Liu, Q., Li, P., Zheng, X., Wu, Y., and He, Q. (2015). Analysis of VLBI Observation for Tianma Radio Telescope in Chang'E-3 Orbit Determination. *Sci. Sin.-Phys. Mech. Astron.* 45 (3), 039501. doi:10.1360/sspma2014-00323

Neidhardt, A., Ettl, M., Rottmann, H., Plotz, C., Muhlbauer, M., Hase, H., et al. (2010). "E-Control: First Public Release of Remote Control Software for VLBI Telescopes," in International VLBI Service for Geodesy and Astrometry 2010 General Meeting Proceedings, 8 Jun. 2021, 439–443. Available at: http://ivscc.gsfc.nasa.gov/publications/gm2010/neidhardt2.pdf.

Neidhardt, A. N. J. (2017). "Applied Computer Science for GGOS Observatories," in *Applied Computer Science for GGOS Observatories: Communication, Coordination and Automation of Future Geodetic Infrastructures* (Cham: Springer International Publishing (Springer Textbooks in Earth Sciences, Geography and Environment), 1–12. doi:10.1007/978-3-319-40139-3

Ruztort, C., Hase, H., Zapata, O., and Pedreros Bustos, F. (2012). "Remote Control and Monitoring of VLBI Experiments by Smartphones," in International VLBI Service for Geodesy and Astrometry 2012 General Meeting Proceedings, 8 Jun. 2021, 286–290. Available at: http://ivscc.gsfc.nasa.gov/publications/gm2012/Herrera.pdf.

Xia, B., Liu, Q., and Shen, Z. (2021). Shanghai Station Report for 2019-2020. [online] Available at: https://ivscc.gsfc.nasa.gov/publications/br2019+2020/nsshao.pdf (Accessed Jun 8, 2022).

Xie, J., Fuller, G. A., Li, D., Chen, L., Ren, Z., Wu, J., et al. (2021). The TMRT K Band Observations towards 26 Infrared Dark Clouds: NH3, CCS, and HC3N. *Sci. China Phys. Mech. Astron.* 64 (7), 279511. doi:10.1007/s11433-021-1695-0

Zhang, C., Wu, Y., Liu, X.-C., Tang, M., Li, D., Esimbek, J., et al. (2021). Carbon-chain Molecule Survey toward Four Low-Mass Molecular Outflow Sources. *A&A* 648, A83. doi:10.1051/0004-6361/202039110

Zhao, D., Zhao, R., Jiang, Y., Li, J., Liu, C., and Zhang, H. (2019). "The 13-Meter Radio Telescope Monitor and Control Software," in *Advances in Computational Science and Computing*. Editors N. Xiong, Z. Xiao, Z. Tong, J. Du, L. Wang, and M. Li (Cham: Springer International Publishing (Advances in Intelligent Systems and Computing), 37–44. doi:10.1007/978-3-030-02116-0_5

Zhao, R.-S., Yan, Z., Wu, X.-J., Shen, Z.-Q., Manchester, R. N., Liu, J., et al. (2019). 5.0 GHz TMRT Observations of 71 Pulsars. *ApJ* 874 (1), 64. doi:10.3847/1538-4357/ab05de

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

frontiers | Frontiers in Astronomy and Space Sciences

TYPE Original Research
PUBLISHED 23 August 2022
DOI 10.3389/fspas.2022.897065

OPEN ACCESS

EDITED BY
Frederic Victor Hessman,
Institut für Astrophysik, Georg-August-Universität Göttingen, Germany

REVIEWED BY
Bjoern Poppe,
University of Oldenburg, Germany
Chuanjun Wang,
National Astronomical Observatories (CAS), China

*CORRESPONDENCE
Jian Sun,
sunjian@nao.cas.cn

SPECIALTY SECTION
This article was submitted to Astronomical Instrumentation, a section of the journal Frontiers in Astronomy and Space Sciences

RECEIVED 15 March 2022
ACCEPTED 13 June 2022
PUBLISHED 23 August 2022

CITATION
Sun J, Cheng H-W, Jiang H, Liu J and Zhao Y-Y (2022), A 36 cm robotic optical telescope: Equipment and software.
Front. Astron. Space Sci. 9:897065.
doi: 10.3389/fspas.2022.897065

A 36 cm robotic optical telescope: Equipment and software

Jian Sun[1,2]*, Hao-Wen Cheng[1,2], Hai Jiang[1,2], Jing Liu[1,2] and Yuan-Yuan Zhao[1,2]

[1]National Astronomical Observatories, Chinese Academy of Sciences, Beijing, China, [2]Space Debris Observation and Data Application Center, China National Space Administration, Beijing, China

The paper describes an optical telescope system and control software for robotic observation of space debris. The telescope has a main aperture of 355 mm, adopts the optical design scheme of primary focus with a large field of view, and is equipped with a highly sensitive 4 K sCMOS camera to achieve a large field of view of 2.6° × 2.6°. The telescope is equipped with an environmental monitoring system and a highly reliable dome to ensure the safe operation of the telescope. The control software of the telescope consists of two parts. One part is deployed locally to comprehensively schedule the robotic operation of various equipment of the telescope system, and the other part is deployed remotely to realize the functions of equipment status monitoring, networking scheduling, remote control, and data management. At present, four telescopes have been deployed in Korla, Xinjiang, China to form a telescope array, basically realizing the remote "unattended" observation of space debris.

KEYWORDS

space debris observation, telescope control system, robotic control, ZeroMQ, optical telescope, networking scheduling

1 Introduction

An important way to improve Space Situational Awareness (SSA) is to obtain sufficient, reliable, and timely observation data to support subsequent track correlation, orbit determination, and conjunction assessment (Crowther, 2002; Kennewell and Vo, 2013). Therefore, for decades, the observation technology of optical telescopes has been applied by astronomical researchers to the observation of space debris in order to obtain the orbit and characteristics of them (Seitzer et al., 2004). Moreover, with the development of modern science and technology, the software and hardware of the telescope have been greatly improved. Automatic or robotic methods have substantially enhanced observation efficiency while also reducing the workload of observation assistants. As a result, the cost of using optical telescopes to observe space debris is gradually decreasing, with some low-cost, small, and medium-sized telescopes still playing an important role in space debris observation, and many automatic space debris monitoring networks emerging.

For example, a global network of the 100 optical telescopes at 42 observatories in 18 countries was created by ISON (International Scientific Optical Network), which is primarily designed to observe space debris (mainly for GEO), asteroids, and GRB afterglows (Molotov et al., 2019). The announced amount of space debris in GEO is in the leading position. The FTN (Falcon Telescope Network) has deployed 12 exactly replicated 500 mm telescopes around the world to study artificial satellites and the nearby universe (Chun et al., 2018). Slightly different from ISON's control mode, all telescopes in FTN are remotely and automatically controlled by the central control system located in USAFA (The United States Air Force Academy). Similarly, Park et al. (2018) has developed the OWL-Net (Optical Wide-field patrol Network) in five countries to obtain the orbital information of Korean LEO and GEO satellites. And all telescopes in OWL-Net are identical, operated in a fully robotic manner. In addition, many countries and institutions also have many similar observation networks, which are committed to space debris observation and other application fields, especially time-domain astronomy (Castro-Tirado, 2011; Martone et al., 2019; Dyer et al., 2020).

Our team has also been engaged in the research of SSA, and hopes to use some low-cost telescopes to collect more observation data of space debris. Therefore, this paper will introduce our progress in the telescope software and hardware, and summarizes a robotic telescope system. The system can realize the remote control of multiple telescopes, optimize and formulate observation plans for each telescope based on user demands, and automatically implement them. It can also carry out established processing for some typical emergencies, protect the safety of telescopes, and reduce the dependence on station personnel, reaching the degree of "unattended" as far as possible. Therefore, the second section will introduce the materials and methods to implement the system, and the third section will describe some of its observation results. Then, a discussion of the results will be presented in the fourth section. Finally, the fifth section will describe the conclusions and suggestions for future research.

2 Materials and methods

2.1 Observation site

Weather conditions, observable days, infrastructure, maintenance support, and other aspects must all be considered when selecting a decent astronomical observatory. The Korla observatory (86.2°E, 41.5°N) in China was chosen as our test site. Korla City is dry and rainless all year due to its location on the edge of the Taklimakan Desert, and there are over 300 observable nights throughout the year (the observation night defined here is not a strict astronomical photometric observable night. Stars can be observed at any time after dusk, which is called observable night, even if it is a little cloudy.). The photometric instrument SQM-L obtained a sky light background darker than 19 mag./arc sec^2. Moreover, the observation site has convenient transportation, which is conducive to equipment maintenance and the early experiment.

FIGURE 1
A 36 cm telescope in Korla observatory, deployed in early 2021.

Therefore, we have deployed four sets of 36 cm telescopes at this site to observe space debris and asteroids. One of the telescopes is shown in Figure 1. And Figure 2 shows all the main hardware devices and their connection modes in this observatory. The following section will introduce the software and hardware of the whole system one by one.

2.2 Hardware components

2.2.1 Telescope, mount, camera, and dome

Although program-controlled equipment has been relatively common, there are still few COTS (commercial-off-the-shelf) telescopes that meet the needs for robotic observation of space debris. So, some hardware has been upgraded to fit our needs, allowing the telescope to perform better and adapt to tracking and surveying objects in varied orbits.

The whole telescope system is a passive optical approach to observe space debris. The optical tube of this telescope is improved from the Celestron 36 cm Rowe-Ackermann Schmidt Astrograph (RASA) telescope, which has an aperture of 355 mm, a focal length of 790 mm, and an optical field of view of 4.3°. Based on the original structure, the structural strength is improved, and automatic equipment such as the electric focuser (deploy at the front end of the tube, focusing is realized by moving the camera.) and electric lens cover are added, which makes it more suitable for the application of SSA and other space surveillance applications.

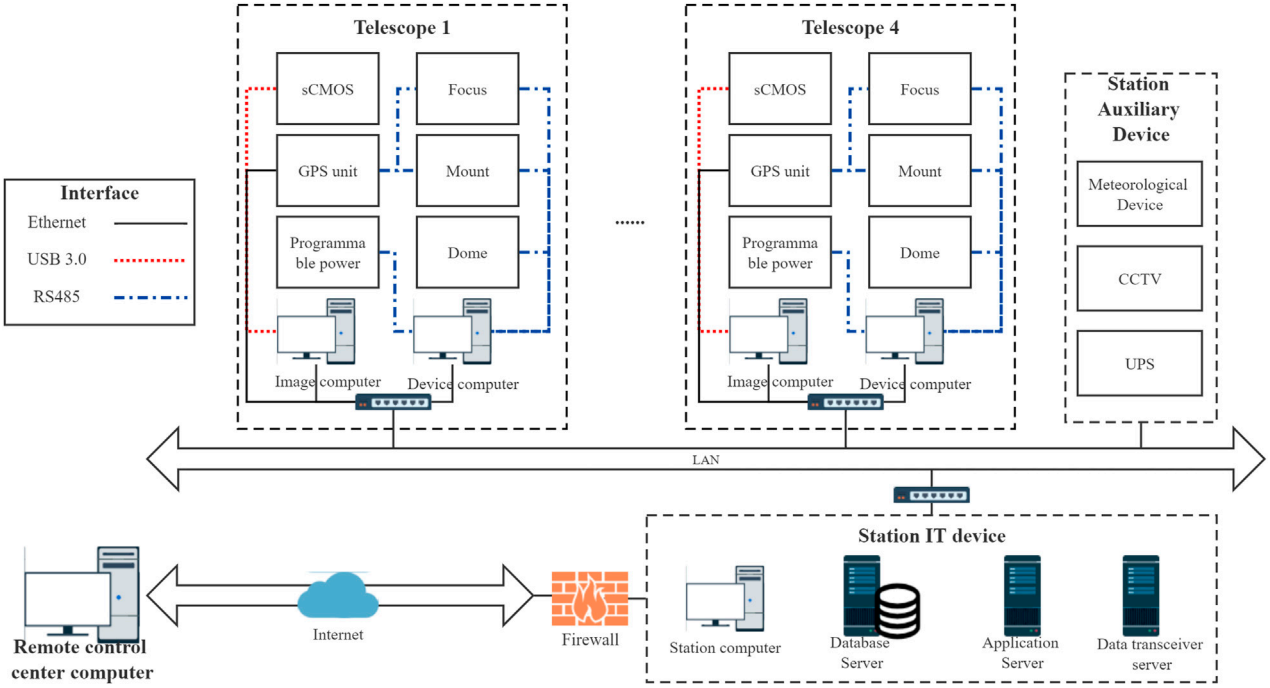


FIGURE 2
Hardware layout of Korla observatory. There are four 36 cm telescopes deployed at this modest observation site. The network port, USB 3.0, and RS485 are utilized to connect each component of the telescope as a whole (including the optical tube, mount, sCMOS camera, dome, and other additional components). Finally, the observatory's lone communication outlet is linked to the remote control center through the Internet.

The mount is used to support the optical tube to complete the pointing of the object. In order to realize LEO object tracking and high-speed slew, the manufacturer adopts equatorial mechanical structure, and uses an incremental encoder (resolution: 5 nm) to obtain high-precision position information and brushless permanent magnet synchronous torque motor (continuous torque: 68 nm, continuous speed for continuous torque: 100 rpm) to drive the mount to rotate at high speed, realizing a maximum speed of about 20°/s and a maximum acceleration of about $10^{\circ}/s^{2}$.

The camera, one of the most important components to determine the performance of the system, adopts a high sensitivity, high frame rate, and large target area front-illuminated sCMOS camera (4,096 × 4,096 pixels, 9 μm pixels size, 10 fps, manufactured by Tucsen Photonics, Fujian, China). The Peak quantum efficiency (QE) is 74%@600 nm and the readout noise is about 3.6e-. The optical system of the lens tube covers a wavelength range of 450–850 nm, which is basically within the highest quantum efficiency range of the sCMOS camera (the average QE is more than 60% in the range of 400–750 nm). So, the combination of optical system and camera gives each basic imager a field of view of about 2.6° × 2.6°.

One difference from traditional astronomical telescopes is that our system is not equipped with a filter system. Because the front of the optical lens tube is a little crowded, it also maximizes the light collection efficiency when used without the filter.

The dome is used to protect the telescope. We chose a steel structure box customized by the manufacturer, which is 2.5 m high * 2 m wide * 3 m long, and opened from the middle to both sides. The main reason for choosing such a structure is that it is relatively simple and cheap compared with the traditional full-open shell dome structure.

2.2.2 Additional components

The tracking of space debris requires high time accuracy. So, a GPS unite (manufactured by Baijun Electronic Tech., Xi'an, China) is integrated into the system, and the synchronization accuracy is better than 100 ns. Moreover, since the mount and camera have their own time locking devices and programs, as long as the GPS device transmits reference pulse signals to each piece of equipment with high precision, the time of the whole system can be unified to ensure the exposure accuracy of the camera and the pointing accuracy of the mount. As for IT devices including computers, the system's NTP (Network Time Protocol) network timing service will be used to maintain the consistency of time.

At the same time, other auxiliary hardware, such as meteorological system (the brand is Vantage Pro2, manufactured by Davis Inst., California, United States), Closed-Circuit Televisions (CCTV) (with infrared camera function, manufactured by Hikvision, Hangzhou, China), and programmable power supply device, are equipped to provide auxiliary parameters required for robotic operation and ensure the safety of the telescope system. Finally, two workstations are used for a single telescope, one for the control of all equipment and one for data processing, as shown in Figure 2.

2.3 Software

2.3.1 System architecture and interaction

The software system is a critical component in achieving the robotic operation of the entire telescope system. Many professional telescopes or telescope networks have developed corresponding control systems to meet their specific needs. ASCOM (Astronomy Common Object Model) and RTS2 (Remote Telescope System 2nd Version) (Kubanek, 2010) are the most frequently used. ISON (Elenin and Molotov, 2020) and BOOTES (Burst Observer and Optical Transient Exploration System) are two examples of successful projects. However, with the upgrading of network technology, a better network framework can be used.

At present, the commonly used communication frameworks include: EPICS (Experimental Physics and Industrial Control System), CORBA (Common Object Request Broker Architecture), ICE (Internet Communications Engine), DCOM (Distributed Component Object Model), and ZMQ (ZeroMQ). Li et al. (2021) analyzed and compared these frameworks, and found that ICE, EPICS, and TANGO are more suitable for building control systems. While Tz et al. (2013) had chosen ZMQ to replace COBRA for TANGO software in 2013. So, our software framework adopts the ZMQ communication component (Dworak et al., 2012) to realize message bus communication, which has also been verified on many telescopes (Lyard et al., 2015; Wang et al., 2021). ZMQ provides sockets that carry whole messages across various transports. And the ZMQ asynchronous I/O model is suitable for scalable multicore applications (Tz et al., 2013).

Based on the experience of previous researchers (Zhang et al., 2016; Dyer et al., 2020; Wang et al., 2021), the control system is basically divided into a multi-layer architecture. So, this system will consist of the following three levels: instrument interface layer (that is Instrument Control System, ICS), telescope control layer (that is Telescope Control System, TCS, including Data Process System, DPS), and business layer (that is Observation Control System, OCS, including User Interface, UI). However, because these telescopes need to be remotely controlled, a Telescope Network Schedule System (TNSS) will be developed to realize the optimal scheduling of multiple pieces of equipment. The final software architecture is shown in Figure 3.

In terms of single telescope control, the system is developed layer by layer according to three conventional levels, as depicted in Figure 3. The instrument interface layer integrates all real devices to realize Input/Output (IO) control. ZMQ is adopted in this system to realize the communication of different devices in different environments (such as different computers, operating systems, or different interface languages, etc.). The telescope control layer realizes the logical control of each device, defines various Process Variables (PV), and implements some complex commands, such as meridian flipping, star tracking, and image processing. The business layer schedules all devices to complete users' tasks automatically and responds to changes in the PV of each device.

As for remote networking control of telescopes, the remote scheduling layer will play its role. Its core function is to formulate observation plans for optimization and ensure the maximization of the observation efficiency of all networking telescopes. Furthermore, the remote GUI in this layer mainly provides a human-computer interaction interface for personnel on duty, astronomers, equipment maintenance personnel, and managers, and remotely manages and controls the telescopes. The local status information will be continuously sent to the central database through the Internet.

The OCS of the telescope also draws lessons from the function of RTS2 and adopts distributed programming logic to design and realize a robust and easy to expand control system. The overall language is based on a hybrid of Python and C++. The Python language is mainly used for framework development, business process processing, and hardware execution scheduling. The modules that require high-performance computing and high-speed communication adopt C++, which mainly involves the realization of camera communication, image data gathering and processing, and other functions.

Next, this paper will describe several function points from the perspective of space debris observation.

2.3.2 Tasking

With the increasing amount of space debris and the deployment of more and more telescopes, the effective use of telescopes is particularly important. Since the amount of space debris is considerably greater than the number of telescopes, it is a classic N-P hard problem to figure out how to use the minimum cost to meet the observation needs of diverse tasks. Therefore, many works will adopt some intelligent optimization algorithms to solve such complex problems.

The mode of direct remote control by the control center is used due to the rich computing resources and basic orbit database. Then, the system integrates the Particle Swarm Optimization (PSO) algorithm (Jiang et al., 2017) to establish the central task scheduler of TNSS. The central scheduler will select the working telescopes according to the working state of

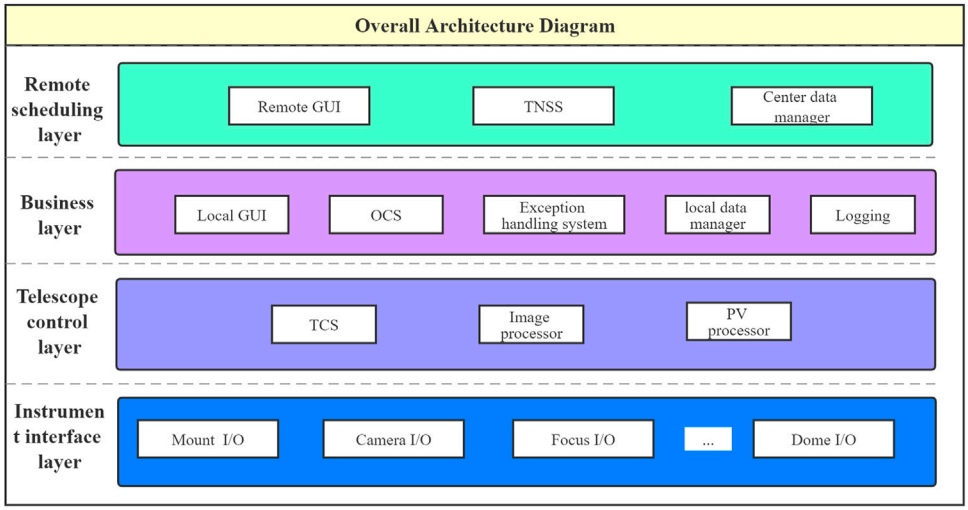


FIGURE 3
The overall architecture diagram.

TABLE 1 Definition of task-level under different tasks.

Item	Task type	Task level
1	Temporary important tasks	1[a]
2	Short-term space event response tasks[b]	2
3	Key object observation tasks	3
4	Maintaining cataloging observation task	4–6[c]
5	Other tasks	7
6	Testing tasks	8

[a]Level 1 is the highest priority and can replace any other task.
[b]Space events refer to space dangerous conjunction, such as large objects reentry, satellite breakup, etc.
[c]The cataloging requirements of space debris will be divided into three levels according to the object updating time.

each telescope and the station environment every day, then select the object to be observed based on the daily task requirements, and then make orbit prediction according to the prior information of these objects, so as to formulate an efficient and practical observation plan for each telescope.

In general, task priority is the most important consideration in the optimal scheduling of the whole observation network. It determines the satisfaction of observation requirements of users in the final decision of an intelligent algorithm. The Table 1 shows the classification of specific tasks by the system.

At the same time, some observation details must be considered while optimizing the observation plan, such as observation elevation, the phase angle of Sun and moon, the object switching time, etc. By adjusting the weight of these factors, we can finally make full use of time to complete the observation task, then automatically distribute it to telescopes.

2.3.3 Automatic observation

When the telescope receives the observation plan, it can automatically perform the task. The core of cooperative scheduling of devices by actuators in the business layer is to adopt the timeline-based mode. The received plans or commands to be executed are decomposed into conflict-free time task queues of each device according to the state of each device, and then the actuators of each device are distributed to each device in real-time according to the time sequence. In addition, the actuator can arbitrarily add, delete, and modify the task queue of each device, to make timely modifications in the case of task insertion, equipment failure, or a sudden change of weather, for example.

Figure 4 depicts the core process of the automatic observation process of space debris. Firstly, the scheduler in the control center will analyze the observation demand of the day, and formulate an observation plan for each telescope. Then, after receiving the plan, the local OCS service will analyze the equipment's working condition as well as environmental information. When the observation threshold is reached, the telescope equipment will be automatically powered on step by step to make various preparations before observation, such as the flat. If the observation conditions do not meet the requirements, the system will wait until the end of the whole observation task. So, when the observation is ready, the actuator will begin loading the observation objects one by one, and then coordinate the rotation of the mount, camera exposure, and DPS to process the data. Various task data and working condition data will be sent back to the control center in real-time and backed up in the local

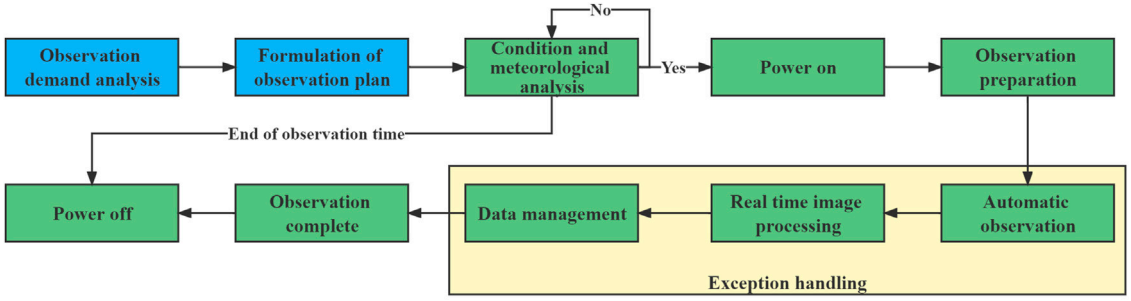


FIGURE 4
Observation process of the whole system.

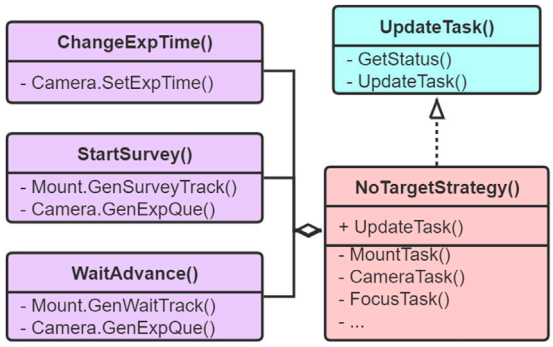


FIGURE 5
UML diagram of task failure exception handling.

database till the observation task is completed. In this process, the exception handling system (described in Section 2.3.4) always monitors the status of all equipment and the meteorological environment. If an anomaly is detected, the system will make corresponding disposal decisions according to the established operation strategy.

In addition to conventional observation tasks, the system also presets other automatic task scripts, such as flat exposing, dark exposing, focusing, etc., which are executed after reaching the preset conditions or after manual commands.

2.3.4 Exception handling

It was necessary for a system without human supervision to have a robust and reliable mechanism, which can take corresponding protective measures according to the risk degree of exception. Therefore, three daemons have been set up. One is the weather conditions, the other is the status of the telescope, and the last one is the observation execution results.

The daemon of weather conditions and telescope status are relatively easy to implement. Usually, the daemon processes the collected data (such as rain, temperature, wind speed, etc.) into three status flags according to the corresponding threshold: normal, warning, and error. When an error flag from a weather condition is received, for example, the system will be able to protect the telescope according to the established processing flow (such as stopping the observation task and closing the dome).

As for observation task exception handling, more empirical strategies are needed to deal with it. So, this daemon adopts the Strategy Pattern (SP) mode to realize the flexible combination, addition, deletion, and management of various specific strategies in later use. Take the example of no target found in the image, shown in Figure 5; specific strategies such as changing the camera exposure time (the function of ChangeExpTime), terminating the current tracking plan and searching in a certain area (the function of StartSurvey), and terminating the current tracking plan and waiting along the trace direction in advance (the function of WaitAdvance) are defined. The function of NotTargetStrategy is used to set and select exception handling strategy. Finally, the class of UpdateTask can update the task according to the system status when the task fails.

3 Result

The development of the remote automatic control system for the space debris telescope introduced in this paper began in early 2020. With the deployment of four 36 cm telescopes in Korla City, Xinjiang, China in early 2021, the whole system began to operate. The users can carry out remote Internet control on these four telescopes in the duty room in our control center, and carry out observation experiments of conventional space debris and asteroids. Next, some results of the experiment are briefly described.

3.1 Examples for space debris detection

After the installation and commissioning of the telescope, we tested the performance of the telescope. Figure 6 shows the origin

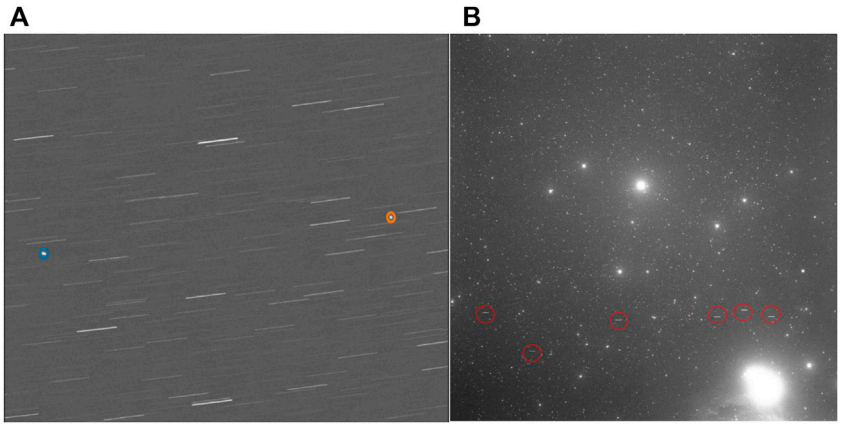


FIGURE 6
The observed images in different modes. **(A)** The observation image of the Starlink satellite, two satellites were captured in one image; **(B)** the GEO object image in stellar velocity (about 15″/s) observation mode, six objects were observed at the same time.

images captured by the 36 cm telescope in various modes. Due to its high-speed tracking ability, the telescope can track the LEO small satellite of the Starlink, as shown Figure 6A. Similarly, this large field of view telescope can survey and observe multiple objects. As shown in Figure 6B, six GEO objects are captured in one picture.

3.2 Telescope performances

Stable tracking is the premise for the telescope to obtain effective data. Figure 7 shows the tracking performance of the telescope. We selected the GPS satellite (in MEO and GEO) with high-precision ephemeris to conduct the tracking experiment, and then obtained its tracking error of telescope by comparing the observed data with the high-precision orbit data published by the satellite. As can be seen from Figure 7, the tracking accuracy of the telescope obtained by tracking four GPS satellites in MEO is basically maintained at 1-2 arcseconds. It is worth noting that the tracking accuracy here actually includes a combined error of telescope tracking and data processing.

We also conducted long-term monitoring of the ultimate detection capability of the 36 cm telescope. To begin with, we chose Gaia (Brown et al., 2018) as the reference star catalog, with a passband that covers the range [330, 1,050] nm, which covers our spectral range. Furthermore, the catalog has announced 1.69 billion, and its limit magnitude has reached 20.7 m_V, which can better meet the test requirements. The object position and magnitude are then extracted using the SExtractor software package (Bertin and Arnouts, 1996), and the real star position and magnitude in the image are filtered through star catalog matching. Finally, the magnitude of the darkest objects in the image is filtered.

From 10:00 p.m. to 11:00 p.m. (Beijing Time), we captured at least 10 images (exposure time: 10 s, tracking star mode) to calculate the average value on the same sky area. Furthermore, the weather is not always pleasant. Because the clouds can be dense on some evenings, we decided to observe for 1 month. Furthermore, the pointing area and time each night are slightly adjusted according to the actual situation. Figure 8 shows the change in the limiting magnitude of the telescope in a month. It can be seen that, under the condition of the crescent moon (around 2 January 2022), the performance of the telescope can reach 16.6 m_V, and when the moon is full (around 19 December 2021), the limit performance is reduced to 15.4 m_V. This performance has achieved relatively good results in telescopes with the same aperture, and this is the result after almost 1 year of installation. The dust accumulation on the mirror and the decline of the camera performance are inevitable. At the same time, it should be emphasized that the results obtained by us are not strictly the atmospheric apparent magnitude of the star, since we ignored some errors, such as the influence of atmospheric extinction, and used the standard filter system.

Image quality and focusing performance are also important factors that determine the final performance of the system. We defined the Full-Width Half Maximum (FWHM) of the image as the index to evaluate the image. We used the Sextractor package to calculate this value. When the threshold of FWHM is larger than 2 pixels, the system will automatically focus according to the V-curve method. Figure 9 shows a focusing process curve and the actual image. The 36 cm telescope usually takes images one by one with a focusing step of 0.05 mm to obtain the FWHM at each position (usually five images are taken before and after the current position). Finally, the best focusing position is obtained through the quadratic curve.

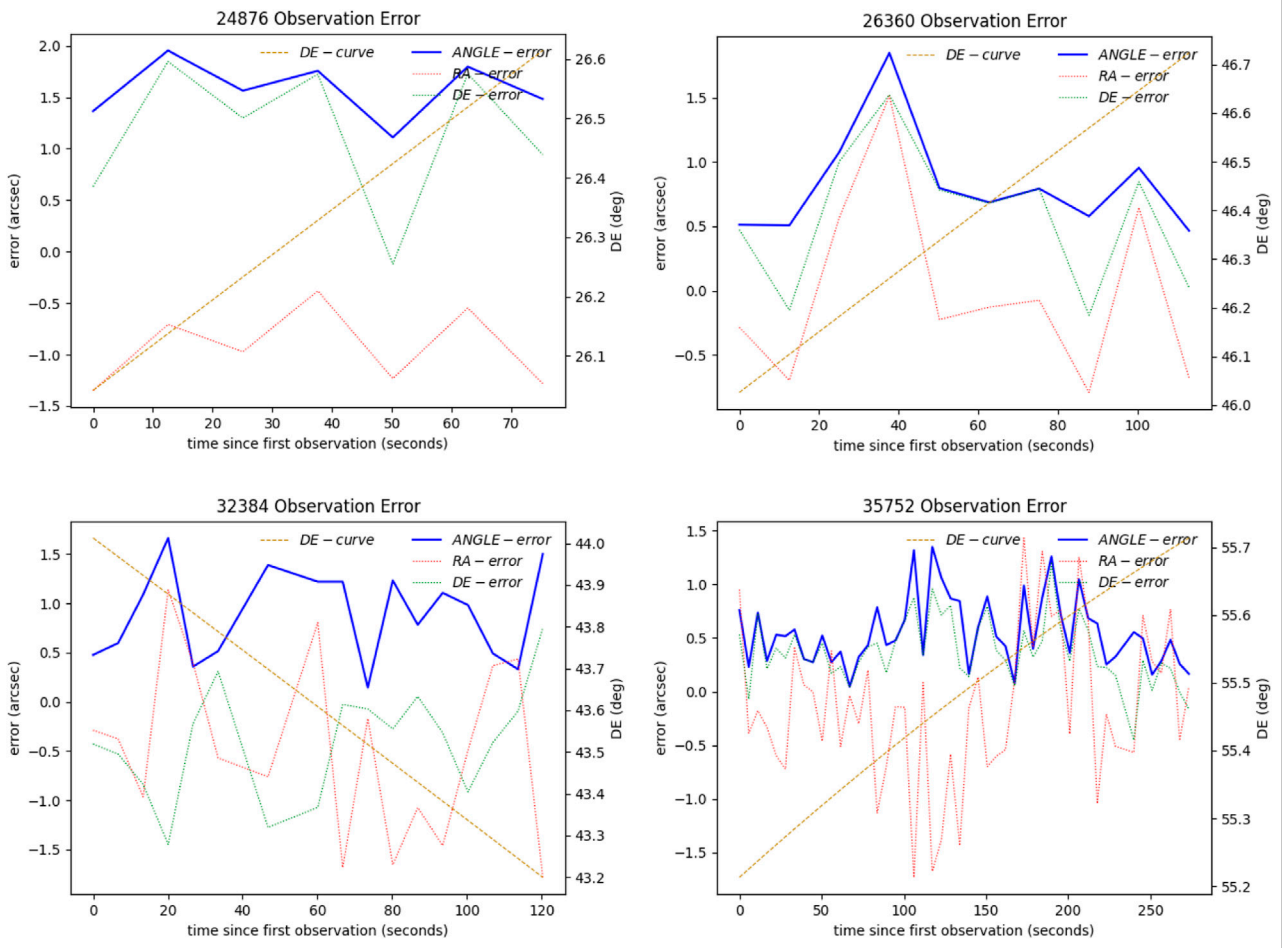


FIGURE 7
The tracking performance of the telescope. The NORAD code of the four GPS satellites (in MEO) are 24,876, 26,360, 32,384, and 35,752 respectively. The blue line (ANGLE-error) represents error synthesized by the two axes, the red line (RA-error) and green line (DE-error) represent the error of the right ascension axis and the declination axis respectively, and the brown line (DE-curve) represents the change of the declination value in the observation tracklets.

3.3 Statistics

The ultimate goal of the whole system is to obtain the position of the space debris. Figure 10 shows the number of object's arcs successfully observed and extracted by a single 36 cm telescope from 2 December 2021 to 3 January 2022. As shown in the figure, approximately 3,300 object arcs were successfully extracted over the observation period, and observation data are obtained every night. Therefore, in terms of the overall numbers, the effect of the telescope is still good. However, the number of arcs fluctuates substantially, with at least 23 arcs and a maximum of 163 arcs. While the data observation plan tracks about 160 space debris arcs every night (the solar altitude angle of -8° is set as the flag of the beginning and end of observation, GEO tracking takes place for 6 min and LEO tracking for 3 min). Moreover, the highest success rate of actual observation (the number of arcs actually observed in the planned arcs/the number of planned arcs) is about 78%, on December 11 and 30, which is a little insufficient to satisfy our requirements. There may be several reasons.

First, it can be noticed that there are several unplanned objects every night. Because our telescope has a field of view of 2.6°, we can capture multiple objects in one image. As a result, unplanned objects are likely to be obtained. Second, the optical telescope is still too weather-dependent. On several evenings, such as December 4 and 5, the humidity hit the 85 percent threshold specified by us, which automatically halted the observation task, resulting in an insufficient observation data. Then, clouds, as well as dust in the air, were particularly terrible, particularly on December 9, 14, 20, etc. When observing, the image has only a few stars or no stars, making subsequent

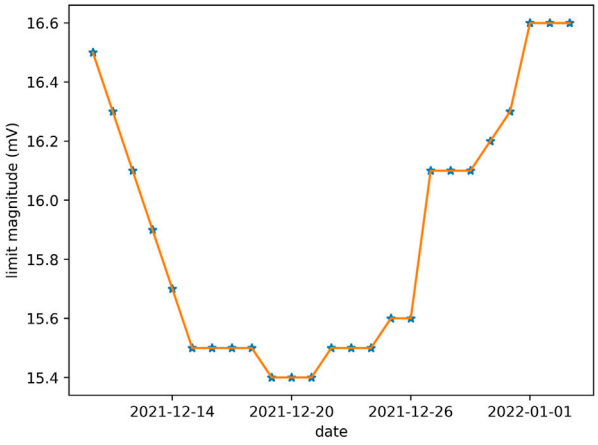


FIGURE 8
The change of the limiting magnitude (tracking star mode) with a 10 s exposure time of the 36 cm telescope in a month (from 2 December 2021 to 3 January 2022). Every night, the same sky area was selected to take images from 10:00 p.m. to 11: 00 p.m., and the limit magnitude of each day was calculated, which is represented by an asterisk in the figure.

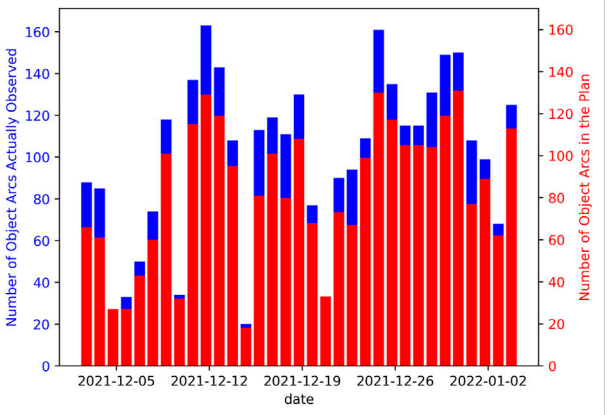


FIGURE 10
The number of objects arcs successfully observed and extracted by a single 36 cm telescope from 2 December 2021 to 3 January 2022. The blue bar represents the number of object arcs actually observed. The red bar represents the number of object arcs in the plan.

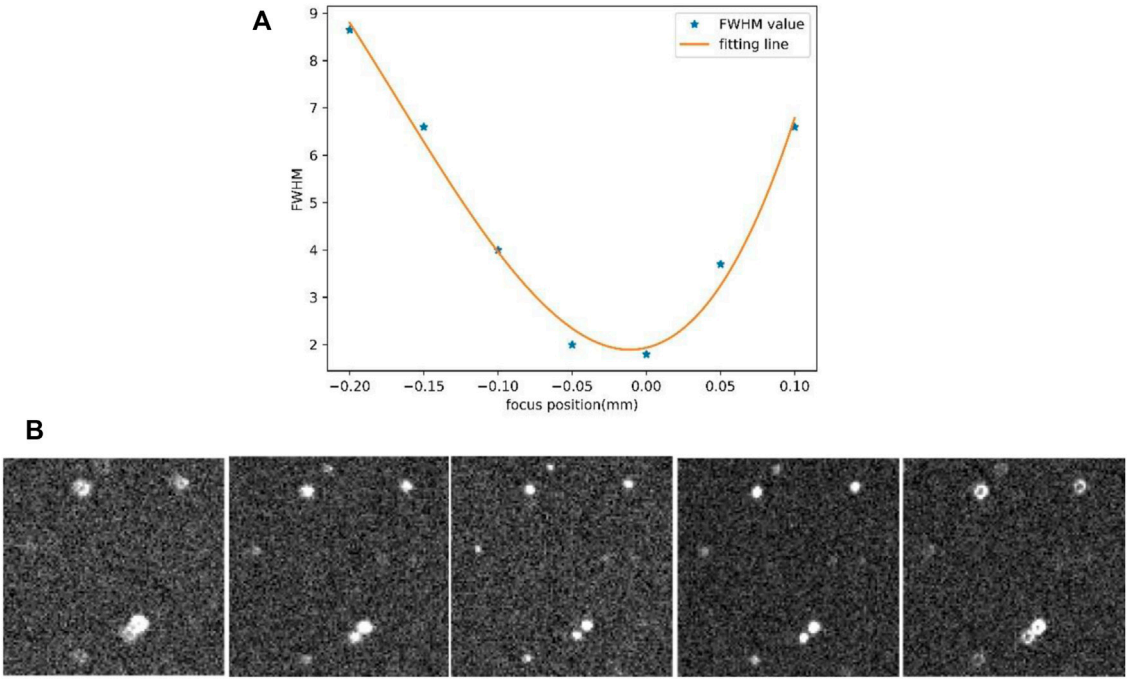


FIGURE 9
The result of FWHM curve fitting using the V-curve method. **(A)** Diagram of curve fitting. The X-axis is the focal position and Y-axis is the mean FWHM value of the image. **(B)** The sequence of images on different focal position.

astronomical positional processing and even simple object extraction impossible.

4 Discussion

This system is designed as a telescope system dedicated to space debris and asteroid observation. The slewing speed of the telescope mount is enhanced in hardware to switch object faster. The equipped sCMOS camera can generate images with a high frame rate (10 Hz, regardless of storage conditions), and capture more data for LEO objects. The installation of electric focusing, electric lens covers, and other facilities have transformed the original amateur telescope into a professional space debris observation telescope. Although it will be unable to conduct a certain study on object characteristics due to the lack of a filter system, it will ensure the telescope system's ability to collect light to some extent. After all, the primary goal of the system is to acquire information on the position of space debris.

From the results of the telescope system deployed in Korla, this 36 cm telescope can perform the tracking and observation task of objects in various orbits, and the tracking error within 1–2 arcseconds also ensures the accuracy of observation data. The limiting magnitude capability of about 16 m_V (10 s exposure with tracking star mode, without considering atmospheric extinction and color index of filter system) will also meet the detection of medium and large-scale space debris. Through the optimal scheduling and robotic operation of the system, the telescope can collect a large amount of data in a short time, for example, a single telescope can track more than 100 space objects arcs on average. The weather of observation site has an impact on the observation arc's quantity. Because of the close proximity to the desert, sand and dust are frequently hung in the air when the wind blows. Furthermore, urban lighting has an impact on the observation of low elevation. So, in general, we cannot label this site a great astronomical site, but rather an acceptable site that can match the objectives of space debris observation while also allowing us to conduct network experiments. Finally, we are currently focusing on tracking and observing certain objects of interest, but if we switch to a survey mode, such as a GEO belt survey, the number will greatly grow due to the big field of view.

The successful implementation of this system demonstrates that it is relatively easy to construct a simple automatic control system. Compared with the current ASCOM and the RTS2 framework, the technical route adopted by this software is simpler and makes the secondary development freer. Generally, the commercial hardware devices have completed the underlying control algorithm. So, the interface provided for user development is relatively simple. Users just need to pay attention to the top architecture and the integration of various devices during software development. For example, in this system, the core interface of the mount has 3 parameters: time, two angles, and two angle speed. The core interface of the camera has two parameters: start exposure time and exposure time. Then, the OCS and TCS send the command to a device in time according to the established time line, and immediately complete the control of the device. In addition, the system adopts mature message middleware ZMQ to solve the communication problems between the center and local, as well as between different equipment interfaces and different systems; the control strategy is written in Python, which has a low learning cost and is easier to implement. However, it should be considered that the GIL lock of Python will result in poor calculation speed and I/O interaction, so it is necessary to adopt multi-process and C++ for modules requiring fast calculation. Furthermore, some mature astronomical packages may be used for data processing, such as SExtactor, Astrometry.NET (Lang et al., 2010), and others, to retrieve the position, magnitude, image quality, and other information of the observation object we require.

However, it must be recognized that, compared with the mature international monitoring networks such as ISON and FTN telescope, we still have a lot of work to do. First of all, we should actively expand the breadth of equipment deployment areas, which will also promote the upgrading of our software system. Secondly, the system's perception of anomalies needs to be further improved, especially the weather perception ability. Due to the desert edge where the telescope is deployed at present, there is little rain and the possibility of sudden change of weather is low. So, once the telescope is deployed to the weather changeable area, the system needs to analyze and predict in combination with the cloud images, wind speed, humidity, and other data collected in real-time, to ensure the safety of unattended equipment. Finally, the next thing we need is to enhance the real-time collaborative scheduling of telescopes. For example, when a single telescope finds a new object, it can schedule another telescope for follow-up tracking, involving single station multi-device collaboration, as well as multi-station multi-device collaboration, which will also promote the upgrading of image processing speed, fast target recognition, and other capabilities.

5 Conclusion

Driven by the current demand for space debris and asteroid observation, this paper introduces feasible telescope software and a hardware scheme. The 36 cm RASA tube with 4 K sCMOS forms a 2.6°*2.6° large field telescope, with high precision tracking of space debris in LEO, MEO, and GEO. Many auxiliary devices also help the system become a robotic telescope. Since the deployment, the telescope has been working well and has obtained a lot of observation data of space debris.

Regarding the software system, this paper introduces the overall software framework. Based on ZMQ control architecture, we establish a lightweight cross-system and distributed system to facilitate the addition or deletion of subsequent devices.

According to the observation process of space debris, each functional module is designed. And this paper focuses on the realization of automatic operation and exceptional handling functions. This system greatly reduces the dependence on telescope operation and maintenance personnel. The remote unmanned operation of four telescopes proves the feasibility of this scheme, which will be extended to other telescopes in the future.

With the accumulation of operation experience, the system will further optimize the robustness of the telescope's remote automatic control software and improve the control ability of the telescope, to cope with the more complex observation environment, strengthen the collaborative control between telescopes, and give full play to the greater efficiency of the equipment.

Data availability statement

The raw data supporting the conclusion of this article will be made available by the authors, without undue reservation.

Author contributions

SJ and CH-W contributed to conception, design and implementation of the whole software. JH contributed to the design of scheduling software. JH and ZY-Y contributed to the design of the image process software. LJ contributed to the design of operation mode. SJ wrote the first draft of the manuscript. All authors contributed to manuscript revision, read, and approved the submitted version.

Funding

We acknowledge the financial support from the Space Debris and Near-Earth Asteroid Defense project of China (Nos. KJSP2020020201, KJSP2020020202, and KJSP2020020204) and the National Natural Science Foundation of China (No. 11803052).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Bertin, E., and Arnouts, S. (1996). SExtractor: Software for source extraction. *Astron. Astrophys. Suppl. Ser.* 117, 393–404. doi:10.1051/aas:1996164

Brown, A. G. A., Vallenari, A., Prusti, T., et al. (2018). Gaia Data Release 2. Summary of the contents and survey properties, *Astronomy & Astrophysics*. 616(A1). 22, doi:10.1051/0004-6361/201833051

Castro-Tirado, A. J. (2011). Robotic astronomy and the BOOTES network of robotic telescopes. *Acta Polytech.* 51 (2), 16. doi:10.14311/1308

Chun, F. K., Tippets, R. D., Strong, D. M., et al. (2018). A new global array of optical telescopes: The falcon telescope network. *Publ. Astronomical Soc. Pac.* 130 (991), 095003. doi:10.1088/1538-3873/aad03f

Crowther, R. (2002). Orbital debris: A growing threat to space operations. *Philosophical Trans. R. Soc. A Math. Phys. Eng. Sci.* 361 (1802), 157–168. doi:10.1098/rsta.2002.1118

Dworak, A., Ehm, F., Charrue, P., and Sliwinski, W. (2012). The new CERN controls middleware. *J. Phys. Conf. Ser.* 396 (1), 012017. doi:10.1088/1742-6596/396/1/012017

Dyer, M. J., Dhillon, V. S., and Littlefair, S., (2020). Developing the GOTO telescope control system, *SPIE*. 11452, 114521–114612. doi:10.1117/12.2561506

Elenin, L. V., and Molotov, I. E. (2020). Software for the automated control of robotic optical observatories. *J. Comput. Syst. Sci. Int.* 59 (6), 894–904. doi:10.1134/s1064230720040036

Jiang, H., Liu, J., Cheng, H., and Zhang, Y. (2017). Particle swarm optimization based space debris surveillance network scheduling. *Res. Astron. Astrophys.* 17, 30. doi:10.1088/1674-4527/17/3/30

Kennewell, J. A., and Vo, B. N. (2013). An overview of space situational awareness, in Proceedings of the 16th International Conference on Information Fusion, Istanbul, Turkey, 09-12 July 2013, IEEE, 1029–1036.

Kubanek, P. (2010). RTS2-The remote telescope system. *Adv. Astronomy* 2010, 9. doi:10.1155/2010/902484

Lang, D., Hogg, D. W., Mierle, K., Blanton, M., and Roweis, S. (2010). Astrometry.net: Blind astrometric calibration of arbitrary astronomical images. *Astronomical J.* 139 (5), 1782–1800. doi:10.1088/0004-6256/139/5/1782

Li, J., Wang, N., Liu, Z., Song, Y., Li, N., Xu, L., et al. (2021). Trends in architecture and middleware of radio telescope control system. *Adv. Astronomy* 2021, 1–10. doi:10.1155/2021/2655250

Lyard, E., Walter, R., and Kosack, K., (2015). *Modern middleware for the data acquisition of the cherenkov telescope array*.

Martone, R., Guidorzi, C., Mundell, C. G., Kobayashi, S., Cucchiara, A., Gomboc, A., et al. (2019). A robotic pipeline for fast GRB followup with the Las Cumbrés observatory network. *Exp. Astron. (Dordr).* 48 (1), 25–48. doi:10.1007/s10686-019-09634-y

Molotov, I., Zakhvatkin, M., Elenin, L., Canals Ros, L., Graziani, F., Teofilatto, P., et al. (2019). Ison network tracking of space debris: Current status and achievements. *RMxAC.* 51, 144–149. doi:10.22201/ia.14052059p.2019.51.25

Park, J., Yim, H., Choi, Y., Jo, J. H., Moon, H. K., Park, Y. S., et al. (2018). OWL-Net: A global network of robotic telescopes for satellite observation. *Adv. Space Res.* 62 (1), 152–163. doi:10.1016/j.asr.2018.04.008

Seitzer, P., Smith, R., Africano, J., Jorgensen, K., Stansbery, E., and Monet, D. (2004). MODEST observations of space debris at geosynchronous orbit. *Adv. Space Res.* 34 (5), 1139–1142. doi:10.1016/j.asr.2003.12.009

Tz, A. G., Taurel, E., Verdier, P., et al. (2013). TANGO–Can ZMQ replace CORBA? in Proceedings of ICALEPCS, San Francisco, CA, USA, November 2013.

Wang, Z., Tian, Y., Li, J., Cao, Z. H., and Zhao, Y. H. (2021). A study on universal observation control system and its application for LAMOST. *Res. Astron. Astrophys.* 21 (6), 149. doi:10.1088/1674-4527/21/6/149

Zhang, G., Wang, J., Tang, P., Jia, M. h., Chen, J., Dong, S. c., et al. (2016). An autonomous observation and control system based on EPICS and RTS2 for Antarctic telescopes. *Mon. Notices R. Astronomical Soc.* 455 (2), 1654–1664. doi:10.1093/mnras/stv2299

frontiers | Frontiers in Astronomy and Space Sciences

TYPE Original Research
PUBLISHED 30 September 2022
DOI 10.3389/fspas.2022.896570

OPEN ACCESS

EDITED BY
Alberto J. Castro-Tirado,
Institute of Astrophysics of Andalusia (CSIC), Spain

REVIEWED BY
Francesco Berrilli,
University of Rome Tor Vergata, Italy
Ignacio Olivares,
Institute of Astrophysics of Andalusia, (CSIC), Spain

*CORRESPONDENCE
Chen Zhang,
zhangchen@pmo.ac.cn

SPECIALTY SECTION
This article was submitted to Astronomical Instrumentation, a section of the journal Frontiers in Astronomy and Space Sciences

RECEIVED 15 March 2022
ACCEPTED 12 September 2022
PUBLISHED 30 September 2022

CITATION
Zhang C and Zhu C (2022), CHES robotic observation software kit. *Front. Astron. Space Sci.* 9:896570. doi: 10.3389/fspas.2022.896570

CHES robotic observation software kit

Chen Zhang[1,2]* and Can Zhu[2]

[1]University of Science and Technology of China, Hefei, Anhui, China, [2]Purple Mountain Observatory, CAS, Nanjing, China

CHES (changing event survey) is an optical survey program that not only aims at searching fast-moving RSOs in the sky for space domain awareness but also takes other scientific goals into account, such as NEO and transient events. After the success of the first array located at the Yaoan site in China, it evolved into a wider network consisting of various types of devices. This study presents a full-function framework for coordinating observation across such a network. The robotic observation system takes both extension flexibility and operation simplicity into account to meet special requirements such as timing, complex tracking, dynamic scheduling, unique device configuration, and distributed collaborative observation. Currently, this Python-based system has been deployed to several sites, supporting observation systems from single, entry-level telescopes to multiple medium-sized professional telescopes and performing predefined routing surveys and user-defined observation for different scientific goals. Some of them run unattended for a regular survey to maintain the base catalog and produce survey images for different purposes.

KEYWORDS

robotic telescope control, sensors, optical measurements, telescope network, space debris, wide field survey

Introduction

Telescope networks and robotic techniques

Astronomy is an observation driven subject oriented by new discoveries. Telescope networks can play important roles in many aspects of this subject, from high energy astrophysics to planetary science. As the development of telescope and detector technology, especially technics related to the robotic observation, more and more networks have been created to meet various scientific goals.

The International Scientific Optical Network (ISON) (Molotov et al., 2008) from KIAM is a very successful global optical telescope network with more than 30 telescopes at more than 20 sites which focus on near-Earth asteroids, space debris, and gamma-ray burst. KDS Polaris is its integrated telescope control system (TCS) written in C# for high-performance, fully automated observation within the framework of the survey program for searching small bodies in the solar system, observation for target designation (both asteroids and comets and space debris in the Earth orbit), as well as for alert (urgent) observation of short-lived optical transients, such as optical components of gamma-ray

bursts afterglow. In addition, another Python-based distributed client–server architecture TCS Forte (Kouprianov and Molotov, 2017) is developed to enable extreme flexibility and scalability to a wide range of sensor apertures and configurations. The Burst Observer and Optical Transient Exploring System (BOOTES) (Castro-Tirado et al., 1998) from IAA is a network of 60-cm telescopes to quickly observe transient events within seconds or minutes of being detected by scientific satellites. Its networked robotic driving system is the well-known Remote Telescope System, 2nd version (RTS2) (Kubánek et al., 2004), which is composed of several device servers, central server, and various observational clients that cooperate over a TCP network. Also, BOOTES has a remotely accessible system whose TCS is based on ASCOM under Windows for device compatibility. Télescope à Action Rapide pour les Objets Transitoires (TAROT) is an optical network for GRB (Boër et al., 1996) and RSO observation (Boër et al., 2017). The developed control software ROS (Klotz et al., 2008) is a set of programs connected by a web interface. The Gravitational-wave Optical Transient Observer (GOTO) (Dyer et al., 2020) is a wide-field telescope network that focuses on detecting optical counterparts to gravitational-wave sources, which consists of multiple telescope units with a shared robotic mount. The GOTO Telescope Control System (G-TeCS) (Dyer et al., 2018) comprises of multiple independent Python-based control daemons and a "just-in-time" scheduler, which are supervised by a master control program. The Arizona Robotic Telescope Network (ARTN) (Weiner et al., 2018) project is a flexible 1- to 3-m class telescopes network to carry out monitoring, rapid responding, and transient/target-of-opportunity following-up in various domain of astronomy. The project creates an INDI-based TCS-NG (TCS Next Generation) along with AzCam to support RTS2 working as a control system. The Stellar Observations Network Group (SONG) (Grundahl et al., 2008) is a global network of 1-m telescopes to be able to observe single objects continuously for days, weeks, and even months. The overall system is operated by a database-driven control system, which consists of several Python software packages (Andersen et al., 2019). The Las Cumbres Observatory (LCO) (Brown et al., 2013) is a private operating network of astronomical observatories mainly for time domain astronomy. Its robotic control system (RCS) (Fraser and Steele, 2004) works along with the telescope control system (TCS) and the instrument control system (ICS). The Search for habitable Planets EClipsing ULtra-cOOl Stars (SPECULOOS) (Delrez et al., 2018) project keeps a close eye on terrestrial planets hunting around nearby cool dwarfs. It uses the commercial observation system DC-3 Dreams® ACP. The Test-Bed Telescopes (TBT) project (Ocaña et al., 2016) works as a prototype of the autonomous optical observing system for future NEO and space situational awareness networks. RTS2 works as its TCS and specialized planning and scheduling softwares (Racero et al., 2015) are adapted to RTS2 to work together.

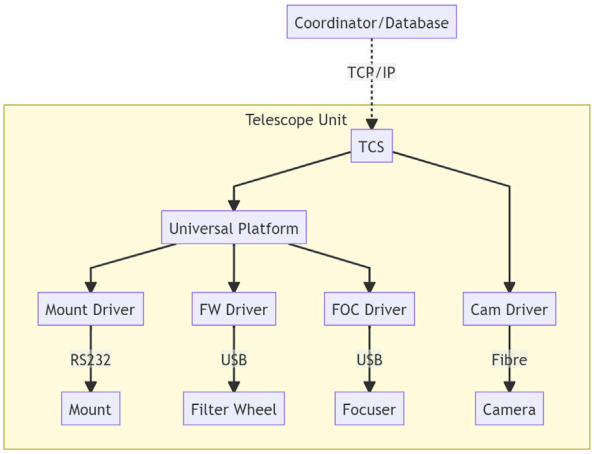


FIGURE 1
ogical connection of TCS. It communicates with upstream modules such as observation coordinator and operation databases and connects all devices within each telescope unit. Typical devices including mount, filter wheel, and focuser are connected *via* a universal platform, which can supply a standard API, but also some devices such as camera can be connected directly due to the performance requirements.

CHES program requirements and existing techniques

Changing event survey (CHES) (Chen and Changyin, 2021) is a general optical survey program mainly aimed at cataloging and discovering Earth orbital resident space objects (RSOs), but it also follows the interests of other astronomical goals. It is carried out by multiple small- to medium-sized wide-field optical telescopes and some other auxiliary telescopes, which form a complex observation network located at multiple sites.

Basically, each sensor unit is a standard astronomical telescope, but to meet the requirements of RSO observation, there are some special features, including the following:

- GPS timing latch for frame high-precision timestamping;
- tracking with a custom rate and a variable rate other than the sidereal rate;
- support special telescope configuration, such as multi-instruments, multi-channels, and collaborative telescope array;
- dynamic observation coordination;
- on-site plan adjustment from real-time observation feedback;
- multi-user and multi-goal observation robotic coordination.

These requirements are ultimately manifested in not only telescope hardware and software but also the upstream coordination system. Because of the development of

instruments techniques and the difference in operating conditions, the network contains many kinds of telescopes, and the observation requirements may expand as new types of devices are used. Therefore, the system should be flexible and scalable to face the variety of devices, drivers, software environments, and observation requirements.

The core component of this kit is the telescope control system (TCS), shown in Figure 1. It connects devices, interprets observation requests, and carries out actions. To support different hardware configurations within a uniform operating framework, the device-specific code and general function code should be separated properly; the common practice is wrapping each device's code into a standard API.

There are many TCS solutions, from open source Ekos; to proprietary Software Bisque® TheSkyX, Cyanogen Imaging® MaxIM DL, and DC-3 Dreams® ACP; professional systems such as OCS (Hickson, 2019) used by ES-MCAT; and systems mentioned before such as RTS2 and FORTE. These open and commercial systems usually could perform scheduled observation sequentially and have been widely used in astronomical photography and small observation projects. However, there are some limitations if we want to apply them to the CHES program, such as missing the networking, rate tracking, or timing function. Because of this reason, we decided to develop a whole system for CHES robotic operation as other professional systems do.

A universal device platform provides a consistent API for the same type of devices, which helps to reduce the compatibility code within TCS for various devices support. Currently, in open source community there are several universal platforms that have broad compatibility, such as ASCOM/ASCOM Alpaca and Indi/Indigo. Also, RTS2 has its own universal platform. ASCOM is called a *de facto* standard for amateur to entry-level professional telescope, which provides a universal API for different types of telescope mounts, cameras, auxiliary devices, and even useful telescope control functions. To add cross-platform capability, the ASCOM team published a new platform called ASCOM Alpaca. ASCOM Alpaca uses restful technology to implement a web-based API other than the traditional ASCOM's Microsoft® COM. After years of development, there are plenty of support from manufacturers and software clients. Indi and its fork Indigo are a cross-platform distributed telescope framework, which uses XML-based C/S architecture. They are more welcomed in the open source community under the POSIX system and support many embedded systems such as Arduino. In addition, there is a Windows Indi server to bridge the ASCOM API. RTS2's bridge is not a standalone platform. It supports Indi API, but in most cases it uses independently developed drivers for performance. Limited by the developing community, the device support capability is insufficient, and many supported devices are old model.

Each device has its own driver, and a driver wrapper if it supports other platforms. Usually, this is enough to adapt to the TCS, but because of the compatibility consideration, this solution may lack performance and some functions, especially noticeable in the camera driver. The image data throughput of modern high-resolution high frame rate CMOS is far beyond the capability of a universal platform, also the rich settings. So, in some cases, a customized driver directly from device SDK is preferable.

FIGURE 2
Photo of the CHES-YA telescope array located at the Yaoan site which consists of 12280-mm refractor under a rolling roof.

Hardware systems

Currently, the CHES observation system has been used on multiple sets of telescopes at multiple sites, which involves different types of instruments.

- The first CHES array is located at Yaoan. It consists of 12 280-mm refractor, shown in Figure 2, and two 800-mm reflector and comprises an ASA DDM85P equatorial mount, FLI PL09000 CCD camera, ASA AZ800 altazimuth telescope, and Andor iKon XL 231 CCD camera;
- the Dragonfly telescope network is located at Lenghu, Muztagh, and Samoa; each site has one 280-mm refractor and one 400-mm reflector and comprises an ASA DDM100 equatorial mount and a FLI KL4040 CMOS camera;
- 400-mm reflector at Ali uses a 10 micron GM1000HPS equatorial mount and a FLI ML50100 CCD camera;
- 200-mm reflector at Xuyi uses a Paramount MX equatorial mount and a QHY600P CMOS camera.

All these sites are connected to the Nanjing center for universal coordination.

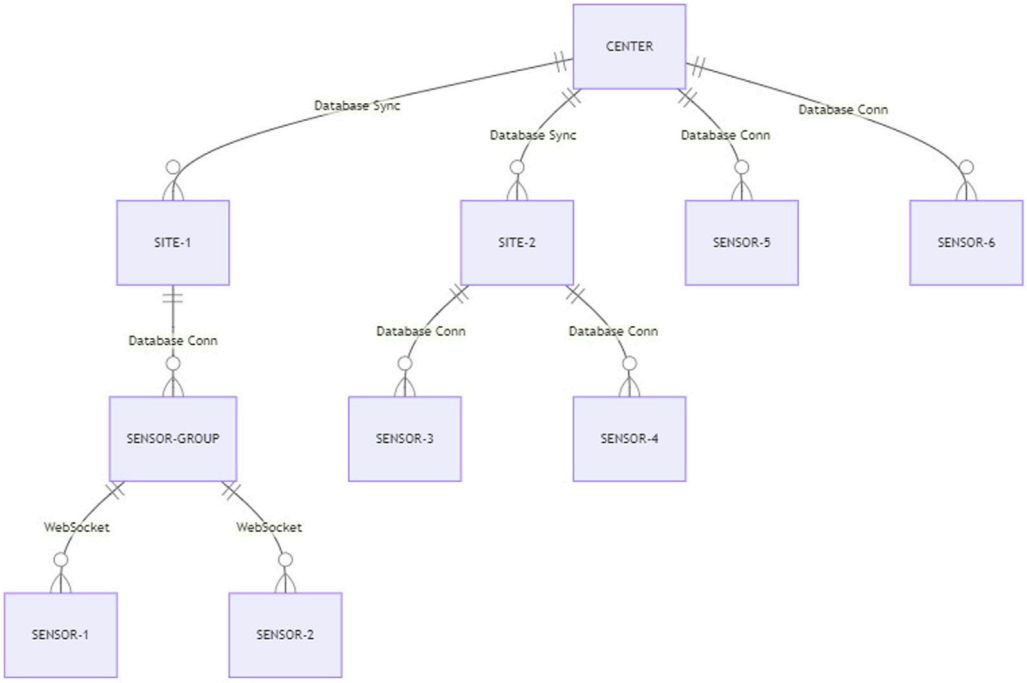


FIGURE 3
CHES optical telescope network architecture. Sensor is a minimum single telescope unit which can be work independently; sensor group is a bunch of sensors which logically work as an entirety in the same site; site means all sensors in a geographical observatory that share one CHES supporting facility, such as weather and on-site data center; center is a logical data center which aggregate all the sensors' I/O together, either locates in PMO or its redundant backup center.

Observation system design

System architecture

The dedicated observation system for a certain project or facility has different design patterns. The all-in-one (AIO) system integrates all functions into an integral software program works as a black box with plan input and data output, usually be developed for a fixed model facility that has fixed functionality. The do-one-thing (DOT) system follows the philosophy of do one thing and do it well from unix community, consists of a bunch of programs and uses scripts to call these programs sequentially to finish a job, and usually works with research-grade facility. The AIO system has consistent usage experience, but it lacks flexibility and robustness. It is difficult to extend the observation mode and needs intensive maintenance to ensure the availability of all mutually influenced parts. The DOT system is flexible enough, can meet almost any needs with extraordinary design, and has lower development cost by introducing massive community resources. The problem is the user experience and maintenance difficulty.

The CHES survey system takes both ideas into account by using a multilevel scheme. The software architecture is designed as several layers, center, site, sensor group, and single sensor. The basis of this distinction is the operations' mutual dependency within each layer and the decoupling between layers. Each layer has a single program to deal with the internal logic, and they work together *via* communication. Some failure may only cause functional loss other than crash the whole system, and it can be replaced easily. For example, if the weather monitor in the site layer fails, the robotic operation will be damaged. But we can replace the monitor with a simple time-related logic temporarily if we are sure about the weather until we put the weather monitor back online. This could keep the system operational capability as much as possible. The center layer works as a data hub and user interface, which involves observation requests, working status, and final data products. The site layer provides support for observation such as operating conditions and environment monitoring. Sensor groups and single sensors are the actual operators, and the difference is about telescope arrays. The sensor group program can perform joint observation by controlling several telescopes to work together. Inside each program, procedures follow a strict logic and work as an entirety; the communication between programs uses a file system, database, or WebSocket depending on the timeliness requirements. The overall system architecture is shown in Figure 3.

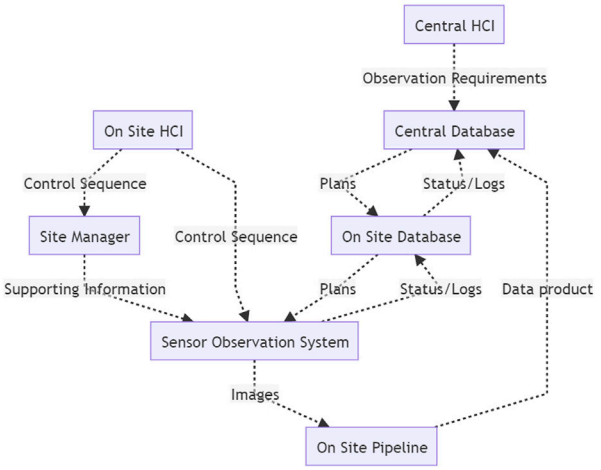


FIGURE 4
CHES network system connection and data flow. Each part runs a set of closely connected programs from this software kit and work together by exchanging different types of information.

Data model and work flow

Under full-function operation, the data center receives generally described observation requests from upstream calculations and user indications. These requests will be translated into specific observation plans that can guide certain sensors and then be distributed to these sensors. When the plan is acquired by the sensor, the sensor will ask the site for assistant information to decide whether to perform. Finally, the sensor program controls the telescope to carry out actual actions. Because of the need of real-time feedback, there is an on-site data reduction facility to reduce raw images, and the reduced product is fed back to the system as well as the center database. The overall workflow is shown in Figure 4.

Most non-real-time data communication is performed *via* database systems, such as plans, data products, operation logs, system status, and environmental information. DBS provides data consistent communication for multiple clients synchronization, data permanence, and rich query capability. Under the circumstance of a poor internet connection, a loosely bidirectional database synchronization mechanism is implemented with a restful server at the center.

In addition, there are some other real-time communication needs, such as device manual operation and massive communication for HCI whose data do not need to be retained. These will be done using WebSocket communication. The WebSocket protocol can be used easily in web environments.

The computation, storage, and transmission throughput of modern computer systems are high enough to let us choose flexible formats other than formatted or binary data structures. Here, we use JSON as the primary information exchange format among different parts of this software kit, such as control sequence and state quantity. JSON is simple, human-readable, flexible, scalable, and perfectly matches the dict type of system language Python. We have defined JSON keys for observation plans, device operations, and site conditions to fulfill the fine-tuning of each observation.

A sample JSON plan is shown in Appendix 1. This is the observation guidance part of a plan directly related to the telescope control; the other parts for plan query are presented in other column of the plan database. According to this plan, the TCS will get device prepared such as bias and dark calibration frame acquisition, switch filter, and set up camera parameters. Then move the telescope and take exposures. Most of the information in this plan is action parameters. We used SkyCoord from astropy.coordinates (Astropy Collaboration et al., 2018) other than the values in the special coordinate frame for flexibility; no matter what coordinate the telescope used, the device code will transform the coordinate to the right one. Following the ASCOM convention, in a common scenario, the coordinates will be transformed to true equator true equinox (TETE) when sidereal tracking need to be activated immediately after slewing and altitude–azimuth (AltAz) when we want the telescope to keep still after slewing. The coordinate frame is prepared based on the site location and time in advance. The camera control is the most complicated part if we want to achieve the best performance under various scenarios. The JSON format is suitable to extend parameters keywords. Other than the common readout mode, binning, subframe, offset, and gain, we can have more such as the CMOS merge mode, USB traffic, DDR buffer, and even FITS (Wells et al., 1981) header keywords setting. The action can be guided according to the presence of these optional parameters. Moreover, the parameters can be repeated to adjust the behavior of a certain frame, to observe a series of coordinates or with a series of filters.

Data archive

The main data product of CHES program is images. The format of image data is FITS supported by the astropy package, including tile compression for bandwidth saving and multi-extension for exposure grouping. The ASDF (Greenfield et al., 2015) format is another supported modern choice, and the image saving–related code is isolated to support various formats.

Typically, one sensor can produce 200–1000 GB raw images per night depending on the camera resolution, because of the short exposure. Images will be sent to the on-site storage for reduction while being cached in an acquisition computer for a couple of days. Raw images of short exposure are cached in on-site storage after reduction for several months. Stacked, ROI, and other scientific images are tile compressed, archived, and sent to data center for permanent retention *via* Internet, tape, or disc. This reduces each sensor's image archive size to tens of gigabytes per night. The raw images can be accessed internally for backtracking during the caching period, whereas the archived images can be accessed publicly from data center.

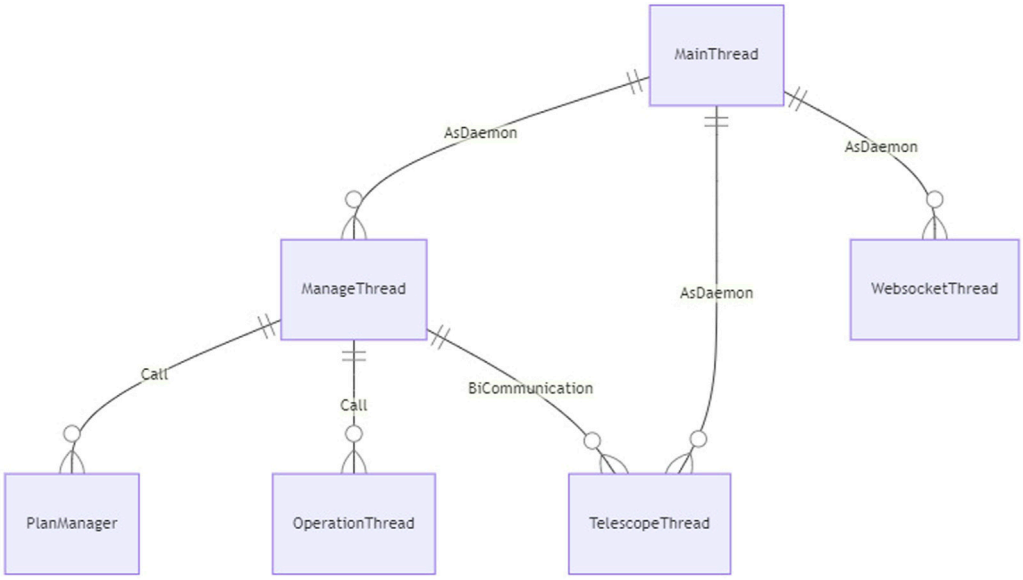


FIGURE 5
Modules and relations among them within the TCS program structure. MainThread, ManageThread, WebsocketThread, and TelescopeThread are long-lived daemon thread to maintain the system operation. The extensible OperationThread performs the observation plan and PlanManager provide plan acquisition strategy.

Observation system

Programming language

Programming technology affects the implementation. Thanks to the high performance and broad compatibility, static languages such as C++ or C# are used when hardware and massive calculation are involved. Meanwhile, dynamic languages can provide more flexibility and scalability when dealing with the control pipe logics. Python community has extraordinary base in machine learning, scientific computing, hardware controlling, and even web hosting. This made us to choose Python as the main programming language of TCS, site manager, web interface, and other auxiliary programs. It is also a good choice to glue all static language drivers together even by Cython coding when performance is required.

TCS and observation program

To adapt different working environments and telescope configurations, this program is designed as cross-platform and multithreaded. The program has the following features:

- direct access to all hardware devices, including mount, focuser, filter wheel, rotator, timing, and so on;
- translate the plan into device actions and operate properly;
- feedback system overall status, especially device status during operation;
- query plan with predefined strategy which may differ for each telescope;
- auxiliary features, such as configuration, logging, HCI, WebSocket communication, and database access.

This program has several modules to accomplish the observation goal and the relations among them, as shown in Figure 5. During the whole life cycle of a running instance, there are four threads that keep working, to deal with GUI responding, websocket communication, telescope controlling, and operation management. The practical operation related to a specific plan is performed by an independent operation thread which can choose different modules according to the plan indication, such as observing a series of coordinates or tracking a moving object. Currently, the operation module collection includes coordinates observation and orbital tracking observation and can be extended *via* a program plug-ins mechanism in the future. The decision of which plan to perform is made by Plan Manager according to the information in the plan database. Basically, all the modules communicate with the central management thread.

Telescope control

To support different devices, the usual method is to abstract each type of device driver and encapsulate it into a common API,

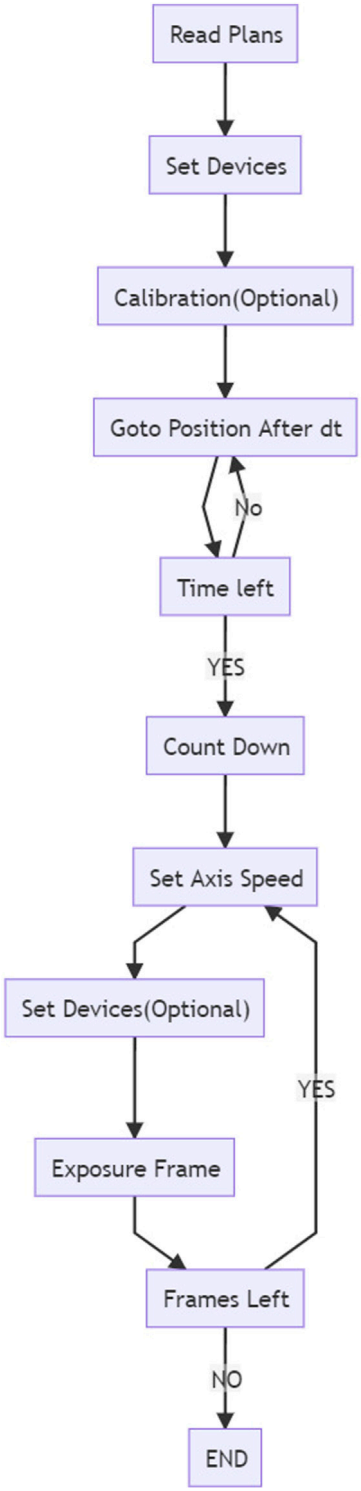


FIGURE 6
Open loop orbital object tracking observation workflow. This is key to observing very faint fast-moving object, to make sure the signal accumulate correctly. Due to the strong correlation of time and position, the workflow is time sensitive.

such as ASCOM or Indi. If it is impossible to use all devices with ASCOM or Indi platform, then the driver must be encapsulated again. For most cases, this would be a duplicated work; and if there is no common API for some special devices, the code base could be messier. In this system, we decide to abstract the device code from another perspective that encapsulates several actions into a common operation, such as slew, set rate, take image, or even startup and shutdown. This can still use ASCOM or Indi platform, if available, to reduce the development complexity. Each operation involves multiple devices, so they work together more closely and have many more possibilities. In addition, we can add system features related to the telescope, such as preparing a standard FITS header for data reduction and managing the remaining disk space. For CHES project currently used devices, we implement a device configuration based on ASCOM drivers and a customized timing tag device. However, because of the particularity of Ascom 2X Mount Adaptor's offset tracking rate used by the Paramount mount, we have another telescope configuration using TheSkyX RASCOM.

Image acquisition

Using a device from Python *via* ASCOM is very convenient, but there is a performance issue when reading a camera. ASCOM uses Microsoft COM technology and stores image data in SAFEARRAY. When converting a very large image from a SAFEARRAYto a NumPy array, it takes a very long time because of the loop used. A safearray_as_ndarray decorator from the comtypes module can improve slightly but not sufficiently when handling the modern large format CMOS. Therefore, we develop two Python modules, python-qhy and python-fli, with ASCOM compatible APIs to support the QHYCCD® and FLI® cameras. The module is written in Cython language to call the native SDK on Windows and Linux. The code can operate the camera as fast as the SDK original speed. The module has both official API and ASCOM API to use under different circumstances.

When observing RSO, the time accuracy should be higher than micro seconds. The operation of the camera is controlled by internal logics, so a high precision time tagging system is used. One method is to use a GPS PPS signal to trigger the camera externally, and the other is to trigger the time latch by using the frame exposure indicator signal from the camera. The timestamp tagging procedure can be performed by using the camera internal system or read from a serial port. This system supports either way.

Coordinate system and tracking

The coordinate system used by the telescope includes topocentric equatorial, J2000 equatorial, and altitude–azimuth coordinates. Thanks to the astropy.coordinates module, the system supports different coordinate inputs and transforms

them to required coordinates. The topocentric frame is equivalent to the TETE frame from Astropy.

When tracking RSOs, the telescope moves nonsidereally. Usually telescopes support offset tracking and/or axis movement. Offset tracking is more precise, as the axis rate is calculated with pointing model, but the axis movement can be faster in some of mounts' software. This system supports both methods and chooses offset tracking preferentially. When performing rate tracking, the average rate from position difference is recommended rather than the instantaneous rate. This is to keep the telescope axis rate fixed during the exposure, to ensure that the center of the star streak is related to the middle of the exposure time.

Observation method

The designed survey types include sky field surveys, solar body surveys, and Earth orbital surveys. The major difference is the tracking rate because latter's tracking rate is time-varying. Each observation mode is implemented as an individual derived class of `threading.Thread` and works as a plug-in to the main program. This means we can extend the observation mode later without modifying the main program. Each time it starts a new plan, a new observation thread instance is started until the plan is completed or aborted, so the actual observation night is divided into multiple observation units. The observation thread and daemon thread can access the devices at the same time.

Sky field observation involves taking several images at specific coordinates with specific rate sequentially. First, all parameters in the plan are set, including exposure time, camera parameters, and filter, and then calibration frames are taken with set camera mode if necessary. The second step is slewing the telescope to the coordinates and setting the tracking rate. After that, the frames are exposed. It is worth mentioning that the system can perform all set procedures before a specific frame so that the plan is flexible enough for most cases.

For NEO or RSO observation, the tracking rate is strictly time-related. To maximize the detection depth, we should set the rate before each frame. We do not implement it with frame lists only like the sky field survey mode. Instead, we use instance-guided open loop tracking with a custom coordinate generator, such as TLE propagator, Ephem propagator, or ephemeris table lookup generator. The procedure is shown in Figure 6. Similar to sky field observation, first, the telescope should go to the coordinate at 20 s later and check the remaining time until there is still some time left. After that, count down and start to move the telescope by using the axis rate control.

A plan defines a set of actions with sequential logic, and it is irrelevant between plans. The life cycle of a plan is shown in Figure 7. Therefore, there is no need to make a strict sequence for plans; instead, the observation resource can be maximized by dynamically adjusting oversaturated plans. The possible conflict of plans requires an instant scheduler. In each plan record, in addition to the guidance data, there are also metadata to help the decision system in the scheduler. Metadata includes user-defined priority, valid period, usage timespan, requested fields, objective, and list of demanding sensors. The scheduler will query the plan database and use this metadata to decide which one to start or even abort the running one. With this strategy, all 12 CHES wide-field telescopes can be used as one telescope.

There is an all sky field grid map for each telescope according to the field of view, which is recorded in an individual database table. Other than arbitrary coordinates the system can also observe with a predefined sky field serial number. It is good for sky template reduction during the image differential. Any visit of these sky fields will be recorded so that the system can determine which field needs to be visited if the system is free. This helps the system to perform a blind all sky survey while accomplishing other requests.

We define several run levels to support the robotic operation, and daemon thread operates according to the run level. The run level will be assigned by operators, and there are two procedures related to the switch of run level, startup, and shutdown. The shutdown procedure will park the telescope, turn off the camera cooling, switch filter to protector (an aluminum sheet in filter wheel to protect CMOS camera without shutter), turn off mount motor, and turn off mirror shutter, if they are available. The startup procedure will operate in reserve, take the master bias frame, wait for the cooling temperature, and manage the remaining disk space by deleting the outdated file directory.

- −1: System uninitialized;
  - 0: Stopped, if switched from other levels, all operations will be aborted, and then shutdown procedure will be performed;
  - 1: Paused, keep the telescope standby, started but do nothing;
  - 2: Manual, accept operating instruction from GUI and Websocket server, to prevent the conflict between robotic operation and manual operation;
  - 4: Guided, operate telescope totally according to the plans, usually for system debugging;
  - 5: Robotic, the system will read plans, site condition, and calculate Sun elevation to decide the operation according to the robotic strategy.

Database and interface

The whole system relies on a database system and uses JSON as the primary data exchange format. Therefore, PostgreSQL is the first choice because of its good support of JSON type. Considering the various conditions, the system uses object–relational mapping (ORM) technology from the

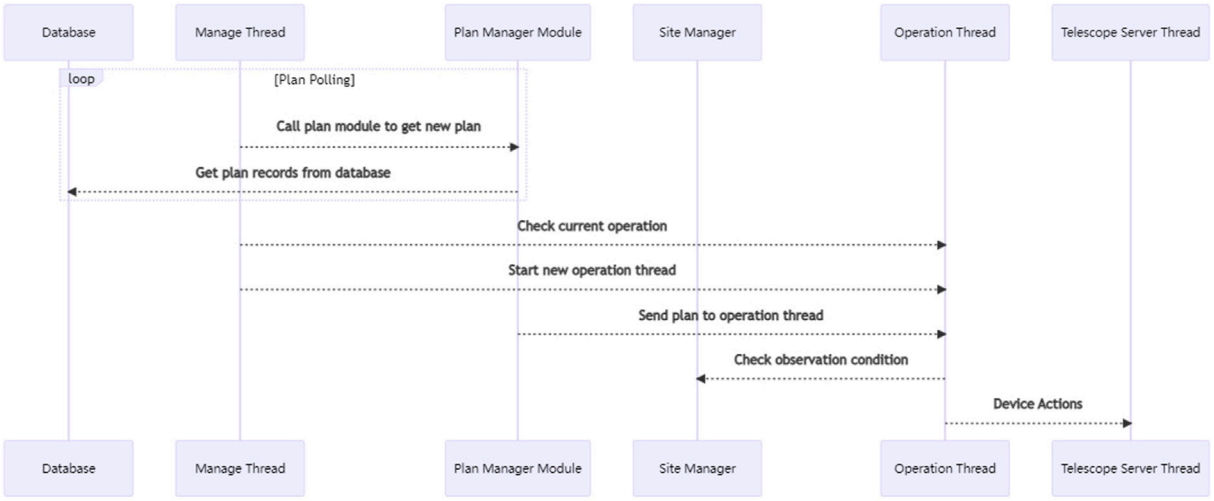


FIGURE 7
Observation plan life cycle. This shows a plan how to be acquired, performed, and which modules are involved, after the plan is inserted to the plan database by user. In addition to the modules in TCS, the site manager is involved to provide the observation condition related to the plan, such as the partial sky field weather condition that plan requests. The observation steps are from top to bottom in order.

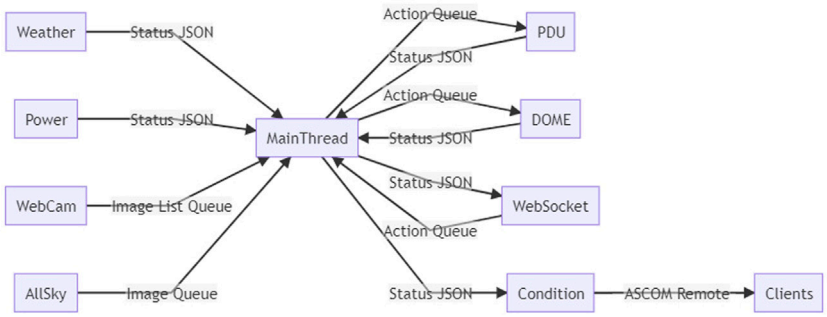


FIGURE 8
CHES site manager architecture. The main thread starts the other threads. Each one is responsible for one function, such as weather monitor device reading, dome controlling, and remote request answering. All the function are coded into the separated thread class so that it can be easily modified, extended or replaced to adapt different scenarios which usually do not have common platform API.

`SQLAlchemy` package so that it is easy to turn to other database systems. Similar to the SQLite in the Ali 400 telescope case. The databases between the center and sites are synchronized in some way but are not fully synchronized. Therefore, the synchronization bridge must be customized. The current solution is starting a Restful server at the center, and each site accesses this server to retrieve and post necessary updates.

In some cases, the human–computer interaction (HCI) is still needed, such as system maintenance or single node operation, especially when the operator wants to check the large image interactively. We implement optional user interface (UI) from command line to graphic and web interface by `cmd`. CUI `PySimpleGUI` and `WebSockets`.

For end users and operators who want to access multiple telescopes, site managers and databases at the same place, aggregated web pages are very practical. Users need just a browser to accomplish it. As mentioned before, Python is powerful enough in multiple areas including serving the web. We choose `Django` from several famous frameworks. It has many apps to make the development easier. This web page primarily serves telescope checking and basic controlling, site environment reviewing, power and dome controlling, and plans adding viewing and modifying.

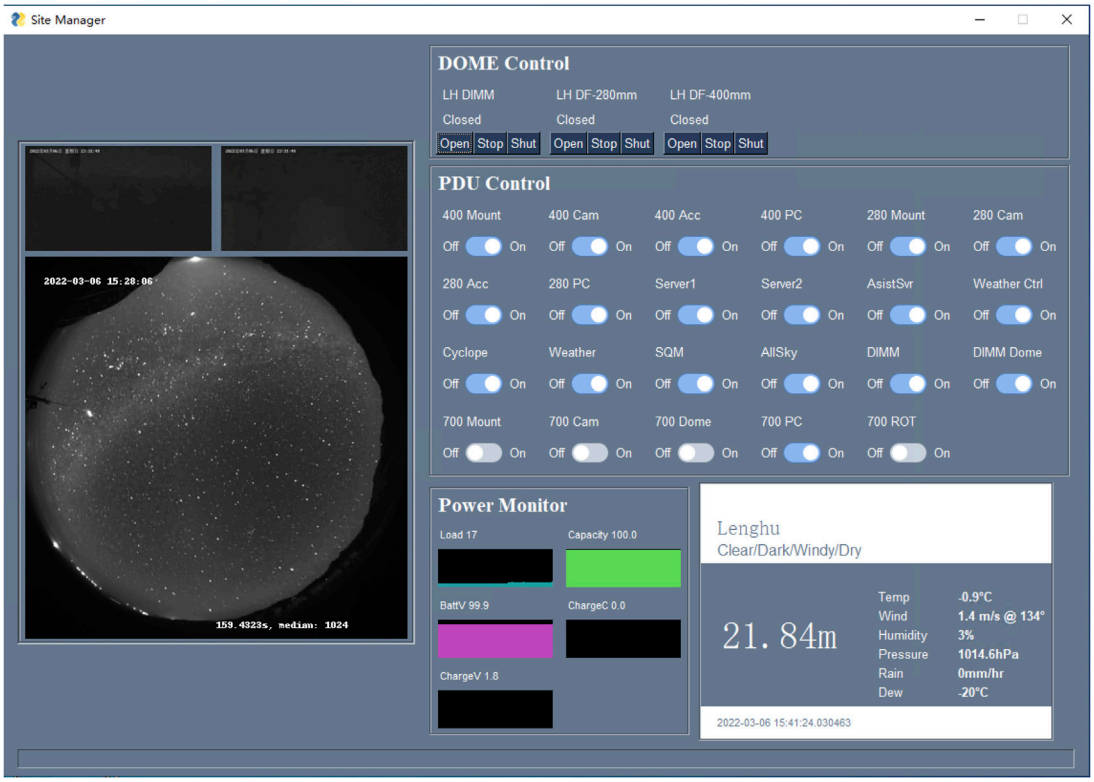


FIGURE 9
CHES site manager UI was used to show the webcam, all sky cam, solar power and weather, and to control the dome and power distribution unit. Each part of this UI is shown only if the related device is presented in the configuration, to keep the code universal for different site as much as possible. The similar UI is implemented in the Django website and communicate with this manager *via* websocket.

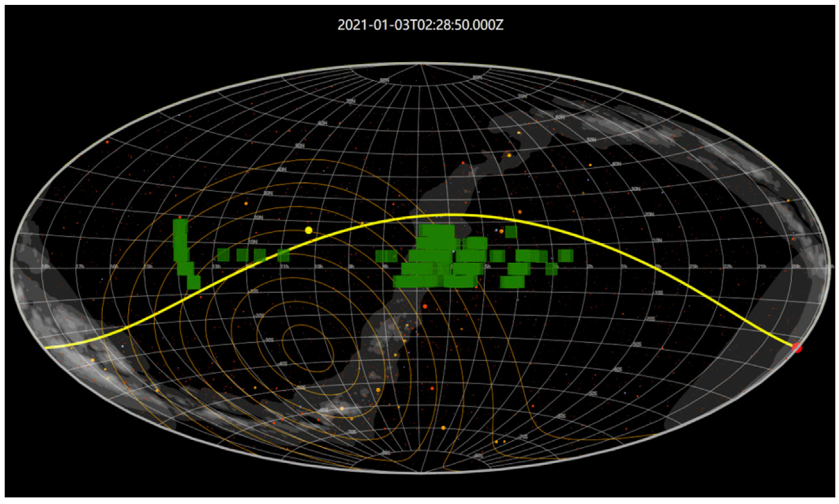


FIGURE 10
A sample survey plan for CHES-SA. The green patches are field of view of the telescope. The observation order is arranged by the movement of twilight terminator and Moon. The survey region is typically five belts around the equator with 6° wide based on the telescope FoV.

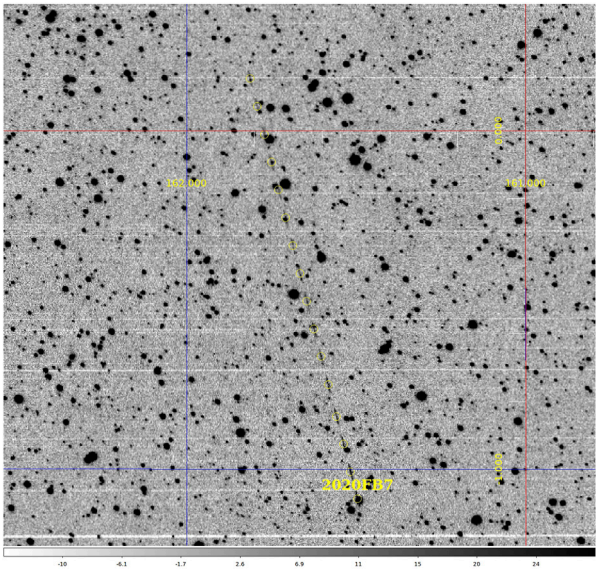


FIGURE 11
The stacked image of 2020FB7 near-Earth asteroid detected by the CHES-YA 280-mm wide field survey. The moving rate is faster than GEO objects and detected as new debris candidate. The following reduction confirmed it as a flyby asteroid.

Site support system

Site management plays an important role in robotic observation. In addition the common features such as manual device control and environment monitor, it can consolidate all information to make decision of observability. The manager should access different kinds of devices, such as.

- APC remote power distribution unit, to control the power supply of each device;
- Boltwood Cloud Sensor and Davis Weather Station, to collect the temperature, wind and cloud metrics;
- All Sky Monitor to record visible sky image;
- SQM sky quality meter to measure the brightness of the night sky;
- Cyclope seeing monitor to measure the seeing;
- Solar power monitor, to retrieve the status of panel and battery;
- Webcams.

Site manager access these devices *via* ASCOM, serial port, TCP/IP socket, SNMP and other methods. Each type of device is operated in individual thread to gather the information then converge to main thread. The architecture is shown in Figure 8. An analytic thread reduces these data by simple threshold and machine learning to do the decision, then feedback the observability as ASCOM Alpaca SafetyMonitor. Clients can use Restful links or COM object to get the value. Also, the reduced data will be downsized and recorded to the database.

This manager is also a terminal. There is a PySimpleGUI powered GUI and a WebSockets powered server. This websocket server will interact with the web page's websocket client, so that all features can be available on the website. The GUI of the manager shown in Figure 9.

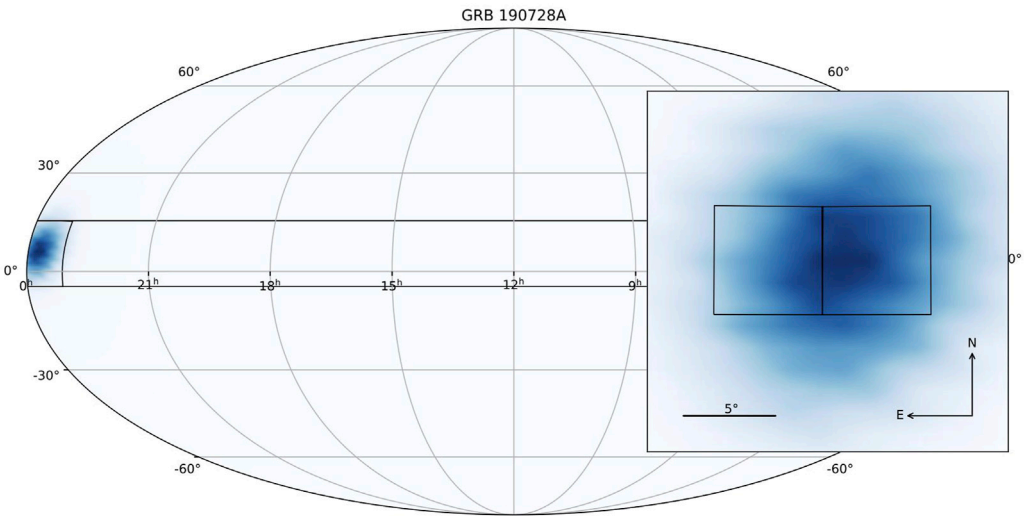


FIGURE 12
This is the searching fields of GRB 190728A event. Based on this, a fast searching plan for this event is carried out by the CHES-YA 280-mm wide field survey.

Summary

Current status

Currently, this software has worked routinely on several sensors, including:

- CHES-YA telescope array at the Yaoan site;
- CHES-SA at the Bonnie Vale site;
- CHES-ES with a collaborative Celestron RASA 11 at the El Sauce site;
- Dragonfly-LH, Dragonfly-MF, and Dragonfly-TU at Lenghu Muztagata and Tuamasaga sites separately;
- CANDLE-XY at the Xuyi site;
- CANDLE-AL at the Ngari site.

These telescopes share the same data center for unified coordination and data aggregation.

A GEO region survey, such as in Figure 10, will be carried out three times per night routinely, acquiring thousands of higher orbital tracklets belonging to hundreds of objects by using one large FoV telescope. Using CHES-YA array, wider field can be covered to get more data of up to more than 2000 objects.

During the survey, other objects will be captured too. In the observation of 2020-03-31, CHES-YA got a new bright NEO in the routine survey; the stacked image is shown in Figure 11. It is confirmed as 2020FB7 whose magnitude fell from 11^m to 20^m in 2 days and then disappeared. The first measurement from CHES-YA is earlier than the public data.

The system can receive other observation requests and make proper plan for each sensor. For example, a custom plan generator which is triggered by the gamma-ray coordination network (GCN) is implemented to follow important events. The upstream generator produce a searching area based on the event position estimation and a galaxy catalog, as shown in Figure 12. Followed by this procedure, LIGO/Virgo S191216ap (Sun et al., 2019b), GRB 191122A (Sun et al., 2019a), and GRB 190728A events had been followed by CHES-YA for optical counterpart searching.

Another scientific goal of interest is solar bodies' occultations for faint objects using high-precision positioning. Thanks to the wide distribution of telescopes, there may be greater probability of capturing these events. Several observations have been carried out including Pluto (2020-06-06), Callisto (2020-06-20), Varuna (2020-12-27), and Quaoar (2022-05-17) occultation events.

Conclusion

In this article, the software techniques of a robotic large-field survey network has been discussed in depth, especially the features related to RSO observation and the unified architecture to support various telescope configurations.

The mentioned techniques of software architecture, programming pattern, data model, and network topology are implemented for the CHES-YA telescope array and distributed to other facilities. The system function can satisfy the optical survey needs for RSOs, solar bodies, transient events, and the hybrid coordination of them. Also, currently, the software kit is still in development actively for more telescope support and function such as closed loop tracking and AI observation sequence optimization.

Data availability statement

The raw data supporting the conclusion of this article will be made available by the authors, without undue reservation.

Author contributions

ChZ contributed to conception and design of the software and implemented the local application. CaZ contributed to the user interface design and implemented the web part of the system. ChZ wrote the manuscript. All authors contributed to manuscript revision, read, and approved the submitted version.

Funding

This work was funded by the Basic Research Program of Jiangsu Province, China (Grant No. BK20201510) and the National Natural Science Foundation of China (Grant No. 12073082).

Acknowledgments

The author would like to thank the CHES project team for supporting this development, the technical team for the hardware support, and the staff of Yaoan site for the operation of the CHES telescope array.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated

organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Andersen, M., Grundahl, F., Christensen-Dalsgaard, J., Frandsen, S., Jørgensen, U., Kjeldsen, H., et al. (2019). *Hardware and software for a robotic network of telescopes-song*. arXiv preprint arXiv:1901.08300.

Boër, M., Atteia, J., Barat, C., Niel, M., Olive, J., Chevalier, C., et al. (1996)., 384. American Institute of Physics, 594–597.The tarot project: An optical glance at grbs*AIP Conf. Proc.*

Boër, M., Klotz, A., Laugier, R., Richard, P., Pérez, J. C. D., Lapasset, L., et al. (2017). "Tarot: A network for space surveillance and tracking operations," in *7th European conference on space debris ESA/ESOC*.

Brown, T., Baliber, N., Bianco, F., Bowman, M., Burleson, B., Conway, P., et al. (2013). *Las cumbres observatory global telescope network*, 125. Bristol, England: Publications of the Astronomical Society of the Pacific, 1031.

Castro-Tirado, A. J., Gorosabel, J., Hudec, R., Soldán, J., Bernas, M., Pata, P., et al. (1998). "The status of the burst observer and optical transient exploring system (bootes)," in *AIP conference proceedings* (American Institute of Physics), 428, 874–878.

Chen, Z., and Changyin, Z. (2021). Activities of pmo optical space debris survey. *An. Acad. Bras. Cienc.* 93, e20200827. doi:10.1590/0001-3765202120200827

Delrez, L., Gillon, M., Queloz, D., Demory, B.-O., Almleaky, Y., de Wit, J., et al. (2018). "Speculoos: A network of robotic telescopes to hunt for terrestrial planets around the nearest ultracool dwarfs," in *Ground-based and airborne telescopes VII* (Bellingham, WA: International Society for Optics and Photonics), 10700, 107001I.

Dyer, M. J., Dhillon, V. S., Littlefair, S., Steeghs, D., Ulaczyk, K., Chote, P., et al. A telescope control and scheduling system for the gravitational-wave optical transient observer (goto). *Observatory Operations Strategies, Process. Syst. VII* (2018)., 10704. Bellingham, WA: International Society for Optics and Photonics, 107040C.

Dyer, M. J., Steeghs, D., Galloway, D. K., Dhillon, V. S., O'Brien, P., Ramsay, G., et al. (2020). "The gravitational-wave optical transient observer (goto)," in *Ground-based and airborne telescopes VIII* (Bellingham, WA: International Society for Optics and Photonics), 11445, 114457G.

Fraser, S., and Steele, I. A. (2004). Robotic telescope scheduling: The liverpool telescope experience. *Optim. Sci. Return Astronomy through Inf. Technol.* 5493, 331–340.

Greenfield, P., Droettboom, M., and Bray, E. (2015). Asdf: A new data format for astronomy. *Astronomy Comput.* 12, 240–251. doi:10.1016/j.ascom.2015.06.004

Grundahl, F., Christensen-Dalsgaard, J., Kjeldsen, H., Frandsen, S., Arentoft, T., Kjaergaard, P., et al. (2008). Song–stellar observations network group. *Proc. Int. Astron. Union* 4, 465–466. doi:10.1017/s174392130802351x

Hickson, P. (2019). *Ocs : A flexible observatory control system for robotic telescopes with application to detection and characterization of orbital debris*.

Klotz, A., Boër, M., Eysseric, J., Damerdji, Y., Laas-Bourez, M., Pollas, C., et al. (2008). *Robotic observations of the sky with tarot: 2004–2007*, 120. Bristol, England: Publications of the Astronomical Society of the Pacific, 1298.

Kouprianov, V., and Molotov, I. (2017). *Forte : Ison robotic telescope control software*, 18–21.

Kubánek, P., Jelínek, M., Nekola, M., Topinka, M., Štrobl, J., Hudec, R., et al. (2004). "RTS2 - remote telescope system, 2nd version," in *Gamma-ray bursts: 30 Years of discovery*. Editors E. Fenimore and M. Galassi (American Institute of Physics Conference Series), 727, 753–756. doi:10.1063/1.1810951

Molotov, I., Agapov, V., Titenko, V., Khutorovsky, Z., Burtsev, Y., Guseva, I., et al. (2008). International scientific optical network for space debris research. *Adv. Space Res.* 41, 1022–1028. doi:10.1016/j.asr.2007.04.048

Ocaña, F., Ibarra, A., Racero, E., Montero, Á., Doubek, J., and Ruiz, V. (2016)., 9906. SPIE, 2135–2143.First results of the test-bed telescopes (tbt) project: Cebreros telescope commissioning*Ground-Based Airborne Telesc. VI*. Bellingham, WA.

Astropy CollaborationPrice-Whelan, A. M., Sipőcz, B. M., Günther, H. M., Lim, P. L., Crawford, S. M., et al. (2018). The astropy project: Building an open-science project and status of the v2.0 core package. *Astron. J.* 156, 123. doi:10.3847/1538-3881/aabc4f

Racero, E., Ocana, F., Ponz, D. and TBT Consortium (2015). Towards an autonomous telescope system: The test-bed telescope project. *Highlights Span. Astrophysics* 8, 828–833.

Sun, T., Zhang, C., Ping, Y., and Wu, X. (2019a). Grb 191122a: Ches optical observations. *GRB Coord. Netw.* 26271, 1.

Sun, T., Zhang, C., Ping, Y., and Wu, X. W. (2019b). Ligo/virgo s191216ap: No candidates found in ches observations of hawc error region. *GRB Coord. Netw.* 26487, 1.

Weiner, B. J., Sand, D., Gabor, P., Johnson, C., Swindell, S., Kubánek, P., et al. (2018)., 10704. International Society for Optics and Photonics, 107042H.Development of the Arizona robotic telescope network*Observatory Operations Strategies, Process. Syst. VII*. Bellingham, WA.

Wells, D. C., Greisen, E. W., and Harten, R. H. (1981). Fits - a flexible image transport system. *Astronomy Astrophysics Suppl.* 44, 363.

Appendix: Plan sample in the JSON format

```
{
    "Bias": 9,              //Taking 9 bias frames median stacking at start
    "Mode": "obsCoords",    //Working mode as function name
    "Coord": {              //Observation coordinate as Astropy SkyCoord
        "Frame": "icrs",    //Coordinate frame
        "Value1": "60d",    //Coordinate value with unit as string
        "Value2": "45d"
    },
    "Tracking": "Sidereal", //Tracking mode
    "CamMode": {            //Camera mode that should be set at start
        "ReadoutMode": 0,   //Readout mode defined in API
        "BinX": 2,          //Pixel binning
        "BinY": 2,
        "StartX": 10,       //Subframe origin
        "StartY": 10,
        "NumX": 50,         //Subframe size
        "NumY": 50
    },
    "Exposure": 1.0,        //Exposure time in seconds
    "Filter": 0,            //Filter number defined in API
    "Frames": 3,            //Frames count that should acquire
    "Frame3": {             //Set action before taking third frame
        "Exposure": 2.0     //Any setting before this frame
    }
}
```

frontiers | Frontiers in Astronomy and Space Sciences

TYPE Original Research
PUBLISHED 03 October 2022
DOI 10.3389/fspas.2022.895732

OPEN ACCESS

EDITED BY
Frederic Victor Hessman,
Georg-August-Universität Göttingen, Germany

REVIEWED BY
Norbert Zacharias,
United States Naval Observatory (USNO), United States
Jean Baptiste Marquette,
UMR5804 Laboratoire d'astrophysique de Bordeaux (LAB), France

*CORRESPONDENCE
Tobias Hoffmann,
tobias.hoffmann2@uni-oldenburg.de

SPECIALTY SECTION
This article was submitted to Astronomical Instrumentation, a section of the journal Frontiers in Astronomy and Space Sciences

RECEIVED 14 March 2022
ACCEPTED 26 July 2022
PUBLISHED 03 October 2022

CITATION
Hoffmann T, Gehlen M, Plaggenborg T, Drolshagen G, Ott T, Kunz J, Santana-Ros T, Gedek M, Reszelewski R, Żołnowski M and Poppe B (2022), Robotic observation pipeline for small bodies in the solar system based on open-source software and commercially available telescope hardware.
Front. Astron. Space Sci. 9:895732.
doi: 10.3389/fspas.2022.895732

Robotic observation pipeline for small bodies in the solar system based on open-source software and commercially available telescope hardware

Tobias Hoffmann[1,2]*, Matti Gehlen[1], Thorsten Plaggenborg[3], Gerhard Drolshagen[1], Theresa Ott[1], Jutta Kunz[2], Toni Santana-Ros[4,5], Marcin Gedek[6], Rafał Reszelewski[6], Michał Żołnowski[6] and Björn Poppe[1]

[1]Department of Medical Physics and Acoustics, University of Oldenburg, Oldenburg, Germany, [2]Institute of Physics, University of Oldenburg, Oldenburg, Germany, [3]Faculty of Mathematics and Science, University of Oldenburg, Oldenburg, Germany, [4]Departamento de Física, Ingeniería de Sistemas y Teoría de la Señal, Universidad de Alicante, Alicante, Spain, [5]Institut de Ciències del Cosmos, Universitat de Barcelona, Barcelona, Spain, [6]6 Remote Observatories for Asteroids and Debris Searching (6ROADS), Cracow, Poland

The observation of small bodies in the Space Environment is an ongoing important task in astronomy. While nowadays new objects are mostly detected in larger sky surveys, several follow-up observations are usually needed for each object to improve the accuracy of orbit determination. In particular objects orbiting close to Earth, so called Near-Earth Objects (NEOs) are of special concern as a small but not negligible fraction of them can have a non-zero impact probability with Earth. Additionally, the observation of manmade space debris and tracking of satellites falls in the same class measurements. Telescopes for these follow-up observations are mainly in a aperture class between 1 m down to approximately 25 cm. These telescopes are often hosted by amateur observatories or dedicated companies like 6ROADS specialized on this type of observation. With upcoming new NEO search campaigns by very wide field of view telescopes, like the Vera C. Rubin Observatory, NASA's NEO surveyor space mission and ESA's Flyeye telescopes, the number of NEO discoveries will increase dramatically. This will require an increasing number of useful telescopes for follow-up observations at different geographical locations. While well-equipped amateur astronomers often host instruments which might be capable of creating useful measurements, both observation planning and scheduling, and also analysis are still a major challenge for many observers. In this work we present a fully robotic planning, scheduling and observation pipeline that extends the widely used open-source cross-platform software KStars/Ekos for Instrument Neutral Distributed Interface (INDI) devices. The method consists of algorithms which automatically select NEO candidates with priority according to ESA's Near-Earth Object Coordination Centre (NEOCC). It then analyses detectable objects (based on limiting magnitudes, geographical position, and time) with preliminary ephemeris from the Minor Planet Center (MPC). Optimal observing

slots during the night are calculated and scheduled. Immediately before the measurement the accurate position of the minor body is recalculated and finally the images are taken. Besides the detailed description of all components, we will show a complete robotic hard- and software solution based on our methods.

KEYWORDS

robotic telescopes, near-earth objects, minor planets, space environment, software, observatory, open-source

1 Introduction

Follow-up observations and tracking of fast-moving small objects in the Space Environment are an important task in modern astronomy. Scientific knowledge can be gained in particular from a deeper understanding of the dynamics of small bodies in the Solar System and from the interaction with the gravitational influences of the Sun and the planets (Koschny et al., 2017). Furthermore, potential impacts of sufficiently large objects pose a danger for the Earth or space missions (Rumpf et al., 2016). Due to the immense increase of space missions in the last years (new-space-era) also follow-up observations of space-debris play a more and more important role in this field of astronomy.

Objects that may come close to Earth are called *Near-Earth Objects* (NEOs). In 2005, NASA was given a mission to find and track 90% of all Potentially Hazardous Asteroids (PHA) by the end of 2020. The significantly increased efforts resulted in an exponential increase in the number of objects discovered in the following years.[1] Ground-based search campaigns like the Vera C. Rubin Observatory or the Flyeye telescope (ESA) and space missions like the NEO surveyor space mission (NASA) are expected to increase the number of discovered objects in the next years significantly (Cibin et al., 2019; Mainzer et al., 2021). Data are collected internationally in a centralized manner by the *Minor Planet Center* (MPC) and made available for further evaluations.[2]

The detection and tracking of the observable, generally called "small objects," is therefore a key challenge for international risk assessment and planetary defense (Rumpf et al., 2016; Mainzer et al., 2021). To support the task of follow-up observations, observatories all around the world observe Minor Planets (Figure 1). Typically, telescopes up to 2 m in diameter are used here.

However, the number of available observatories is still too small compared to the necessary number of follow-up observations. Amateur astronomers often operate small observatories, whose instruments would in principle be able to perform meaningful measurements. Even though some sites are already submitting measurements to the MPC, a large part of this potential remains unused. Among other things, we see one of the major reasons in the relatively complicated object selection, where many parameters (ephemerides, brightness, location of the observatory, limiting magnitude, . . .) must be considered, some of which change dynamically, in order to carry out a successful measurement.

However, many robotic telescope systems and networks for NEO follow-up observations also exist (e.g., Las Cumbres Observatory Global Telescope (LCOGT, (Shporer et al., 2010))) or are under construction (Lister et al., 2015; Dotto et al., 2021), but there are no commercially available solutions for amateur or small professional equipment (e.g., for university observatories) so far for that specific use, if then only in parts (Gupta et al., 2015; García-Lozano et al., 2016). In the recent years, powerful software and hardware enhancements have been developed, especially for the amateur sector, which make it possible to operate even small observatories worldwide robotically or at least remotely.

In this work we will describe a viable modular solution with commercially available hardware and software that makes it possible to perform follow-up observations of Minor Planets and thus contribute to their improved orbit prediction. The robotic system consists of a planning, scheduling and observation pipeline that is based on the open-source software KStars/Ekos with INDI devices.[3] It automatically obtains the objects and their position data from ESA and the MPC, constantly makes updates on the ephemeris, controls the observatory and automatically schedules different objects for observing nights.

2 Observational targets

2.1 Small Solar System bodies

The International Astronomical Union (IAU) defines small bodies of the Solar System as all objects apart from planets, dwarf planets and (natural) satellites orbiting the Sun (IAU General Assembly, 2006). Therefore, all Comets and Minor Planets

1 https://cneos.jpl.nasa.gov/stats/totals.html

2 www.minorplanetcenter.net/iau/mpc.html

3 www.indilib.org/what-is-indi/discover-indi.html

(without dwarf planets) belong to this category. Within this group Asteroids and Comets are the major types of Small Solar System Bodies (SSSB). Due to their low mass, their physical and orbital behavior like the shape and the orbit stability can be quite different from planets.

The study of these objects is of great interest because gravitational and non-gravitational perturbations and collisions may lead to a change of orbit (Bottke et al., 2006). It turns out that the size distribution follows an inverse power law, so there are far more smaller objects than larger ones, which makes most of them quite difficult to detect because they are so faint (Peña et al., 2020). Since SSSBs are either remnants from the formation of the Solar System or fragments of a collision, it makes them scientifically relevant for the understanding of our Solar System (Hestroffer et al., 2019).

2.2 Near-Earth Objects

The movement of the SSSBs is highly dynamic. Up to date, we know more than 1 million Minor Planets, the majority of them orbiting between Mars and Jupiter in the Main Belt. However, orbital resonances and disturbances can bring some of these objects to inner orbits approaching Earth. SSSBs that have a perihelion of less than 1.3 AU are defined as Near-Earth objects (NEOs). The term Near-Earth Asteroid is also used because most NEOs are Asteroids and only a small fraction are Comets. So far, nearly 30.000 Near-Earth Asteroids have been discovered, with increasing numbers daily.[4] The minimum orbit intersection distance (MOID) is the minimum distance between the orbits of two objects. If such an NEO has a size of more than 140 m and a MOID of less than 0.05 AU to the Earth's orbit, it is classified as a Potentially Hazardous Object (PHO, (Huebner et al., 2009)). Currently, there are more than 2.000 of these PHO known.[5] Even if being smaller, some NEOs can be a threat for our planet. For example, the Chelyabinsk meteor event in 2013, which was an asteroid with an estimated effective diameter of about 18 m, led to many injured people.[6] According to current research, such an event is possible about every 50 years (Boslough et al., 2015).

To detect such objects early in advance, sky surveys for NEOs like the Panoramic Survey Telescope & Rapid Response System (Pan-STARRS) or the Catalina Sky Survey are regularly scanning the sky for new objects (Larson et al., 1998; Hodapp et al., 2004).

2.3 Minor Planet Center

The Minor Planet Center (MPC), which is under the guidance of the International Astronomical Union (IAU), is in charge of the worldwide organization of all data of SSSBs, especially NEOs and PHOs.[7] It collects astrometric and photometric data from the individual observatories, combines and processes them to calculate the orbits.

Every observatory that contributes positional data of Minor Planets to the MPC has an assigned individual code, consisting of three characters, in the form of a combination of letters and numbers. This MPC code is needed in advance to submit the data. It is assigned to an observatory when its initial submission with specified requirements is accepted.[8] The positional accuracy of the submitted objects is expected to be within two arcsec compared to the predicted ephemeris. There is a specific format for reporting the measurements, which is already implemented in some evaluation software (e.g., *Astrometrica*[9]).

2.4 NEO search campaigns

NEO candidates, that have recently been discovered, require confirmation by follow-up observations of other observatories (Micheli et al., 2015). They allow to confirm that the object is real and has the appropriate perihelion distance (Seaman et al., 2021). Further measurements will then be needed to improve the accuracy of the ephemeris, which is needed for further investigations on the object's physical properties. NEOs are usually only observable during their close approach to Earth, meaning that they can be lost if their ephemeris has a large uncertainty due to observations with low precision or only from a small orbital arc (Micheli et al., 2014). Additionally, perturbation by gravitational forces of other objects or non-gravitational forces like absorption and emission of radiation are increasing the uncertainty (Bottke et al., 2006; Perna et al., 2013). The targeted recovery of such lost object is so difficult, such that it rather will be found again by ordinary survey observations (Milani, 1999). This shows that sufficient follow-up observations over as long a period as possible are of importance to minimize the risk for lost objects and clarify the future impact possibilities (Micheli et al., 2014).

The MPC recommends observers to generally make two or three measurements per object per night.[10] It is sufficient if the observations are made over the period of a few hours. To remove ambiguity, this procedure should be repeated on another nearby night. It is explicitly not necessary to make more than three measurements for one object per night. For a potential new discovery, it is reasonable to make measurements on groups some hours apart on a single night. For follow-up observations of

4 https://cneos.jpl.nasa.gov/stats/totals.html
5 https://cneos.jpl.nasa.gov/stats/totals.html
6 https://cneos.jpl.nasa.gov/news/fireball_130301.html
7 www.minorplanetcenter.net/iau/mpc.html
8 www.minorplanetcenter.net/iau/info/Astrometry.html
9 www.astrometrica.at
10 www.minorplanetcenter.net/iau/info/Astrometry.html

TABLE 1 Components and specifications of the telescopes "GHOST" and "ORT" of the University Observatory of Oldenburg (MPC Code: G01).

Telescopes	*GHOST*	*ORT*
OTA	16-inch, 3250 mm (f/8) Ritchey-Chrétien Telescope	6-inch, 420 mm (f/2.8) Corrected hyperbolic Astrograph
Mount	High-precision GoTo GEM with absolute encoders	GoTo GEM
Focus	Temperature compensated	Temperature compensated
Filters	7 × 2-inch: L, R, G, B, SII, Ha, OIII	8 × 1.25-inch: L, Photometric BVRI, SII, Ha, OIII
Camera	Cooled b/w CCD camera, 17.6 mm × 13.52 mm, 5.4 µm pixels	Cooled b/w CMOS camera, 13.2 mm × 8.8 mm, 2.4 µm pixels
Field-of-View	0.31° × 0.24°	1.80° × 1.20°
Pixel scale	0.34"/pixel (1 × 1 Binning)	1.18"/pixel (1 × 1 Binning)
Add. Scope	50 mm, 205 mm (f/4.1) Refractor (for Alignment)	50 mm, 190 mm (f/3.8) Refractor (for Guiding)
Dome	3.0 m diameter, 0.5 m shutter controlled and powered wirelessly	0.8 m diameter, two-part folding dome

a new discovery, it is recommended to make measurements on pairs of nearby nights for every seven to 10 days repeatedly as long as the object is visible and unidentified. So, it is not necessary to observe a new object each night. If an object has already passed through several oppositions, measurements should be made on pairs on nearby nights around each opposition.

According to our analysis of the MPC database in 2021, for the initial discovery there are on average 21.9 ± 8.6 measurements needed to allow a clear and reliable classification. However, several (most ground-based) follow-up observations in much larger numbers in appropriate time intervals are usually necessary for sufficient orbit calculations over a longer period (Vereš et al., 2018).

2.5 Object selection

Due to the variety of observing strategies of NEOs, efficient methods are needed to make the best use of the observation time.[11] This can be done by prioritizing certain objects that benefit more from further observations than other (Micheli et al., 2015). A protocol was established for the ESA's NEOCC Priority List that selects NEOs for follow-up observations for which the improvement of orbital accuracy will be maximized with only minimal observing efforts (Boattini et al., 2007). Besides the MOID and the object's Sky Uncertainty (SU), an estimate of the difficulty to recover an object depending on the visual magnitude and further parameters are considered in this list. By combining these factors, the according urgencies are categorized into priority classes.

The Priority List can be accessed by an automated HTTP GET request.[12] The list contains additional information about the object's positions, physical characteristics, and orbital uncertainties. The individual entries can be used for further processing. For example, objects can be sorted out according to own requirements for urgency and visual magnitudes.

Besides the NEOCC Priority List there are also other ways to select suitable objects for observation nights. The NEA Observation Planning Aid (NEAObs) of the MPC makes a user-orientated approach and creates a list of suitable objects that fulfills the user's criteria (e.g., magnitude, motion, and uncertainty ranges) and are observable for the site.[13] The NEO Confirmation Page (NEOCP) contains the current NEO candidates that need confirmation for the discovery. Both can be accessed similar to the Priority List.

3 Instrumentation and implementation

3.1 Robotic Telescope instruments

The two Robotic Telescopes used in this work are located at the University Observatory of Oldenburg (08° 09' 55.0" E, 53° 09' 10.3" N, Elevation: 22 m above sea level, MPC Code: G01). The main telescope, called "**G**roßes **H**aupteleskop der **O**ldenburger **St**ernwarte" (GHOST), is a 16-inch f/8 Ritchey-Chrétien telescope used for fainter and small-sized astronomical objects. The smaller 6-inch f/2.8 corrected Astrograph, the "**O**ldenburg **R**obotic **T**elescope" (ORT), is used for fast wide-field imaging. Table 1 lists the detailed components and specifications of both telescopes. All hardware is commercially available.

The methods are developed and tested at the GHOST telescope in Oldenburg. The KAF8300 (onsemi[14], Phoenix, AZ, United States) chip of the camera is set in a 2 ×

11 https://neo.ssa.esa.int/priority-list

12 https://neo.ssa.esa.int/priority-list

13 www.minorplanetcenter.net/cgi-bin/neaobs.cgi

14 www.onsemi.com/

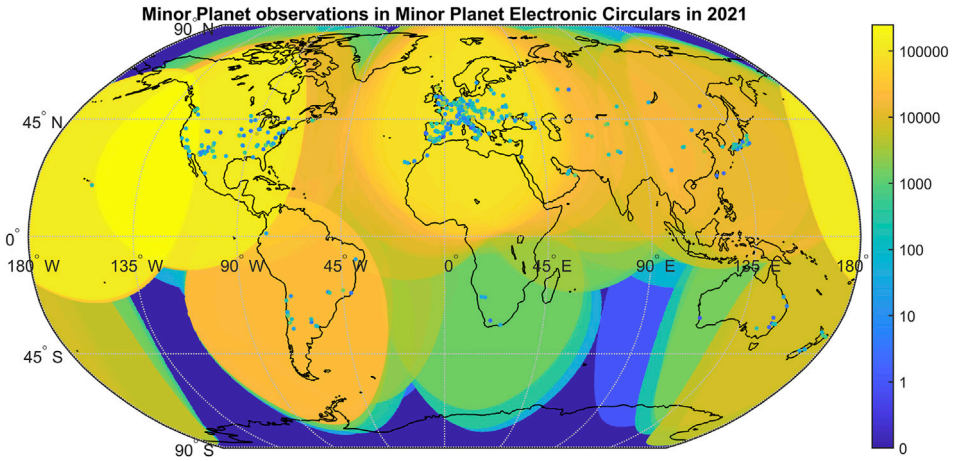


FIGURE 1
Number of observations published in the Minor Planet Electronic Circulars (MPECs) in 2021. The individual MPC observatories are mapped as points on their geographic coordinates with their number of observations as color. Observatories with no observations in 2021 are removed. The cumulated observations within an arc of 45° of the geographic coordinates is contoured.

2 Binning for NEO observation with an exposure time of t_{exp} = 60 s. The bitrate is 16 bits with a full well of 25,000 e^-, the gain is g = 0.41 e^-/ADU (analog to digital unit) and the mean Quantum Efficiency is $QE = 0.54$ (for wavelengths $\lambda = (550 \pm 150)$ nm). The bias of the chip is measured to be $N_{bias} = 250$ e^-. According to the chip's data sheet, the readout current is $N_{readout} = 7$ e^- and the dark current is 0.1 e^-/s at −10°C with a doubling temperature of 5.8°C. For other temperatures the dark current N_{dark} at temperature T can be calculated with:

$$N_{dark} = t_{exp} \cdot 0.1 \frac{1}{s} \cdot 2^{\frac{T+10^\circ C}{5.8^\circ C}} \quad (1)$$

where a temperature of $T = -20$°C is usually used. All the values N refer to the corresponding number of electrons measured per pixel. The used clear glass filter has, according to its data sheet, a transmission of at least 98%. The main and secondary mirror of the telescope each have a dielectric high-reflectivity coating with at least 92% transmission. In total this results in a transmission of the optical tube of $\tau = 0.98 \cdot 0.92^2 \approx 0.83$. The secondary mirror itself has a diameter of 191 mm, thus resulting in a relative obstruction $a_{obstr} = 0.2213$ of the light blocked for the primary mirror.

As an illustration of the number of expected follow-up observations, the presented methods are compared for different sites and observatories. For this purpose, we use the observatories of the 6ROADS network (Remote Observatories for Asteroids and Debris Searching)[15]. The network consists of six observatories with different sky qualities and telescope diameters/power classes. Additionally, the data for the ESA *Optical Ground Station* (OGS)[16] is added to the list since it is one of the most active telescopes in this field. Additional data on the telescopes are summarized in Table 2.

3.2 Software

The robotic telescope's software is based on *INDI Library* (Instrument-Neutral-Distributed-Interface) components. The system uses the INDI protocol to control the hardware, automate processes, collect data and exchange information among the devices and the software front-ends.[17] INDI consists of drivers to control the astronomical equipment of an observatory, a server as a central hub in between the drivers/devices and the clients/software, which can be accessed within the network, and the clients itself. One of the most generic GUI clients to control the devices is *KStars*. In Figure 2, the connection of the different INDI components from the client to the individual devices is schematically described. There are several more clients in INDI, like the *DBus Interface*, which can be used additionally to make automations for the observatory. It is fully scriptable and can control all devices with an interface. Together with an automated script, which also can connect by itself to the INDI server, but on a more fundamental way, it can be used for a Robotic control system.

15 https://6roads.com.pl/

16 https://sci.esa.int/web/sci-fmi/-/36520-optical-ground-station

17 www.indilib.org/what-is-indi/discover-indi.html

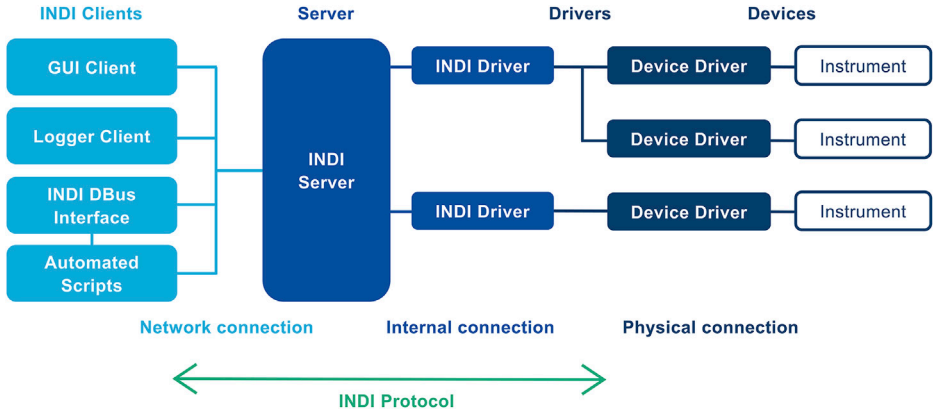


FIGURE 2
Schematic representation of the operation principle of INDI from typical clients to the different telescope instruments with the INDI Server and Drivers controlled by the INDI Protocol and their connection types.

KStars as one of the most common GUIs is an open-source cross-platform Astronomy Software.[18] In addition to the control over the INDI devices, it has capabilities for observation planning and graphical simulations of the night sky. The database contains up to 100 million stars, 13,000 deep-sky objects, all planets etc. Catalogs for Minor Planets or other objects can also be extended individually by adding external data from the MPC.

Ekos is the framework of KStars for data-acquisition and observatory control.[19] Due to its modular structure, it can be used for many automation processes. There are modules for the automatic capture-, focus-, mount slewing-, alignment- and guiding-process each. Additional accessories like weather monitoring and dome control can also be used in separate modules. All these modules can be automatically controlled *via* the Ekos Scheduler, which checks objects from a given list of targets for the current observing conditions and then makes a completely automatic and adjustable observation.

In addition to the optional startup and shutdown scripts, additional scripts can also be embedded before and after each observation of an object and each individual recording. This can be done by executable Python scripts. The INDI DBus Interface is implemented into Python.

After images are obtained by the robotic telescope system with Ekos, the positions and magnitudes of the object need to be analyzed. This astrometric and photometric data reduction can be done with the astrometry software *Astrometrica*.[20] The software can stack the images with a shift resulting from the expected movement of the object. For this, the orbital parameters from the MPC database are automatically obtained. This will improve the Signal-to-noise ratio and the distinction of the object from the stars as it prevents objects from becoming trails at long exposure times. The software then uses a Gaussian Point-Spread-Function (PSF) as fit function for data reduction in order to match reference stars (in our case from the Gaia DR2 catalog with the Gaia Broadband color band) in the picture (Raab, 2002). With that, the coefficients of the two-dimensional coordinate transformation, called plate constants, are calculated with a fourth-order polynomial fit. After extracting the astrometric and photometric measurements, the program allows to send a properly formatted MPC report.

3.3 Sky Brightness

For estimating the general brightness of the night sky a *Sky Quality Meter* (SQM) is used (Hänel et al., 2018). The detector measures the luminance in a field with a full width at half maximum (FWHM) of 20° near the zenith. The measurements are given in mag/arcsec^2, where this quantity corresponds approximately to the visual magnitude of the sky m_{sky}. A local deviation of the brightness can occur due to further influences (e.g. the Moon) and thus worsen the observation conditions. To minimize these influences, a minimum Moon distance of 30° is set.

With a field factor F, typically being a value between 1.4 and 2.4 according to the observer's visual capabilities and experience, Crumey (2014) derived the following expression for the limiting visual (naked eye) star magnitude m_0:

$$m_0 = 0.3834 \cdot m_{sky} - 1.4400 - 2.5 \log F \quad (2)$$

if $(20 < m_{sky} < 22)$ mag/arcsec^2. With m_0 a classification in the Bortle scale according to Bortle (2001) can be made.

18 https://edu.kde.org/kstars/

19 www.stellarmate.com/support/ekos/17-support/documentation/ekos.html

20 www.astrometrica.at

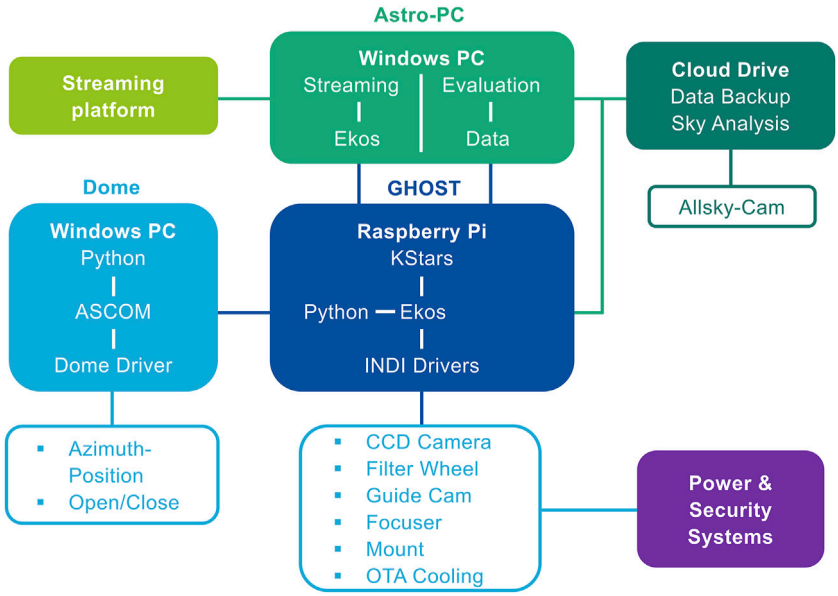


FIGURE 3
Schematic representation of the overall Telescope System "GHOST" including the controlling Raspberry Pi (StellarMate OS 64-bit v1.7.1, 2GB RAM), all components and their individual power and security systems, the separate PC controlling the Dome, which is connected to the Raspberry Pi over an SSH connection controlled by Python scripts. The telescope components are controlled *via* the INDI protocol (Figure 2) from KStars/Ekos to the INDI Drivers. Additional systems include the "Astro-PC" for Data Reduction and Livestreaming, and the Cloud Drive for storing image data and accessing the Sky Monitoring System, which is operated by an Allsky-Cam.

3.4 Estimation of imaging Limiting Magnitude

A crucial role for the automatic selection of observable objects is the imaging Limiting Magnitude (LM). It indicates up to which magnitude objects can still be observed and is mainly determined experimentally. However, it is useful to determine this value theoretically in order to estimate the influence of environmental and technical conditions. The LM depends on various telescope properties and settings. The following derivation is motivated on the calculations from Koschny and Igenbergs (2020). We start with the definition of the apparent magnitude different of two objects depending on the ratio of their flux densities Φ. When we consider one object as the Sun and the other as the observed object (e.g., NEO), the following is given:

$$m_{\mathrm{obj}} = m_{\mathrm{Sun}} - 2.5 \log\left(\frac{\Phi_{\mathrm{obj}}}{\Phi_{\mathrm{Sun}}}\right) \tag{3}$$

with the apparent magnitude m_{obj} of the observed object, $m_{\mathrm{Sun}} = -26.74$ mag of the Sun[21] and the corresponding energy flux densities Φ_{obj} and $\Phi_{\mathrm{Sun}} = 1361\ \mathrm{W/m^2}$ (Mamajek et al., 2015). Since the spectrum of the Sun extends over a wide range, but the telescopes only measure the visible spectrum from 400 to 700 nm, the energy flux density is reduced to $\Phi_{\mathrm{Sun}} = 535.5\ \mathrm{W/m^2}$ (Meftah et al., 2018). For a given magnitude of the object its flux can therefore be calculated. With that, the total power of the signal P_{cam} measured on the camera is calculated by multiplying the surface area of the telescope, depending on its diameter d_{tele} of the primary mirror and its relative obstruction by the secondary mirror a_{obstr}, and a total transmission rate of the telescope τ with the energy flux density Φ_{obj}:

$$P_{\mathrm{cam}} = \Phi_{\mathrm{obj}} \cdot \frac{\pi}{4} \cdot (d_{\mathrm{tele}})^2 \cdot \tau \cdot (1 - a_{\mathrm{obstr}}) \tag{4}$$

The integrated power of the signal over the exposure time t_{exp} will give us the total energy of the photons captured by the camera. To convert this into the photoelectrons released in the camera sensor, we divide this by the energy per photon and multiply it with the quantum efficiency QE of the camera. Assuming a mean wavelength of the photons $\bar{\lambda}$ and a constant power P_{cam}, we will get the number of electrons captured at the center of the signal N_{signal}:

$$N_{\mathrm{signal}} = QE \cdot p \cdot \frac{P_{\mathrm{cam}} \cdot t_{\mathrm{exp}}}{hc \Big/ \bar{\lambda}} \tag{5}$$

with p the percentage of the signal in the center pixel, Planck's constant h and the speed of light c. Besides the signal to be measured, other disturbances occur. In addition, the readout noise N_{readout}, the thermal noise N_{dark} and the camera's offset value N_{bias} need to be considered. The main disturbance of the

21 https://nssdc.gsfc.nasa.gov/planetary/factsheet/sunfact.html

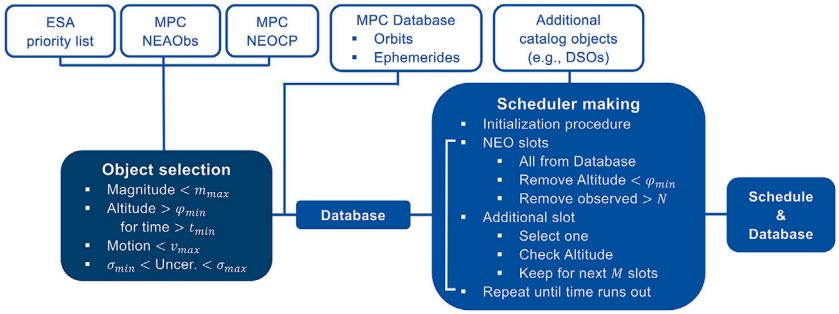


FIGURE 4
Pre-Observation Pipeline illustrated with its workflow. First suitable NEOs are selected from ESA's NEOCC Priority List, the MPC NEA Observation Planning Aid (NEAObs) and the MPC NEO Confirmation Page (NEOCP) with given boundaries on the magnitude, altitude, motion and sky uncertainty. Orbits and Ephemerides are downloaded from the MPC Database and converted for KStars. With the selected objects and additional catalog objects an observation schedule is automatically made in an optimal running order. The system continues with the Observation pipeline in Figure 5.

signal is the light from the empty sky N_{sky} measured at the camera. This value depends significantly on the location (suburban, rural sky, *etc.*). To quantify this lightness, we can use the already known visual magnitude m_{sky}. Eq. 3 is analogously considered with magnitudes m_{sky} and flux densities Φ_{sky} of the sky. Since the magnitude of the sky is given per arcsec^2, we need to multiply the resulting flux with the square area of one pixel with size s_{px}. Rearranged to the flux from the sky this results in:

$$\Phi_{\text{sky}} = s_{\text{px}}^2 \cdot \Phi_{\text{Sun}} \cdot 10^{\frac{m_{\text{Sun}} - m_{\text{sky}}}{2.5}} \tag{6}$$

The Eqs 4, 5 can also be used analogously, where p can be neglected, because the sky signal is equally distributed over the sensor. These results in the following:

$$P_{\text{cam, sky}} = \Phi_{\text{sky}} \cdot \frac{\pi}{4} \cdot (d_{\text{tele}})^2 \cdot \tau \cdot (1 - a_{\text{obstr}}) \tag{7}$$

$$N_{\text{sky}} = QE \cdot \frac{P_{\text{cam, sky}} \cdot t_{\text{exp}}}{hc \Big/ \bar{\lambda}} \tag{8}$$

Altogether, the signal-to-noise ratio *SNR* as in Merline and Howell (1995) can be calculated:

$$SNR = \frac{N_{\text{signal}}}{\sqrt{N_{\text{signal}} + n_{\text{px}}\left(1 + \frac{n_{\text{px}}}{n_{\text{b}}}\right)\left(N_{\text{bias}} + N_{\text{dark}} + N_{\text{sky}} + N_{\text{readout}}^2 + g^2\sigma_f^2\right)}} \tag{9}$$

with n_{px} pixels considered for measurement of the signal and n_{b} pixels for the measurement of the background signal. The additional term $g^2\sigma_f^2$, with the gain g and an estimate of the 1σ-error by the A/D converter of the camera $\sigma_f \approx 0.289$, indicate the error from the A/D conversion (Merline and Howell, 1995). In order to get the LM of a telescope, we need to make the calculation in Eq. 9 backwards. We presume a certain threshold for *SNR* and get the required object's signal value:

$$N_{\text{signal}} = \frac{SNR}{2}\left(SNR + \sqrt{SNR^2 + 4n_{\text{px}}\left(1 + \frac{n_{\text{px}}}{n_{\text{b}}}\right)\left(N_{\text{bias}} + N_{\text{dark}} + N_{\text{sky}} + N_{\text{readout}}^2 + g^2\sigma_f^2\right)}\right) \tag{10}$$

Using Eqs 3–5 we get for the magnitude m_{obj} of the object for a given *SNR*:

$$m_{\text{obj}} = m_{\text{Sun}} - 2.5\left[\log\left(\frac{2hc}{\pi\Phi_{\text{Sun}}}\right) - \log\left(t_{\text{exp}}QE\tau\bar{\lambda}p(1 - a_{\text{obstr}})(d_{\text{tele}})^2\right) + \log(SNR) + \log\left(SNR + \sqrt{SNR^2 + 4n_{\text{px}}\left(1 + \frac{n_{\text{px}}}{n_{\text{b}}}\right)\left(N_{\text{bias}} + N_{\text{dark}} + N_{\text{sky}} + N_{\text{readout}}^2 + g^2\sigma_f^2\right)}\right)\right] \tag{11}$$

Finally for the limiting magnitude of the telescope, we need to correct the object's magnitude from Eq. 11 with the atmospheric extinction causing a dimming of the light by the terrestrial atmosphere. With the air mass χ and the extinction coefficient κ this will lead to the following expression:

$$m_{\text{lim}} = m_{\text{obj}} - \kappa \cdot \chi \tag{12}$$

where, for an object in the zenith, $\chi = 1$ and $\kappa = 0.245$ for the V-band (Jurado Vargas et al., 2002). In addition, astronomical seeing, i.e. the broadening and blurring of point sources due to air turbulence in the atmosphere, is an important quantity affecting the LM. It attenuates the intensity of the light and thus increases the extinction coefficient κ due to increased scattering of light. On the other hand, the widening of the recorded light spot from the object causes a decrease in the percentage p of the signal in the center pixel. This gives us an expression for the LM of a telescope depending on its specifications and the requested *SNR*. Typically, a value of $SNR > 5$ is needed for the detection of an object.

3.5 Robotic Telescope system

Building on the existing INDI framework with KStars/Ekos as the interface, it was possible to create a robotic observation pipeline. All the observatory's equipment is connected to a small

single-board computer (Raspberry Pi 4B) *via* USB (Focus, Filter Wheel, Cameras) and Network (Mount, Dome). The Dome's shutter opening and closing is controlled directly *via* a Bluetooth connection. Figure 3 shows a general illustration of the observatory's components and accessories connected to the controlling computer of GHOST. An analogous Setup has been used for the ORT telescope.

With the hardware set up, an appropriate software is necessary for the use of the Robotic Telescope. The difficulty is, that most available software is not developed for the measurement of NEOs due to their relative movement in the equatorial coordinate system. Many programs like KStars support the observation of planets and Minor Planets, but only well-known numbered objects. Since orbits are continuously updated, follow-up observations of unnumbered, new or even unconfirmed objects require frequent connection to the latest data. We solved this problem by using the customization options of Ekos in the form of embeddable scripts to ensure a continuous update of the coordinates of the objects before each observation.

3.6 Pre-observation pipeline

Figure 4 shows the Pre-Observation Pipeline developed for the Robotic Telescope. It starts with the selection of suitable objects. For that, ESA's NEOCC Priority List and MPC's NEOCP is accessed with a Python script at the beginning of the observation night. Also, a NEAObs list from the MPC will be produced for specified boundaries for the magnitude (brighter than m_{max}), the motion (below v_{max}) and the sky uncertainty (in-between σ_{min} and σ_{max}). The Priority List and the NEOCP will be filtered by the script with the same boundaries as well. Additionally, the time for which the object's altitude is high enough (larger than φ_{min}) to be observable is calculated. Objects that fall under a given threshold t_{min} for the time are sorted out. What remains is a list of suitable objects for observation.

For the determination of preliminary positions, the current orbital parameters and ephemerides of all objects are downloaded from the MPC Database and the MPC Ephemeris Service, respectively. After converting the parameters into the format of KStars, the objects are included into its database by adding them in the Asteroids file of KStars. Besides, a separate database with the ephemerides is created.

In the next step, an automated Observation Schedule is made for the Ekos Scheduler. For this purpose, an XML file containing all information for the Ekos Scheduler is created, i.e., all individual observation slots must be specified here in the appropriate order. In the beginning there is an initialization procedure used for testing all instruments and then doing an accurate auto-focusing after the camera sensor and the main mirror are cooled down (regular automatic refocusing can be set in the schedule and is also recommended). After that, the actual observation slots are assigned. All objects currently having an altitude below the threshold φ_{min} are removed. This procedure is repeated for several runs with updated altitudes until the nighttime ends. Objects that have already been selected more than N times are also removed. With that, a complete Observation Schedule for NEOs is made and the Pre-Observation procedure is finished.

3.7 Observation pipeline

For the actual Observation Pipeline, which is shown graphically in Figure 5, the schedule and the database are loaded into KStars/Ekos. After manually checking the schedule for correctness, the Ekos Scheduler can be launched. It begins with a Startup Script enabling the power supply of the instruments and checking the weather conditions. If the weather is safe, the dome's shutter will open. After a successful connection and security check of all components, the Startup Script is finished, and the Scheduler will continue with its built-in features (connecting the INDI Server to the devices, unparking dome and mount, *etc.*). Then, the actual Schedule starts with an initialization object (ideally some field near the zenith) to prepare the system (e.g., focusing). After that, Ekos performs each observation job with its selectable modules and settings (e.g., checking starting conditions, slewing mount, slaving the dome, focusing, aligning and guiding). The various modules communicate with the devices *via* the INDI protocol (Figure 2).

One major adaptation we made for the NEO observation is embedded in the optional pre-capture script. In this Python script the current best possible orbital parameters of the NEO are downloaded from the MPC Database and with that the current ephemerides are calculated immediately before each measurement. If there is some correction needed, the mount will adjust its position accordingly. This is done independently of Ekos with the INDI DBus Interface. With that, Ekos slews the mount to the approximate position and the DBus Interface will correct it with the latest information from the MPC. It turns out that such correction is useful not only for fast moving objects, but also slower objects due to the large time spans in an observation night. One of the main problems with the Ekos Scheduler is that it only uses the coordinates when the Schedule is created and does not adjust them according to the ephemeris. However, the procedure shown compensates for this limitation.

After all measurements are finished, the Ekos will park the mount and the dome and disconnect all devices. Then, the observation night ends with a Shutdown Script, which closes the dome and turns off all instruments and their power supply. The measured data are sent to a separate computer and can be evaluated. For educational and logging purposes the entire process combined with images of surveillance cameras is

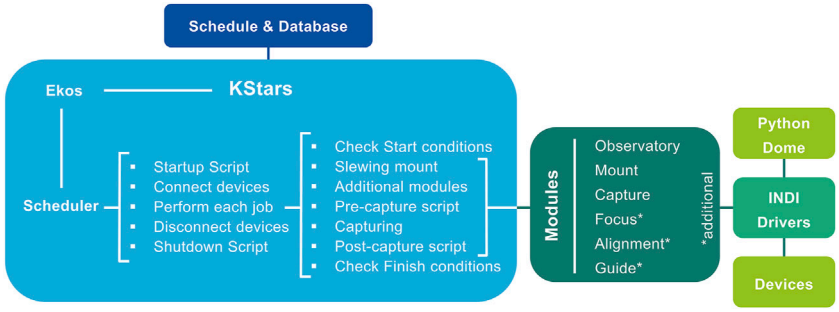


FIGURE 5
Observation Pipeline illustrated with its workflow. It stars with the Schedule and Database created by the Pre-Observation Pipeline in Figure 4 and starts the observation in the Ekos Scheduler. All operations from switching on the components through the capturing process to the end of the observation are executed automatically by the Scheduler functions. Here, customized scripts were embedded in the designated places at Startup, Shutdown and before and after each Capture respectively. When executing the jobs, the scheduler activates the corresponding Ekos modules, which in turn controls the devices *via* the INDI protocol.

broadcasted over a livestream and is stored together with logs for error analyses.

4 Observations and reductions

4.1 MPC submissions

The described robotic system is regularly used for the observation of Minor Planets since Summer 2021. For the initial submission to the MPC in early July 2021, we made the measurements as listed in Supplementary Table S1A with in total 61 measurements of eight different Minor Planets, of which one was a NEO. Of those, 57 measurements were accepted by the MPC, which was enough for the assignment of the MPC code G01 for the observatory.

After that, 20 further reports have been made until mid-March 2022 as listed in Supplementary Tables S1B–U. In total 613 measurements were submitted to the MPC during this period with 570 measurements accepted. Of these, 384 submissions were from NEOs with 341 accepted (measurements may be rejected due to inconsistencies compared to measurements from other observatories).The rejected observations were re-measured, and about half of them were then accepted. The other half had erroneous measurements, so that objects other than the correspondingly indicated NEOs were measured by mistake. These were made on a total of 42 observation nights (27 with NEOs), some of which possessed only short observable phases. So, on average 14.6 measurements of Minor Planets in general and 14.2 of NEOs were made per night. It needs to be considered, that a single object was captured about two to four times per night.

The median of the measured magnitude of each object over the period of the report is calculated. The faintest magnitude is 19.4 mag for the objects 2022 CR3 and 2022 EB3 (Supplementary Tables 1T,U) using a 4×60 s stacked image, but actually the object 2022 EB3 were brighter with 19.0 mag according to the MPC. On the other hand, objects like 2022 DS4 with a measured magnitude of 18.9 mag were fainter with 19.5 mag, which is the LM for moving objects with these settings so far. For longer exposure times and slower moving objects, a higher LM can be achieved. On average, a value for the visual magnitude of (17.68 ± 0.93) mag was measured. Among the NEOs, the average value is (18.13 ± 0.57) mag.

In March 2022, the system also successfully discovered the new asteroid "2022 EX" with the preliminary data from other telescopes around the world in the NEOCP.[22] It was possible to make confirmatory and accurate measurements of the object within a short time after the initial discovery.

4.2 Sky Magnitude

We measured the Sky Brightness for the observation nights with an SQM-LU (Miguel et al., 2017).[23] The SQM is part of "Was het donker?" network of the university of Groningen.[24] The mean Sky Brightness at our site is (19.52 ± 0.48) mag/arcsec^2 with a faintest value of 20.40 mag/arcsec^2 measured.

Using Eq. 2 with a field factor $F = (1.70 \pm 0.30)$ we can calculate the limiting visual star magnitude $m_0 = (5.48 \pm 0.45)$ mag with the measured mean Sky Brightness. This results in a Bortle scale class of 5–6.

22 www.minorplanetcenter.net/mpec/K22/K22E59.html

23 www.unihedron.com/projects/darksky/cd/SQM-LU/SQM-LU_Users_manual.pdf

24 www.washetdonker.nl

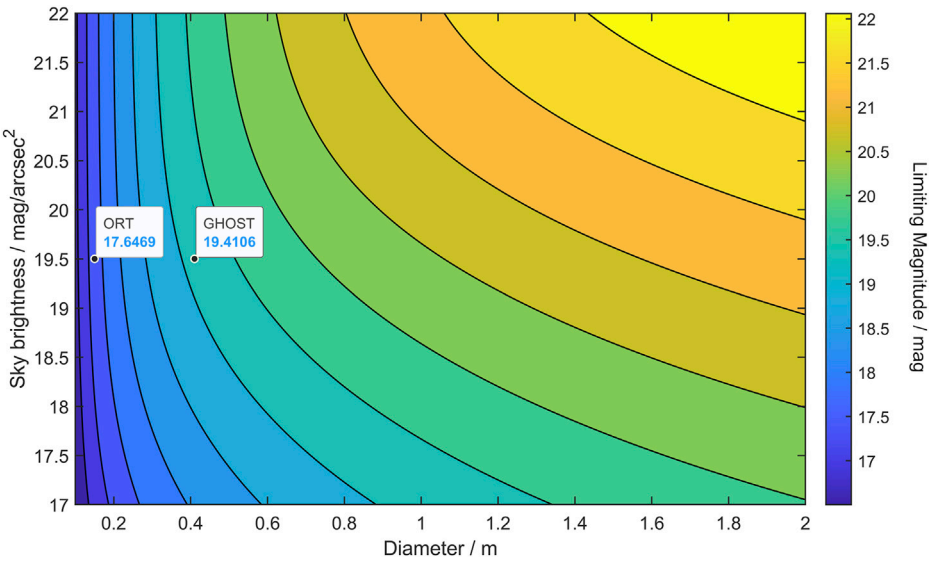


FIGURE 6
Estimation of the Limiting Magnitude for different telescope diameters and Sky Magnitudes calculated by Eq. 12 for an exposure time of 60 s with parameters based on the GHOST telescope (G01). Calculation results for the ORT and GHOST telescopes are marked in the graph.

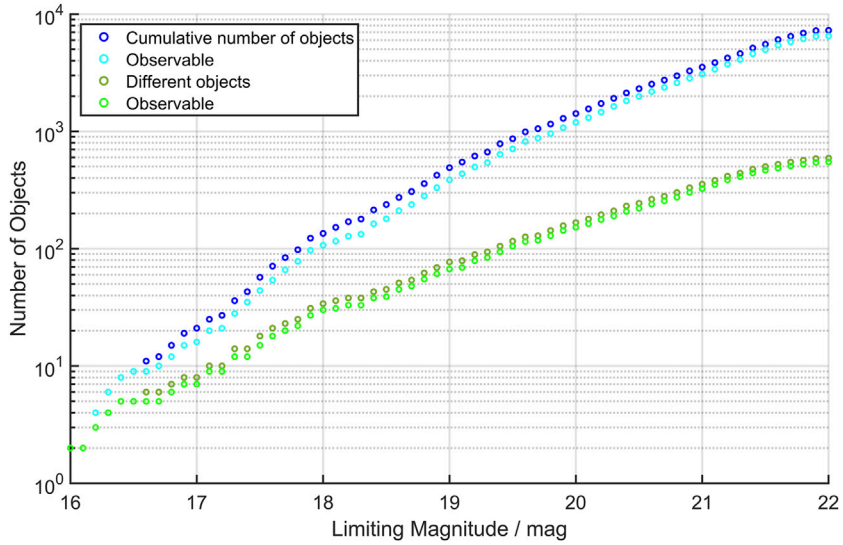


FIGURE 7
Number of objects in the NEOCC Priority List with a magnitude brighter than a particular Limiting Magnitude in the period from 01.12.2021 until 31.01.2022. The cumulative number of all entries (with multiple entries of the same object in different nights) and the observable of them (blue), and the different objects (counted only once for multiple entries on different nights) of these (green) are shown.

4.3 Observational Limiting Magnitude

In order to show, how many possible observations can be reached per night with our proposed methods for different telescope sizes and locations, we use Eqs 11, 12 to calculate an estimation for the LM. For this we assume a minimal *SNR* of 5 for an object to be detectable and take as a basis the parameters of the GHOST telescope and constants from section 3.1 and section 3.4.

For the determination of the *SNR*, we consider only the center pixel of the signal ($n_{\mathrm{px}} = 1$) with a background of $n_{\mathrm{b}} =$

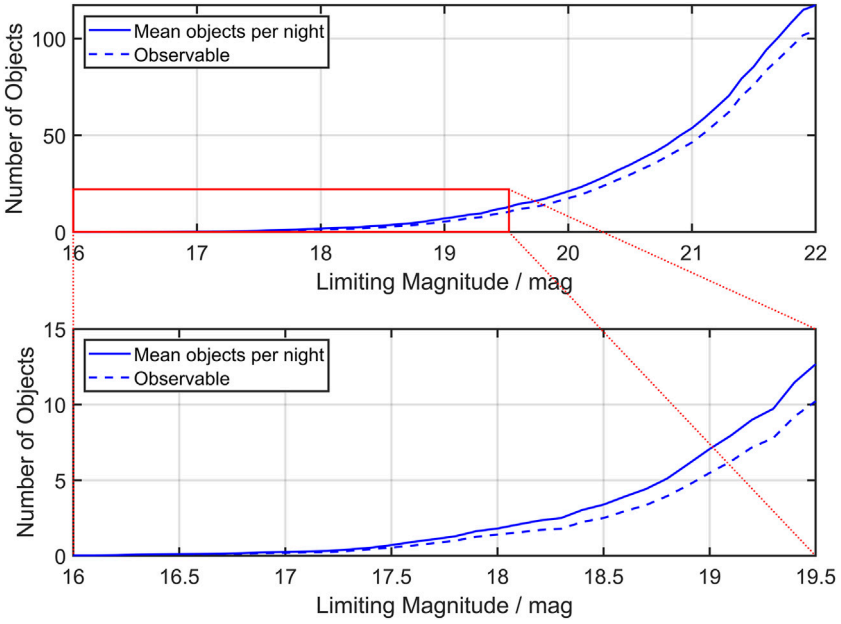


FIGURE 8
Results for the cumulative number of entries from the NEOCC Priority List from Figure 7 averaged per night with a magnitude brighter than a particular Limiting Magnitude and the observable of them (dashed). The range for the Limiting Magnitude from 16 to 22 mag is shown above, the extract of this up to 19.5 mag is shown below.

TABLE 2 Telescopes in Oldenburg (GHOST, ORT), from the 6ROADS Network and ESA's Optical Ground Station (OGS) with their location, MPC Code and the results for the calculated Limiting Magnitude for the given diameter and Sky Magnitudes from Figure 6, the mean observable NEOCC Priority List object from Figure 8 and the mean observation time needed for those objects.

Telescope	Location	MPC code	Diame-ter/m	Sky Mag./mag/arcsec^2	Theo. LM/mag	Priority list objects	Obs. Time needed/h
ORT	Oldenburg,	G01	0.15	19.5*	17.6	0.7	0.18
GHOST	Germany		0.41		19.4	9.2	2.30
Solaris Obs	Cracow, Poland	B63	0.30	20.0	19.0	5.5	1.38
Polonia Obs	San Pedro de Atacama, Chile	W98	0.25	22.0	18.8	4.0	1.00
6ROADS Obs. 1	Wojnowko, Poland	K98	0.40	21.0	19.7	12.7	3.18
Rantiga Obs	Tincana, Italy	D03	0.40	21.0	19.7	12.7	3.18
Springbok Obs	Tivoli, Namibia	L80	0.36	22.0	19.6	11.9	2.98
6ROADS Obs. 2	Nerpio, Spain	Z33	0.40	22.0	19.8	14.0	3.50
ESA OGS	Tenerife, Spain	J04	1.00	21.5	21.4	70.2	17.55

The Sky Magnitude in Oldenburg (*) is measured with an SQM, all other values are obtained from Falchi et al. (2016).

100 pixels (10 × 10 box). Since the signal of the object observed is not focused on this one pixel due to external influences (e.g., seeing, deviations in focusing and collimation), we calculate the portion p of the signal in the center pixel with the mean Full Width at Half Maximum σ_{FWHM} of the signal (assuming Gaussian intensity distribution). In our case a value of σ_{FWHM} = (2.58 ± 0.64)" was determined in Astrometrica. This

includes the average seeing during the measurements. For a gaussian distributed signal σ_{FWHM} is equal to $2\sqrt{2\log 2}$ times the standard deviation σ. By using the properties of the normal distribution, we can calculate p:

$$p = \mathrm{erf}\left(\frac{2s_{px}\sqrt{\log 2}}{\sigma_{FWHM}}\right)^2 = 0.145 \quad (13)$$

with the Gauss error function erf and the pixel scale $s_{px} = 0.68$"/pixel (in binning 2 × 2). Now, for different values of the Sky Magnitude m_{sky} we get the number of photons N_{sky} captured at the camera with the Eqs 6–8. With the telescope's parameters but using different values for its diameter d_{tele}, Eq. 11 with the correction in Eq. 12 lets us estimate the LM m_{lim} of different telescopes. The values of m_{lim} for diameters d_{tele} in range from 0.1 to 2 m and Sky Magnitudes m_{sky} in range from 17.0–22.0 mag/arcsec^2 are shown in Figure 6. Using the diameter of GHOST (d_{tele} = 0.41 m) and our measured mean Sky Brightness (m_{sky} = 19.5 mag/arcsec^2) we will get a theoretical LM of $m_{lim,GHOST}$ = 19.41 mag for a single 60 s exposure. With the values of ORT (d_{tele} = 0.15 m, m_{sky} = 19.5 mag/arcsec^2) this will result in $m_{lim,ORT}$ = 17.65 mag. For longer exposures (α multiples of 60 s), Eq. 11 shows that the LM increases by a value of 2.5 log α. With a typical 4 × 60 s stacked exposure this results in an increase of about 1.5 mag.

4.4 Estimating observational limits

The results for the LM can be used to estimate how many measurements of NEOs can be made per night. Due to the large number of NEOs detected, several of them can usually be observed in one night with a LM of about 17 mag.[25] Therefore, limited only by the total observation time, a large number of measurements can be made for LM fainter than that, but here we focus on observations of objects from the NEOCC Priority List. Such measurements can lead to significant improvement in orbital prediction and are therefore considered a priority (section 2.5).

We analyzed the Priority List for every night in December 2021 and January 2022 for the magnitudes of the objects. First, all entries of the different nights were cumulated and sorted by visual magnitude. Each entry is also checked for observability in terms of its altitude at the geographic coordinates of Oldenburg. Any object that was above 20° altitude for at least 2 h on the night indicated, is considered as observable.

Since objects may appear multiple times on the lists of different night, multiple entries are sorted out separately. Of all occurrences, the lowest value of the magnitude per object is used for the priority list. The cumulative number of all entries and the number of different objects that have a magnitude smaller than a limiting value are plotted in Figure 7. Results from the Priority List over a period of 62 days are used for estimates for individual nights. The resulting mean observable objects per night are plotted in Figure 8.

We can now use these results in connection with the estimations for the LM of different telescope sizes from Figure 6 and compute the mean observable Priority List objects for a given telescope diameter d_{tele} and Sky Magnitude m_{sky}. The results for several observatories (our telescopes, 6ROADS Network and the ESA Optical Ground Station) are listed in Table 2. Assuming an measurement takes 5 min and three measurements per night are required for submission to the MPC, the total observation time required is also given.

5 Discussion

5.1 Robotic observation system

Due to the given customization possibilities in KStars, Ekos and INDI we were able to implement a robotic observation pipeline for Minor Planets and NEOs. Since the pipeline is an extension of the existing INDI framework the generalization of this approach to other observatories should easily be achievable. The pipeline itself is structured straightforward from the input of external data, the object selection to the observation procedure with modular and customizable options (Figures 3–5).

Many observations with our robotic system and the proposed methods were successful so far with increasing number (section 4.1 and Supplementary Table S1). The system delivers reasonable and reliable results and reduces the personnel effort immensely for the observation and planning.

But the pipeline can also be useful for other systems that are not robotic so far or are not specialized for NEOs. On one hand it is useful for larger telescopes, for which many objects requiring follow-up observations are available and thus need better efficiency for the observation time. On the other hand, there are amateur telescopes for which less objects are available, but more effort per capture is needed due to inaccuracies and the difficulty of the handling of the instruments. Therefore, such telescopes could also benefit from the simplicity of use of KStars and Ekos (e.g., its auto-guiding and aligning modules).

There are already successful Robotic Telescopes for NEO follow-up observations that automatically obtain data from the MPC Confirmation Page for the scheduling and update the ephemeris constantly (Holvorcem et al., 2003) or have developed highly automated pipelines (Dotto et al., 2021) and networks (Lister et al., 2015). There are also robotic solutions

25 www.minorplanetcenter.net/whatsup/index

using commercial software (García-Lozano et al., 2016) and also with INDI, KStars and Ekos developed for small telescopes (Gupta et al., 2015), but there is currently no system that combines the usability and expandability of INDI with adaptions needed for NEO observations and the connectivity to the MPC and the NEOCC Priority List to realize an open-source Robotic Observation Pipeline that is applicable for commercially available instruments.

5.2 Limiting Magnitude and Priority List

We can analyze the estimations for the LM of the different telescope sizes and Sky Magnitudes with the results in Figure 6. For small telescopes up to 0.25 m it can be shown that the LM is relatively independent of the Sky Magnitude, which means that is negligible to some extent under which sky conditions the telescope is used. On the other hand, for larger telescopes above 1 m the LM is highly dependent of the sky conditions. This leads to the fact that smaller (and less expensive) telescopes with a good sky condition can make equal or better results as a large telescope with worse sky conditions. So it should be evaluated which telescope is optimal for a given location.

However, also smaller telescopes can obtain similar results if they spend more observation time per object. But since NEOs move relative to the reference stars during measurement, a longer exposure time is not always beneficial. Only a longer line trace of the object on the image is produced. Therefore, it is helpful to stack single measurements with shorter exposure times shifted by the movement of the object. With that, the LM for NEO observations can be increased. Such a stacking tool is implemented in the astrometry software *Astrometrica*.[26]

Figure 7 shows an exponential correlation of the number of objects in the NEOCC Priority List and the LM. This can be expected considering the power law for the absolute magnitude of these objects (Peña et al., 2020) and their distances. Therefore, the large number of faint objects require much observation time, but less telescopes are available for that.

While there is a small difference between the total number of objects and the observable ones, a big difference is noticeable between the cumulative number of entries and the number of different objects. The difference gets larger for higher LMs with a difference of more than one order of magnitude for objects fainter than 20 mag. This means that there are many objects, especially faint ones that remain on the list for many days (on average more than 10 days for objects fainter than 20 mag). This fits with the result that the many faint objects can hardly be covered with the few sufficient telescopes, which is a key result from a study by Seaman et al. (2021). However, a meaningful contribution can already be made with enhanced amateur telescopes with a LM above 18 mag. With these, measurements of several NEOs of the priority list can be performed in one night (Figure 8). Stacking can further increase the LM of smaller telescopes and expand their usage.

5.3 Comparing theoretical and observational results

In our measurements we were able to measure NEOs up to a magnitude of (19.50 ± 0.20) mag for GHOST and (18.00 ± 0.20) mag for ORT under ideal conditions with a stacked 4×60 s measurement. That is slightly above the range of the theoretical calculated values of 19.41 mag and 17.65 mag, respectively. However, the theoretical values refer to a single 60 s exposure, which means that the measured values are brighter than the expectation. Reasons for that might be found in the inaccuracies of some parameters used to estimate the performance (Sky Magnitude m_{sky}, noises N_{bias}, N_{dark} and N_{readout}). Also, the total transmission τ of the telescope and the quantum efficiency QE are based on the manufacturer's data for the camera, mirrors and filters might have decreased over the time. Also, to some extent, Eq. 11 is erroneous due to the assumption of an average wavelength $\bar{\lambda}$ of the incoming photons. Furthermore, we assumed in Eq. 12 an air mass $\chi = 1$, which is true for an object in the zenith, in fact objects have a lower altitude. Together, this can lead to a lower LM of the telescope.

According to the analysis of the NEOCC Priority List in Figure 8 with our estimation for the LM of our telescope in Figure 6 there are on average 9.2 observable NEOs per night in the Priority List for the GHOST telescope assuming a LM of 19.4 mag. In comparison we were able to submit on average 14.2 measurements of NEOs per night (Supplementary Table S1). With an average of three measurements per object per night, this results in 4.7 observed NEOs per night. Indeed, among the 27 NEO observation nights there were at least one third of the nights where the weather conditions were not clear for most of the night. Due to further limiting visual influences, such as the Moon, the faintest magnitude of the Sky Brightness could not be reached on many nights. Considering this, the theoretical expectation agrees with the results.

For other telescopes it needs to be considered that the obtained results are based on the parameters of the GHOST telescope, which are possibly only good assumptions for similarly structured telescopes. This means that the results only give estimates, but can they be calculated more precisely with the general Eqs 11, 12.

5.4 Outlook

A main feature that would be useful for our location is a real-time cloud detection. As we have stated, most nights have only short observation times due to clouds with rapidly

26 www.astrometrica.at

changing weather conditions. A cloud-tracking software that is implemented in the Robotic Telescope system would maximize the possible observation time. Other optimizations can be made in the Scheduler making, such that objects are observed with the ideal observing conditions at the highest altitude possible and thus lower atmospheric extinction.

All in all, the system is useful especially considering an expected increase in the number of follow-up observations needed (Seaman et al., 2021). Currently, there are already not enough observations, and the need will increase by the accelerating rate of discovery caused by more survey telescopes.

6 Conclusion

We developed an optical telescope system with a fully robotic planning, scheduling and observation pipeline especially specialized for NEO observation, which is based on commercially available soft- and hardware components. This allows also other observatories to make use of these methods. Since its completion, it automatically generates measurements that are submitted to the MPC for improvements in trajectory predictions. A decisive improvement in the efficiency of observation time for already existing systems can be achieved by using the pipeline. However, with increasing need for follow-up observations it can also be easily used in the design of new follow-up telescopes.

Data availability statement

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession number(s) can be found in the article/ Supplementary Material.

Author contributions

TH contributed all plots and most text. TH and MG are the instrument scientists and the developer of the software. TP, GD, TO, and JK provided valuable feedback and reviewed the manuscript drafts. TP, JK, and BP are head of the facility. TS-R, MG, RR, and MZ provided external expertise and reviewed the manuscript draft. BP is the supervisor, contributed text, provided valuable feedback, and reviewed the manuscript drafts.

Acknowledgments

We thank ESA's Near-Earth Object Coordination Centre for the provision of parts of the Priority List archive and especially we thank Detlef Koschny for the provided theorical expertise. TS-R acknowledges funding from the NEO-MAPP project (H2020-EU-2-1-6/870377). This work was (partially) funded by the Spanish MICIN/AEI/10.13039/501100011033 and by "ERDF A way of making Europe" by the "European Union" through grant RTI2018-095076-B-C21, and the Institute of Cosmos Sciences University of Barcelona (ICCUB, Unidad de Excelencia "María de Maeztu") through grant CEX2019-000918-M.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fspas.2022.895732/full#supplementary-material

References

Boattini, A., D'Abramo, G., Valsecchi, G. B., and Carusi, A. (2007). A new protocol for the astrometric follow-up of near Earth asteroids. *Earth Moon Planets* 100, 31–41. doi:10.1007/s11038-006-9075-9

Bortle, J. E. (2001). Gauging light pollution: The bortle dark-sky scale. *Sky Telesc. Febr.*

Boslough, M., Brown, P., and Harris, A. (2015). "Updated population and risk assessment for airbursts from near-Earth objects (neos)," in 2015 IEEE Aerospace Conference, 1–12. doi:10.1109/AERO.2015.7119288

Bottke, W. F., Vokrouhlický, D., Rubincam, D. P., and Nesvorný, D. (2006). The yarkovsky and yorp effects: Implications for asteroid dynamics. *Annu. Rev. Earth Planet. Sci.* 34, 157–191. doi:10.1146/annurev.earth.34.031405.125154

Cibin, L., Chiarini, M., Gregori, P., Bernardi, F., Ragazzoni, R., Sessler, G., et al. (2019). "The fly-eye telescope, development and first factory tests results," in Proc. 1st NEO and Debris Detection Conference (ESA Space Safety Programme Office).

Crumey, A. (2014). Human contrast threshold and astronomical visibility. *Mon. Not. R. Astron. Soc.* 442, 2600–2619. doi:10.1093/mnras/stu992

Dotto, E., Banaszkiewicz, M., Banchi, S., Barucci, M. A., Bernardi, F., Birlan, M., et al. (2021). The EU project NEOROCKS — NEO Rapid observation, characterization, and key simulations project. *Eur. Planet. Sci. Congr.* 2021, EPSC2021–389. doi:10.5194/epsc2021-389

Falchi, F., Cinzano, P., Duriscoe, D., Kyba, C. C. M., Elvidge, C. D., Baugh, K., et al. (2016). The new world atlas of artificial night sky brightness. *Sci. Adv.* 2, e1600377. doi:10.1126/sciadv.1600377

García-Lozano, R., Rodes, J. J., Torrejón, J. M., Bernabéu, G., and Berná, J. Á. (2016). The busot observatory: Towards a robotic autonomous telescope. *Rev. Mex. Astron. Astrofísica Ser. Conf.* 48, 16–21.

Gupta, R., Singh, H. P., Kanbur, S. M., Schrimpf, A., and Dersch, C. (2015). U-smart - small aperture robotic telescopes for universities. *Publ. Korean Astronomical Soc.* 30, 683–685. doi:10.5303/PKAS.2015.30.2.683

Hänel, A., Posch, T., Ribas, S. J., Aubé, M., Duriscoe, D., Jechow, A., et al. (2018). Measuring night sky brightness: Methods and challenges. *J. Quantitative Spectrosc. Radiat. Transf.* 205, 278–290. doi:10.1016/j.jqsrt.2017.09.008

Hestroffer, D., Sánchez, P., Staron, L., Bagatin, A. C., Eggl, S., Losert, W., et al. (2019). Small solar system bodies as granular media. *Astron. Astrophys. Rev.* 27, 6. doi:10.1007/s00159-019-0117-5

Hodapp, K. W., Kaiser, N., Aussel, H., Burgett, W., Chambers, K. C., Chun, M., et al. (2004). Design of the pan-starrs telescopes. *Astron. Nachr.* 325, 636–642. doi:10.1002/asna.200410300

Holvorcem, P. R., Schwartz, M., Juels, C. W., Breganhola, M., Camargo, J., and Teixeira, R. (2003). Search by orcid, et alAstrometry of near-Earth asteroids using remotely-operated robotic telescopes. *Astronomy Lat. Am.* 1, 91–100.

Huebner, W. F., Johnson, L. N., Boice, D. C., Bradley, P., Chocron, S., Ghosh, A., et al. (2009). A comprehensive program for countermeasures against potentially hazardous objects (phos). *Sol. Syst. Res.* 43, 334–342. doi:10.1134/S003809460904008X

IAU General Assembly (2006). *Resolution b5: Definition of a planet in the solar system*. Busan, South Korea: IAU General Assembly.

Jurado Vargas, M., Merchán Benítez, P., Sánchez Bajo, F., and Astillero Vivas, A. (2002). Measurements of atmospheric extinction at a ground level observatory. *Astrophys. Space Sci.* 279, 261–269. doi:10.1023/A:1015184127925

Koschny, D., Drolshagen, E., Drolshagen, S., Kretschmer, J., Ott, T., Drolshagen, G., et al. (2017). Flux densities of meteoroids derived from optical double-station observations. *Planet. Space Sci.* 143, 230–237. doi:10.1016/j.pss.2016.12.007

Koschny, D., and Igenbergs, E. (2020). *Near-earth objects: A threat for earth? – or: Neos for engineers and physicists: Script for the course: Near-earth objects for engineers and physicists*.

Larson, S., Brownlee, J., Hergenrother, C., and Spahr, T. (1998). The catalina sky survey for neos. *Bull. Am. Astronomical Soc.* 30, 1037.

Lister, T. A., Greenstreet, S., Gomez, E., Christensen, E., and Larson, S. (2015). The lcogt neo follow-up network. *Proc. Int. Astron. Union* 10, 321–323. doi:10.1017/S1743921315006778

Mainzer, A., Abell, P., Bannister, M. T., Barbee, B., Barnes, J., Bell, J. F., III, et al. (2021). The future of planetary defense in the era of advanced surveys. *Bull. AAS* 53. doi:10.3847/25c2cfeb.ba7af878

Mamajek, E. E., Prsa, A., Torres, G., Harmanec, P., Asplund, M., Bennett, P. D., et al. (2015). "Iau 2015 resolution b3 on recommended nominal conversion constants for selected solar and planetary properties," in XXIX International Astronomical Union General Assembly (IAU).

Meftah, M., Damé, L., Bolsée, D., Hauchecorne, A., Pereira, N., Sluse, D., et al. (2018). Solar-iss: A new reference spectrum based on solar/solspec observations. *Astron. Astrophys.* 611, A1. doi:10.1051/0004-6361/201731316

Merline, W. J., and Howell, S. B. (1995). A realistic model for point-sources imaged on array detectors: The model and initial results. *Exp. Astron. (Dordr).* 6, 163–210. doi:10.1007/BF00421131

Micheli, M., Koschny, D., Drolshagen, G., Hainaut, O., and Bernardi, F. (2014). An esa neocc effort to eliminate high palermo scale virtual impactors. *Earth Moon Planets* 113, 1–13. doi:10.1007/s11038-014-9441-y

Micheli, M., Koschny, D., Drolshagen, G., Perozzi, E., and Borgia, B. (2015). Neo follow-up, recovery and precovery campaigns at the esa neo coordination centre. *Proc. Int. Astron. Union* 10, 274–281. doi:10.1017/S1743921315009175

Miguel, A. S. d., Zamorano, M. A. J., Kocifaj, M., Roby, J., and Tapia, C. (2017). Sky quality meter measurements in a colour changing world. *Mon. Not. R. Astron. Soc.* 467, 2966–2979. doi:10.1093/mnras/stx145

Milani, A. (1999). The asteroid identification problem. *Icarus* 137, 269–292. doi:10.1006/icar.1999.6045

Peña, J., Fuentes, C., Förster, F., Martínez-Palomera, J., Cabrera-Vives, G., Maureira, J. C., et al. (2020). Asteroids' size distribution and colors from hits. *Astron. J.* 159, 148. doi:10.3847/1538-3881/ab7338

Perna, D., Barucci, M. A., and Fulchignoni, M. (2013). The near-Earth objects and their potential threat to our planet. *Astron. Astrophys. Rev.* 21, 65. doi:10.1007/s00159-013-0065-4

Raab, H. (2002). "Detecting and measuring faint point sources with a ccd," in Meeting on Asteroids and Comets in Europe 2002.

Rumpf, C., Lewis, H. G., and Atkinson, P. M. (2016). The global impact distribution of near-Earth objects. *Icarus* 265, 209–217. doi:10.1016/j.icarus.2015.10.026

Seaman, R., Bauer, J., Brucker, M., Christensen, E., Grav, T., Jones, L., et al. (2021). NEO surveys and ground-based follow-up. *Bull. AAS* 53. doi:10.3847/25c2cfeb.9eb9da4e

Shporer, A., Brown, T., Lister, T., Street, R., Tsapras, Y., Bianco, F., et al. (2010). The LCOGT network. *Proc. Int. Astron. Union* 6, 553–555. doi:10.1017/s1743921311021193

Vereš, P., Payne, M. J., Holman, M. J., Farnocchia, D., Williams, G. V., Keys, S., et al. (2018). Unconfirmed near-Earth objects. *Astron. J.* 156, 5. doi:10.3847/1538-3881/aac37d

frontiers | Frontiers in Astronomy and Space Sciences

TYPE Original Research
PUBLISHED 27 October 2022
DOI 10.3389/fspas.2022.929573

OPEN ACCESS

EDITED BY
Chenzhou Cui,
National Astronomical Observatories (CAS), China

REVIEWED BY
Niu Liu,
Nanjing University, China
J. Allyn Smith,
Austin Peay State University, United States

*CORRESPONDENCE
Michael J. Mazur,
mmazur5@uwo.ca

SPECIALTY SECTION
This article was submitted to Astronomical Instrumentation, a section of the journal Frontiers in Astronomy and Space Sciences

RECEIVED 27 April 2022
ACCEPTED 10 October 2022
PUBLISHED 27 October 2022

CITATION
Mazur MJ, Metchev S, Brown RA, Gupta R, Bloch R, Mills T and Pass E (2022), The colibri telescope array for KBO detection through serendipitous stellar occultations: A technical description.
Front. Astron. Space Sci. 9:929573.
doi: 10.3389/fspas.2022.929573

The colibri telescope array for KBO detection through serendipitous stellar occultations: A technical description

Michael J. Mazur[1]*, Stanimir Metchev[1,2], Rachel A. Brown[1,2], Ridhee Gupta[3], Richard Bloch[4], Tristan Mills[1] and Emily Pass[5]

[1]Department of Physics and Astronomy, Western University, London, ON, Canada, [2]Institute for Earth and Space Exploration, Western University, London, ON, Canada, [3]Department of Physics, University of Waterloo, Waterloo, ON, Canada, [4]Department of Physics and Astronomy, York University, Toronto, ON, Canada, [5]Center for Astrophysics, Harvard & Smithsonian, Cambridge, MA, United States

We present the technical design, construction and testing of the Colibri telescope array at Elginfield Observatory near London, Ontario, Canada. Three 50-cm telescopes are arranged in a triangular array and are separated by 110–160 m. During operation, they will monitor field stars at the intersections of the ecliptic and galactic plane for serendipitous stellar occultations (SSOs) by trans-Neptunian objects (TNOs). At a frame rate of 40 frames per second (fps), Fresnel diffraction in the occultation light curve can be resolved and, with coincident detections, be used to estimate basic properties of the occulting object. Using off-the-shelf components, the Colibri system streams imagery to disk at a rate of 1.5 GB/s for next-day processing by a custom occultation detection pipeline. The imaging system has been tested and is found to perform well, given the moderate site conditions. Limiting magnitudes at 40 fps are found to be about 12.1 (temporal SNR = 5, visible light Gaia *G* band) with time-series standard deviations ranging from about 0.035 mag to >0.2 mag. SNR is observed to decrease linearly with magnitude for stars fainter than about $G = 9.5$ mag. Brighter than this limit, SNR is constant, suggesting that atmospheric scintillation is the dominant noise source. Astrometric solutions show errors typically less than ±0.3 pixels (0.8 arc seconds) without a need for high-order corrections.

KEYWORDS

robotic telescopes, kuiper belt objects, occultation, photometry, fresnel diffraction

1 Introduction

1.1 The undiscovered population of kilometre-sized outer solar system objects

Beyond Neptune orbits a population of up to 10^{11} objects with sizes of a kilometre or larger (Roques and Moncuquet, 2000), known as trans-Neptunian objects (TNOs). It can be divided into three main subgroups: Kuiper Belt objects (KBOs), scattered disk objects, and the detached objects such as the Sednoids and the distant Oort Cloud objects (Delsanti and Jewitt, 2006). Extending from about 30 AU to about 50 AU from the sun, the Kuiper Belt contains objects with predominantly low inclinations that are dynamically stable compared to the broader group of TNOs. The KBOs can be subdivided into two groups known as the dynamically cold and hot populations. The cold population typically have low inclination (<5°) orbits and are believed to represent a dynamically primordial population (Levison and Stern, 2001). The cold population is also expected to have the highest sky surface density, since it most closely follows the solar system ecliptic plane.

Compared to asteroids, for which the size-frequency distribution is reasonably well understood, less is known about this population of distant solar system objects. Because of their distance from Earth, smaller KBOs can be difficult or impossible to observe directly. Of the more than one thousand KBOs discovered by direct imaging, only a small fraction (3%) are less than 25 km in size and none are less than 7 km. As a result, the size-frequency distribution of KBOs and, more generally TNOs, is poorly defined for smaller objects.

1.2 Indirect detection through serendipitous stellar occultations

Using stellar occultations as a method for the indirect detection of "invisible" bodies in the Solar System was suggested by Bailey (1976). Using a 1 m telescope with a vidicon imager and a visual (V-band: ≈ 505–595 nm) limiting magnitude of 16, Bailey suggested that a 1000-star field would have an occultation event every 11 h. His analysis, however, is optimistic as it does not consider the effects of line-of-sight Fresnel diffraction nor fully explore the temporal limit imposed by a 10 Hz sample rate.

A theoretical description of Fresnel diffraction during stellar occultations by small bodies was discussed by Roques et al. (1987). Fresnel diffraction is used to describe near-field diffraction and, given the correct geometries and source/object sizes, can be used to describe the observed light curves from KBO occultation events. Roques et al. (1987) examined both the theory and implementation of diffraction models and forms the basis of subsequent observational studies in the field. Roques and Moncuquet (2000) further explored the possibility of detecting small bodies in the Solar System. Their work focused on sub-kilometre KBOs and predicted the possibility of a few to several tens of occultation events per night with 2-m to 8-m class telescopes at a visual limiting magnitude of 10 mag $\leq V \leq$ 12 mag. With today's instrumentation, however, similar limiting magnitudes are achievable with smaller telescopes operating at higher frame rates.

In order to sufficiently resolve a KBO occultation light curve, we need to image at relatively high frame rates. For a 1 km KBO at 40 AU observed at solar opposition, its speed relative to the observer is about 25 km/s. If this object occults a star with a 1 km projected diameter, the occultation will have a duration $\Delta T \approx 80$ ms if we consider the event purely geometrically. However, the diffractive broadening of the geometric shadow prolongs the event by a factor of 2–3. The effect is more pronounced for smaller stellar disks and better photometric precision (Roques et al., 1987; Roques and Moncuquet, 2000).

Taking into account relative velocities, Fresnel geometry, and telescope sensitivity, Bickerton et al. (2009) find that the optimal sampling rate for detecting serendipitous stellar occultations by KBOs at visible wavelengths is 40 fps, with observations toward solar opposition. These are the parameters that we use to set the technical requirements for the Colibri telescope array. These parameters are preliminary. Until reliable KBO occultation detections become routine, the optimal trade-off between telescope size, sampling rate, observing wavelength, and observing geometry has yet to be empirically validated.

Indeed, the Bickerton et al. (2009) study details a much longer set of assumptions to project KBO occultation detection rates. We defer a discussion of the expected event rate specific to the Colibri experiment to a future publication (Metchev et al., 2022; in preparation). In the meantime, we note that some recent and on-going experiments (**Section 1.3**) validate our choice of telescope system and operating mode.

1.3 Other previous or planned experiments

Several prior experiments have reported serendipitous stellar occultations by KBOs. These have often been based on sub-optimal data sets, in some cases acquired for a different science goal. Chang et al. (2006) analyse 90 h of archival x-ray monitoring observations of a single source, Cygnus X-1, with the Rossi X-ray Timing Explorer (RXTE) satellite at 500 fps. The 58 candidate occultations reported in their data are by far the largest in a single data set. However, subsequent re-analyses of the data and their statistical interpretation by Jones et al. (2006) and Bickerton et al. (2008) have put these detections into doubt. Schlichting et al. (2009, 2012) report two different candidate occultations from visible-light guiding operations at 40 fps from

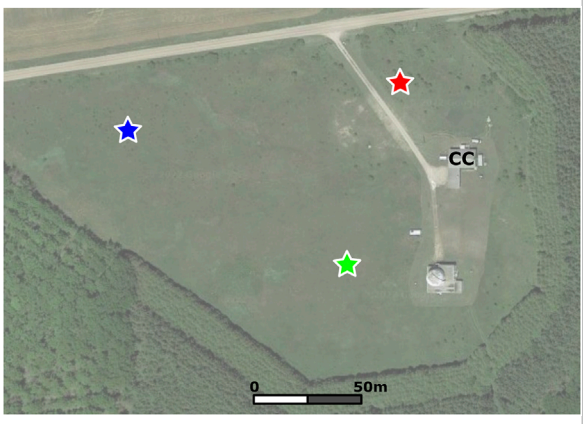


FIGURE 1
A satellite image showing the location of the three Colibri telescopes (stars) and the control center (CC) located in the basement of a house on the site. Maps Data: Google. Imagery © 2022 First Base Solutions, Maxar Technologies.

over 20 years of observatinos with the Hubble Space Telescope. These events do bear the hallmarks of the expected Fresnel diffraction pattern of stellar occultations by kilometre-sized KBOs (e.g., **Figure 1** of Schlichting et al., 2009). Detections of similar events with other facilities would confirm them as representative of this phenomenon.

Early observations designed specifically for the detection of serendipitous stellar occultations have also yielded some candidate detections and mixed results. Roques et al. (2006) report three candidate events in 11 h of dual-band visible wavelength monitoring of two stars at 45 fps with the 4.2 m William Herschel Telescope. Bickerton et al. (2008) discuss that while the rate of events in this study is in line with expectations, their statistical significance is low. More recently, Arimatsu et al. (2019) report a single candidate event using a pair of 28 cm amateur optical telescopes, in a 60 h observation at 15.4 fps in the course of the Organized Autotelescopes for Serendipitous Event Survey (OASES; Arimatsu et al., 2017). The sub-optimal (<40 fps) cadence of the observations and the low (<10) SNR of the four individual measurements that constitute the candidate event leave enough room for it to be a false positive. Nonetheless, the OASES setup and its use of commercial hardware and a rapid-imaging complementary metal–oxide–semiconductor (CMOS) camera are promising for designing large-scale serendipitous stellar occultation surveys.

Most significantly, the Taiwanese-American Occultation Survey (TAOS; Lehner et al., 2009) was specifically designed to identify ~1 km-diameter objects beyond the orbit of Neptune, and to measure the size distribution of KBOs with diameters between 0.5 and 30 km. Seven years of visible-light monitoring with initially three, and then four 50 cm telescopes with TAOS yielded no occultation detections (Zhang et al., 2013). This was attributed to a lower-than-expected event rate, and also the relatively slow (5 fps) sampling of the cameras. A follow-up experiment, TAOS II (Lehner et al., 2012), is in the final stages of development, and will use three 1.3 m telescopes imaging at 20 fps. Much like OASES, and the herein described Colibri Telescope Array, TAOS II will use a CMOS-type visible-light camera.

2 Colibri hardware

Since stellar occultations by KBOs tend to be short-lived, they are difficult to observe. Any experiment to identify these transient events needs to not only rapidly (at 40 fps) image background stars but should also provide a mechanism for confirmation. The three telescopes of the Colibri array are set up in a triangular pattern with spacings of between 110 m and 160 m from each other (**Figure 1**). This arrangement allows us to rule out the possibility of atmospheric twinkling for coincident events but is not sufficient for size/distance determination by differential transit timing. A full description of the technical specifications of the Colibri array is given in **Table 1**.

2.1 Telescopes and domes

The light-weight carbon fiber optical tube assemblies support 50 cm f/3 cellular mirrors cast from Schott Boro 33 glass. The telescope manufacturer is Hercules Telescopes, which both built the OTAs and cast the mirrors. The mirrors were then ground and figured by an external contractor. ASA Wynne correctors correct field aberrations for imaging at prime focus. FLI Kepler KL4040 cameras are attached to a focus assembly that is controlled by a

TABLE 1 Description of the three-telescope Colibri array.

	Specification
Location 1 (Lat., Long.)	43.193365°N, 81.316090°W
Location 2 (Lat., Long.)	43.192415°N, 81.316467°W
Location 3 (Lat., Long.)	43.193108°N, 81.318033°W
Telescope baselines	110–160 m
Telescopes	Hercules 50 cm f/3 w/Wynne Correctors
Cameras	FLI Kepler KL4040
Sensors	GSense 4040
Pixels	4096 × 4096, 9 μm pitch
Field-of-View	1.43°x1.43°
Pixel Scale (2 × 2 binning)	2.52″ per binned pixel
Digitization	low/high gain at 12-bit
Readout Noise	3.7 e^-
Dark Current at -20 °C	<0.15 e^-/pixel/s
Typical Gain	low = 19 e^-/ADU, high = 0.8 e^-/ADU
Quantum Efficiency	>70% (460 nm—680 nm)
Framerate at 2 × 2 binning	>40 fps
Mag. Limit (SNR = 3) in 25 ms	G = 12.5
Mag. Limit (SNR = 5) in 25 ms	G = 12.1

FIGURE 2
The Hercules 50-cm telescope is mounted on an AP1600 GTO mount inside of a 12′ automated dome.

Seletek Platypus controller. The controller is connected to our local VLAN where it is accessible to the control computers. Absolute timing is achieved by a Garmin GPS attached to each camera.

The telescopes are attached to Astro-Physics AP1600 GTO equatorial mounts that sit on custom-built steel piers (**Figure 2**). The steel piers are, in turn, attached to concrete piers extending to about 6 m below the dome floor. Protection from the elements is provided by ExploraDome EDII 12′ plastic domes that have a number of modifications to increase their reliability and allow them to be operated robotically, under the icy and windy conditions common at the site. The original dome control electronics proved to be unreliable and so were replaced with MaxDome II controllers. The dome controllers are connected to USB-over-IP appliances for dome control over much greater distances than would otherwise be possible.

Environmental conditions are monitored on-site by a number of different systems. A central Davis Pro weather station provides an overview of the weather while each dome will be equipped with its own networked environmental station that measures inside/outside temperatures and humidities as well as ambient light levels. Additionally an Arduino-based IR cloud monitor and a dedicated camera continuously aimed at the star Polaris give feedback on seeing conditions. Python scripts talk to each of the devices, parsing and collating the data into a format that can be read by the telescope control software.

2.2 Imaging cameras

Traditional charge-coupled device (CCD) cameras—available in large formats at reasonable costs—are still a common option for many astronomical imaging applications. Although most off-the-shelf solutions use high-speed USB interfaces, their frame rates are typically no faster than a few frames per second. As this is still too low for resolving most KBO occultations, commercial CCD cameras do not meet our requirements.

There are other cameras, using electron-multiplying CCDs (EMCCDs), that often have higher frame rates and better noise characteristics than regular CCD cameras. By using an electron-multiplying stage prior to their output amplifier, EMCCD cameras can achieve gains of more than 1,000 while maintaining a read noise on the order of a few electrons. Their cost, however, tends to be high and their sensor sizes small. A high-sensitivity EMCCD camera borrowed from the Western Meteor Physics Group was tested, but its high cost, moderate frame rate (17 fps), and small chip dimensions (13 mm × 13 mm) effectively ruled it out as a viable option. The testing did, however, allow the Colibri processing pipeline to be validated on real data and reinforce the need for higher frame rates and a large field of view (Pass et al., 2018).

Over the past few years, there has been a shift within the imaging community from CCD sensors to CMOS sensors. While CMOS-based cameras have been found in cameras and mobile phones for years, scientific CMOS (sCMOS) cameras are relatively new to the astronomical market. Their architecture differs from CCDs as, instead of an output register through which pixel data passes serially, they have an output amplifier for every column so that data from individual pixels can be read out in parallel. This has the advantage of significant speed increases over CCDs, but typically comes with the tradeoff of having a rolling shutter instead of a global shutter. Although global shuttering has the desired characteristic of instantaneous capture of an entire frame, a rolling shutter is only a liability when objects are moving across the field of view. Compensations for timing offsets between rows can easily be made during processing if deemed necessary. With a 25 ms exposure time and 2048 (2 × 2 binning) rows being read out, the row-to-row timing offset is 1.2 μs. Similar to the accuracy of our time stamp, it is not significant as long as the pointing differences between the three telescopes is small. Considering all of this, sCMOS-based cameras were the only technology that could meet the speed, field-of-view, and cost requirements of the Colibri project.

Although sCMOS-based cameras were the preferred choice, no suitable cameras were commercially available at the time of project conception. The cameras that were on the market were either equipped with small chips or delivered frame rates much lower than the required 40 fps. At the time, FLI was developing their Kepler series cameras which could be equipped with a large-format sCMOS sensor and, most importantly, would be able to image at frame rates >30 fps over a fiber connection. Because of this, FLI was chosen to supply three KL4040 cameras for the project.

The cameras have a number of possible modes of operation. They can be run in image capture mode where data is buffered and written to disk in several different formats, or they can stream straight from camera to disk. When running in the first mode, the cameras can be connected to the PC *via* either a USB3.0 or a fiber connection. When running over USB, speed is limited and we find a maximum frame rate of about 15 fps when 2 × 2 binned. A fiber connection allows for much higher frame rates (40 + fps), however the raw-to-fits format conversion when writing to disk slows things considerably after the camera's buffer has been filled. To allow for the required continuous high-speed imaging, data from the cameras is written directly to a RAID from a raw data stream with no format conversion. The raw format of the stream is one where each image container holds a header and both the low- and high-gain exposure data. The header takes up the first 246 bytes and contains necessary information about the exposure such as integration time and GPS time. After the header, the 12-bit high- and low-gain image data is interleaved by rows, giving a resultant raw image size of 12.6 MB. The two images can then be split and combined to create a single high dynamic range (HDR) 16-bit image. As the creation of an HDR image adds significant processing time, we currently extract only the high-gain image for processing. Because the use of the high-gain image gives roughly 2 magnitudes greater depth compared to the low-gain image without saturating stars in our target fields, trading off dynamic range and a slight increase in noise for processing speed has only a minor effect on our ability to detect SSOs with our pipeline.

2.3 Network and control computers

Two fiber networks connect the imaging equipment in the domes with control computers in a house located 270 m (along the fiber) from the furthest dome. The first network uses 1000BASE-SX transceivers over single-mode fiber and is used for communications with the telescope, focuser, guide and security cameras, and dome controller. The second network is used exclusively by the imaging camera. To keep up with the camera's high frame rate, variable bit-rate 40GBASE-CSR QSFP + transceivers run at speeds of 8 Gb per second on each of four channels over 8 strands of OM3 multi-mode fiber. Speed tests have shown that frame rates up to 50 fps with 2 × 2 binned images are achievable, but are at the upper limit of our four-disk RAIDs.

Each dome has its own dedicated control computer to manage equipment operations, scheduling, and imaging tasks. The PCs are running Windows 10 with observatory control managed by ACP software. ACP allows us to connect to any device with ASCOM drivers which makes integration of off-the-shelf equipment relatively straight-forward. Technically, the PCs are mid-range machines with 3.2 GHz i7-8,700 processors and 32 GB of RAM. To allow for image streaming to disk at rates of up to 4 Gb per second, each computer has a striped RAID of 4 10 TB Seagate Exos hard disks. This gives a peak performance of about 5 Gb per second with a total storage capacity of 36 TB. At 40 fps this gives enough space for about 20 h—roughly 2 to 3 nights—of data.

3 Colibri data handling

3.1 Data acquisition

Since the streaming mode of camera operation is not available through the FLI ASCOM driver, we have written our own camera control code in C#. This lightweight application runs from the command line and gives the user access to the main camera settings while running in streaming mode. Command line usability also means that the program is accessible from within ACP's scripting environment.

The lack of ASCOM camera support for our operations also means that we need an alternative to ACP's scheduler. We accomplished this by writing JavaScript code to handle rudimentary scheduling that could be run from ACP. After initiation, the script checks for the presence of weather data. If present, the script continues. Otherwise, the user is given the opportunity to either quit or continue without the safety of the weather station data. The next step creates the data directories for that night's data and collects a set of 50 bias images. The script then calculates sunset and sunrise times, as well as the position of the moon relative to a pre-determined list of stellar fields of observation. Details on the assembly of this pre-determined set of stellar fields are deferred to a future publication (Metchev et al., 2022; in preparation).

We iterate through each field in the list and estimate the number of visible stars according to airmass in 6 min steps from sunset to sunrise. The fields are then filtered based on elevation (must be >10°) and moon angle (must be >15°), with ones that do not satisfy these criteria removed from the list of fields for the night. The remaining fields are then ranked, according to the estimated number of stars visible, for each 6 min window. Once this has been done at the beginning of a night, a sunset check is made. If the sun is less than 12° below the horizon, the system pauses until the sun has passed this limit.

Once it has been determined that it is dark enough to open, a weather-check is made and, if it is safe, the dome is opened, homed, and synced to the telescope. After another set of biases are collected, the telescope then slews to the first field and begins capturing data in subsets of 2,400 images. Once another field surpasses the current field, the telescope is slewed to the new location and imaging begins once again. This is repeated until either sunrise or a weather alert shuts the system down.

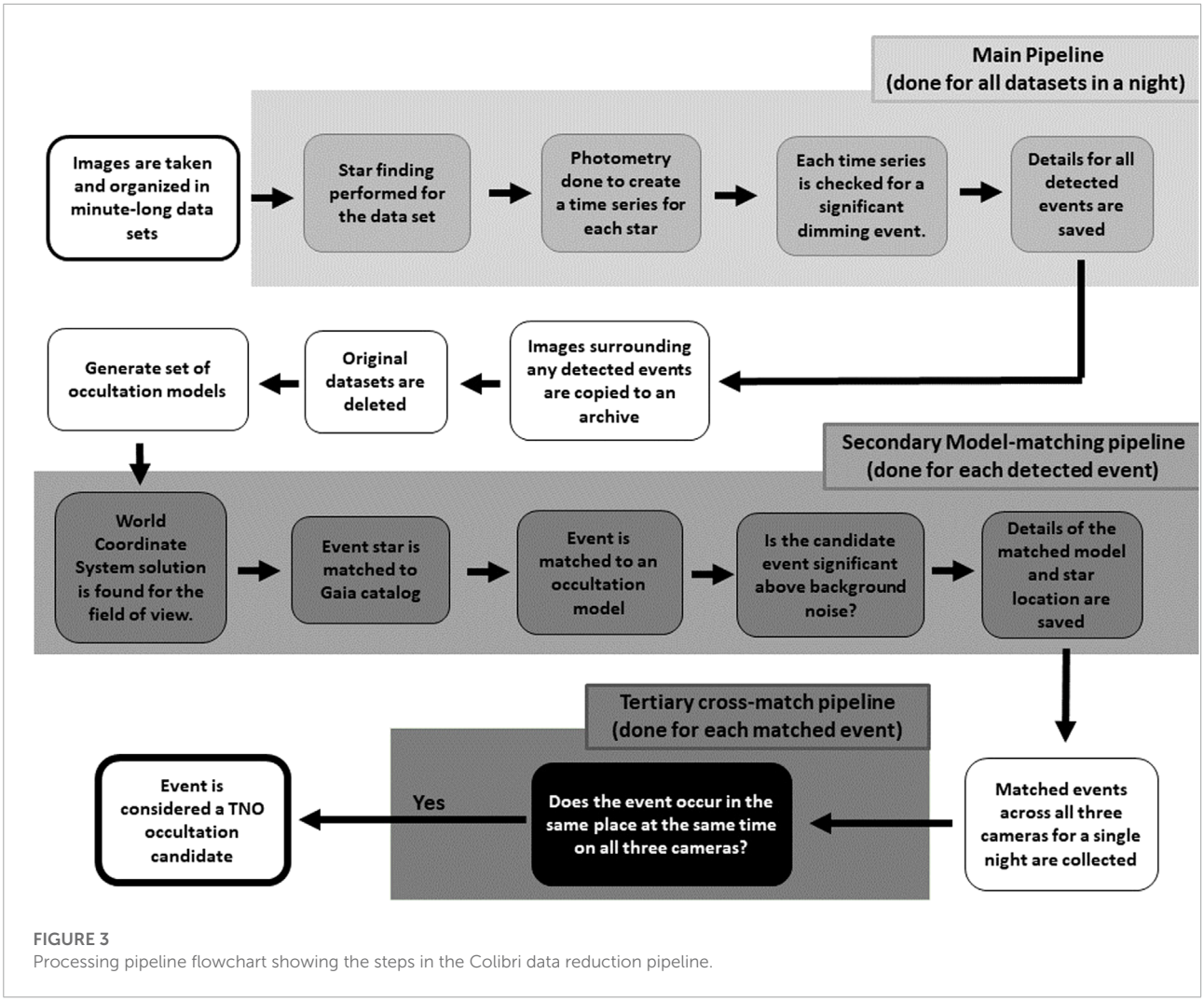


FIGURE 3
Processing pipeline flowchart showing the steps in the Colibri data reduction pipeline.

3.2 Preliminary data processing

After being written to disk, the images can be viewed individually with either FLI's Pilot software or with custom python scripts for viewing and conversion to FITS format. Batch conversion to FITS format averages about 20 ms per image. This can be done after a night of observations, but effectively takes nearly as long as the observations themselves. It is therefore an important limitation as we aim to process data during daylight hours. Our data storage allows up to a few consecutive clear nights of observations, which is generally sufficient given the ~30% fraction of clear nights at Elginfield Observatory.

As with CCD imagers, sCMOS cameras should be corrected for bias, flat-field, and dark current prior to quantitative/absolute work being done. Although the noise specifications are quite low, the sensors in our cameras have a considerable amount of bias structure that is especially visible during rapid imaging. As such, it must be removed by subtraction of a median combined set of bias frames.

With the short (25 ms) exposures that are typical for Colibri operation, the low dark current (0.15 e^-/pixel/s) of the sensor means that we can ignore separate dark current removal. Even then, because of the way that we collect the biases (25 ms, shutter closed), dark current is effectively removed during bias subtraction. Collection of a traditional bias with a 0 ms exposure is not possible currently due to limitations with the camera hardware.

The fast, wide-field nature of our optical system lends itself to vignetting on the image plane. This is visible as darkening outwards from the centre of the field and is easily corrected with a proper flat-field image. For stellar extraction and the relative photometry that we perform in our workflow, however, flat-fielding is not strictly necessary and is not applied during the first stage of processing due to time constraints.

3.3 Occultation detection pipeline

Although the detection pipeline (**Figure 3**) follows the general description given by Pass et al. (2018), changes to the equipment specifications have necessitated modifications to the pipeline.

After the initial preparation of the data has been completed, stellar sources are extracted by running the Source Extraction and Photometry (SEP) module for Python (Bertin and Arnouts, 1996; Barbary, 2016) on a median combination of the first nine images from each 2,400 image subset collected over the night. The output from the SEP *extract* function gives the coordinates for each star with a flux at least 4 σ above the background at the start of each subset. After this, the average stellar drift within each subset is calculated. Every image in a subset is then interrogated using the *sum_circle* function of SEP to produce a time series of fluxes at each drift-corrected x,y location. The light curves are then passed to the first-stage dip detection where stars are filtered based on their SNR (must be greater than 5) and whether they drift out of the field. We then look for geometric dips greater than 40% of the normalized light curve. A dip much less than 40% is considered a common event (due to noise) and would represent a false detection. The geometric dip detections are automatically saved as candidates. In the case of no geometric dip, the light curve is passed to the KBO dip detection function which convolves a Ricker wavelet designed to match the characteristic width of a KBO occultation event with each time series. The size of the wavelet is chosen based on the expected occultation duration given the pointing of the telescope. This means that, with expected durations of between 160 ms and 500 ms, Ricker wavelets of between 4 and 12 frames length are most appropriate. We then look for the most significant dimming events in each time series. If the dips of these events are greater than 3.75σ of the mean of the convolved time series, the event is passed to the next stage of processing where the events are compared to a set of pre-calculated kernels that model diffraction patterns for a range of different physical/dynamical properties. Finally, successful matches from this step are checked for correlation in time with the other two telescopes of the array and saved offsite.

4 System performance

4.1 Optical

When pairing a fast (f/3) mirror design with an off-the-shelf (ASA Wynne) field corrector, the final results can be difficult to predict. The prescription for the ASA corrector is unknown, but the manufacturer claims that it has been designed with parabolic primaries from f/3–f/5 in mind. Extrapolating from their sales literature, we would expect spots ranging from about 4 to 12 μm (400–700 nm) on the optical axis to 12+ μm in the corners of our sensor. Although much better than an uncorrected system, our 9 μm pixels mean that we should be able to observe the difference in spot sizes from centre to sensor corners (**Figure 4**). Raytracing the as-built mirror specs with a generic Wynne corrector prescription, however, suggests that the design has excellent optical performance over the entire field for our observing conditions and our 2.5″ (2 × 2 binned) pixels. With seeing often more than several arcseconds at the Elginfield site, we can tolerate slightly defocused images while still maintaining critically sampled PSFs across our 2 × 2 binned images.

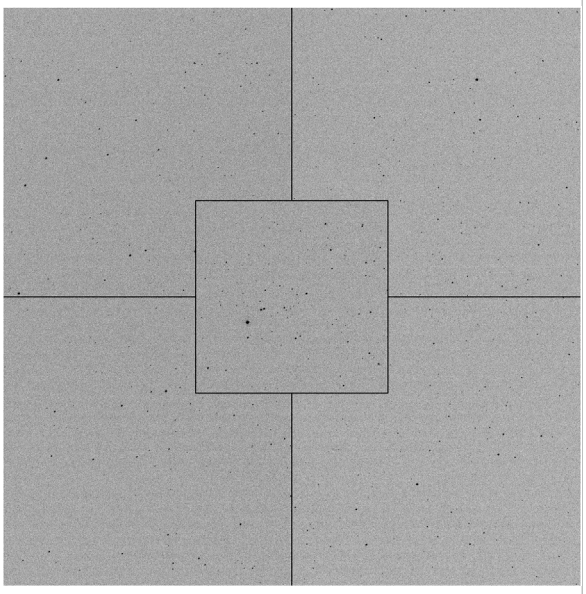


FIGURE 4
The four corners of this 10 s, 2 × 2 binned image (2.51″ per binned pixel) show spot sizes visually similar to the central portion of the image. The outer fields are 30′ square while the central square is 20′ square.

Although the coma-corrected field-of-view is claimed to be 60 mm, there is vignetting at the field edges. This is visible in flat field imagery and amounts to a loss of 1 magnitude in the corners of the images. As a result, a flat-field correction should be done if we want to perform absolute photometry. Because of Colibri's limited processing power and the fact that we are doing same-star differential photometry, however, we do not flat-field correct our images within our processing pipeline.

4.2 Mechanical

After coarse polar alignment, the mounts were polar-aligned more precisely using the drift-alignment technique. Pointing

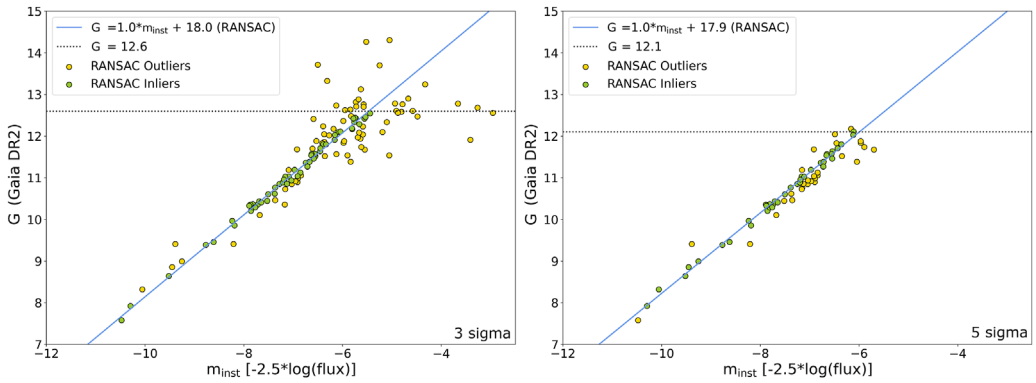


FIGURE 5
Gaia *G*-band vs. instrumental magnitude of 129 sources extracted with a detection threshold ≥3 sigma (left panel) and 69 sources extracted with a detection threshold of ≥5 sigma (right panel). from a 25 ms exposure.

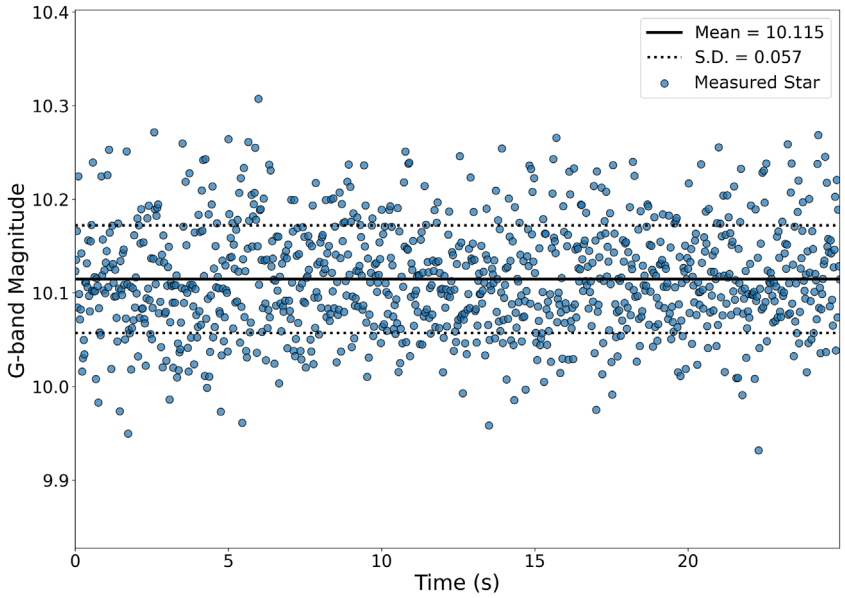


FIGURE 6
A magnitude vs. time light curve for a *G* = 10.1 mag star over a 25 s window in one thousand 25 ms exposures.

models to 10° altitude were then created using MaxPoint software from Diffraction Limited to quantify mount and telescope alignment. Mount alignment was then adjusted and new pointing models created in an iterative fashion until azimuth and altitude misalignments were below 10″. After alignment and pointing model corrections, we find that the pointing performance of all telescopes is typically within a few 10s of arcseconds of the desired position. With our large fields, this is sufficient for ensuring that all three telescopes are monitoring the same stars.

4.3 Photometry

When imaging rapidly with the KL4040 camera, background values can be seen to fluctuate by 10s of counts. To try to understand the effect that this has on rapid-imaging performance, we image a star-rich region of the sky (R.A.: 4.75h, Dec.: 72.75°) at our target frame rate of 40 fps without a filter. The images are corrected for bias and then stars are extracted using SExtractor (Bertin and Arnouts, 1996). A world coordinate system solution is obtained by processing an initial image of

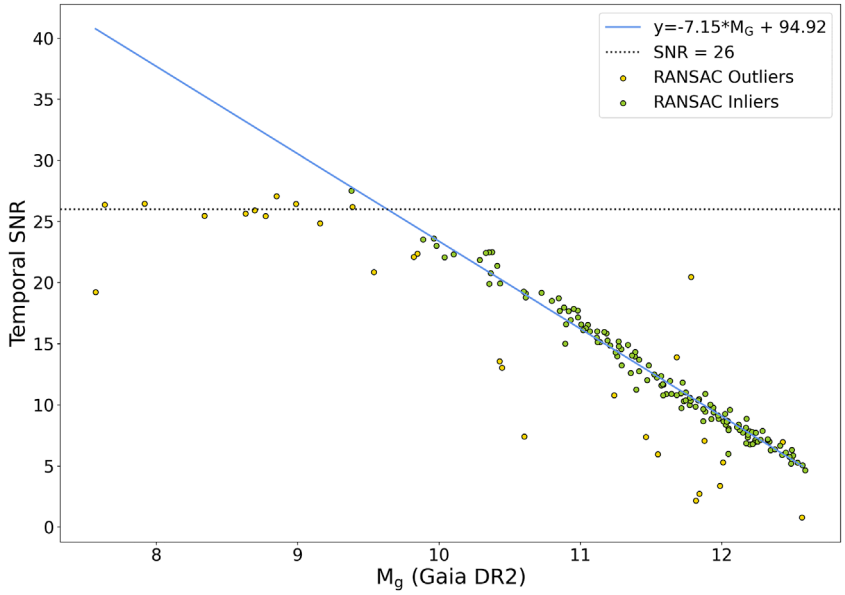


FIGURE 7
The temporal SNR of $G \lesssim 10$ mag objects, extracted from a field centred at an airmass of 1.37, is roughly SNR = 26. At fainter magnitudes, the temporal SNR decreases linearly to SNR = 5 at $G \approx 12.5$ mag.

the chosen stellar field through the astrometry.net algorithm (Lang et al., 2010). Stellar positions are cross-correlated with their Gaia Early Data Release 3 (EDR3; Prusti et al., 2016, Brown et al., 2021) coordinates. Instrumental magnitudes are then compared with Gaia *G*-band (400–860 nm) magnitudes to derive a first-order transformation equation (**Figure 5**). Using a Random Sample Consensus (RANSAC) algorithm to estimate the best fit to our data we find very similar solutions for each of the sigmas tested. One of the main benefits of the RANSAC solution is that misidentified stars or those with bad instrumental magnitudes will be classified as outliers and, as a result, have no impact on the final solution.

Stellar limiting magnitude has been defined Harris (1990) as the magnitude at which only 50% of the objects of that magnitude are detected. By injecting stellar sources with a range of magnitudes into real imagery, Harris counts the number of found sources with the IRAF/*DAOphot* software. The method we describe is slightly different, yet is still effective for our purposes. Instead of injecting synthetic stars into our images, we simply compare our extracted stars to known sources from the Gaia EDR3 catalogue to see at what limit we stop detecting objects. **Figure 5** shows the instrumental magnitude of detected objects to a limit of 3 sigma above the background plotted against the Gaia *G*-band magnitude of the star closest to the extracted position. Beyond about $G = 12.5$ mag, correlation between detections and real stars is lost and we set that as our practical limiting magnitude for a 25 ms exposure.

Although an SNR of 3 is considered to be the absolute limit for most systems, the actual limit for observing occultation events is somewhat higher. For the detection of KBOs, we set our stellar detection threshold to 4 sigma but realistically expect to be able to detect SSOs down to about 5 sigma above the background. We use the fitted relation in **Figure 5** to calibrate our conversion from Colibri instrumental to Gaia *G*-band magnitudes. Since Colibri's is a filter-less system, we do not seek to accurately transform the data to a standard photometric system. Our conversion is likely accurate to ±0.1 mag for most stars.

Using the same 40 fps dataset as above, we also examine the relative photometric temporal stability of the system. To do this, we look at the frame-to-frame residual changes in measured magnitude (instrumental or transformed) for all stars in the field. **Figure 6** shows the measurements for a randomly-selected $G = 10.1$ mag star over 1,000 consecutive 25 ms exposures (25 s sequence). The standard deviation is 0.057 mag. We use a derivative measure, the ratio of the mean flux to the r.m.s. scatter of the flux over a 60-s interval, as the "temporal SNR" of our flux measurements. The temporal SNR is approximately diagnostic of the minimum detectable depth for a stellar occultation. For example, at a temporal SNR of 10 the rms scatter of the light curve is 1/10 = 1%, so detectable occultation would be at least ~30–40% deep. Our first dip-detection pass in our occultation detection pipeline does indeed set a 40% threshold for detection (**Section 3.3**).

Figure 7 presents the temporal SNR of objects (airmass ≈ 1.37, extinction = 0.3 ± 0.1 magnitudes per airmass) with 25 ms exposures as a function of stellar *G* magnitude. The temporal SNR follows a linear trend for stellar magnitudes fainter than about $G = 9.5$ mag. Brighter objects, however, are best described by a

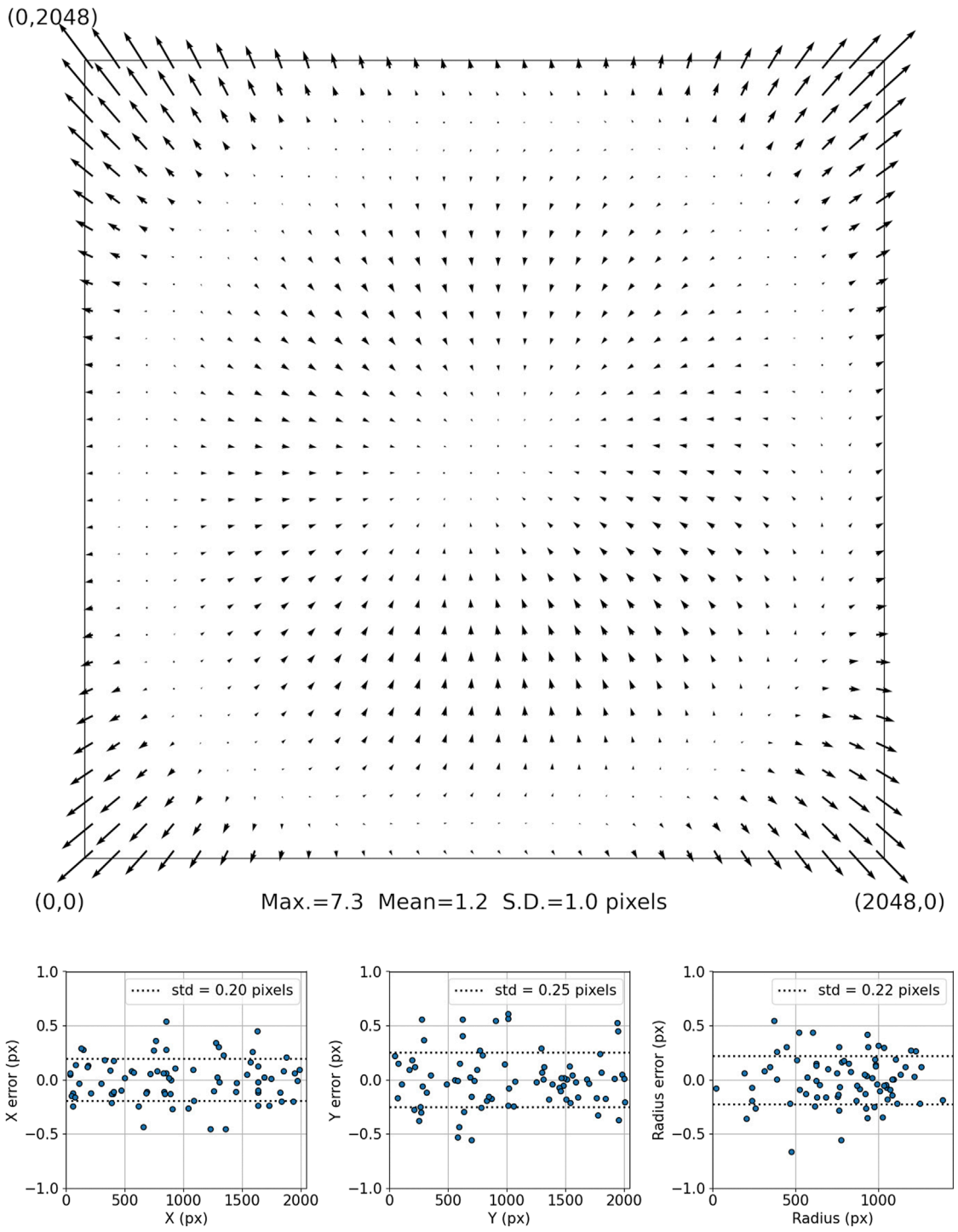


FIGURE 8
Distortion (scaled 20x) as computed using a third-order polynomial with a first-order radial term is shown in the upper panel. The lower panel shows the astrometric residuals after correction.

horizontal line at an SNR of about 26. This is likely an indicator that atmospheric scintillation becomes the dominant source of noise for bright objects.

When the standard deviations of the measurements for all stars against their mean magnitudes are plotted (**Figure 7**), they show a monotonically increasing standard deviation with increasing stellar magnitude. This provides a useful estimate of the threshold for the depth of a detectable dip at a given magnitude. The 0.035 mag error ($G \lesssim 9.5$ mag stars), for example, corresponds to 3.5% of the flux while a 0.2 mag error ($G = 11.5$ mag stars corresponds to 17%) of the total flux. Therefore, we can conclude that most >20%-deep occultations of brighter stars should be visible. Occultations of fainter stars are more likely to be detected under ideal conditions or if ≳50% depths occur under favourable occultation geometry. As already detailed in Pass et al. (2018), stellar brightness will be critical for the precision to which the parameters of a stellar occultation can be determined.

4.4 Astrometry

Using the SkyFit2 tool (Vida et al., 2021), a plate solution is computed using a third-order polynomial with a first-order radial term (**Figure 8**). When compared to Gaia EDR3, the results show the standard deviation of the residuals ranging from about 0.2 to 0.25 pixels (0.5″–0.63″) with no obvious dependence on distance from the field centre. Higher order fits have been tested, but there is no obvious benefit to using them given the low amount of distortion (**Figure 8**, upper panel) in the system and the fact that high astrometric precision is not a strict requirement for the detection of KBOs by stellar occultation. Although this represents a solution from a single pointing direction (altitude = 45°, azimuth = 0°), we do not find a large changes in the plate solution at different pointings.

5 Summary

The Colibri telescope array at Elginfield Observatory in Ontario, Canada, has been built for the purpose of detecting kilometre-sized KBOs by the serendipitous stellar occultation method. The construction used cost-effective, off-the-shelf components to meet the design goals of the Colibri project.

Using sCMOS cameras, Colibri has been designed to continuously monitor the night sky for serendipitous stellar occultation events. At 40 fps, each camera streams imagery at 4 Gb/s over a 40G fibre link to its own dedicated RAID. This generates up 20 TB of data per camera per night to be processed the following day by the processing pipeline.

The photometric and astrometric performance of the system has been measured. Photometrically, the system performs well with a limiting broad-band G magnitude of about 12.5 (temporal SNR = 5) in a 25 ms exposure. The relationship between instrumental and Gaia G magnitudes is linear with a 1:1 slope while light curve analysis shows stability in the measurements within the timescale of the 1-min data subsets. The standard deviation in the magnitude measurements increases with increasing magnitude, ranging from about 0.035 mag on $G \lesssim 9.5$ mag stars to ∼0.2 mag at the SNR = 5 limit. As this corresponds to a roughly 3.5%–17% variation in flux, occultations with ∼50% flux depths over several consecutive exposures should be detectable at even the faintest magnitudes. Astrometrically, the optics show some distortion and better than ±0.25 pixel (±0.6″) errors. As a result, a third-order polynomial plate solution is sufficient for this project.

The Colibri observatory is currently collecting a limited amount of data while testing the automation and processing pipeline routines. Full operational activities are expected to begin in the summer of 2022.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

MM—Technical contruction, automation coding, and testing. Writing of paper SM—P.I. of Colibri project RAB—Coding of Colibri processing pipeline RG—Automation coding RB—Modeling of observational requirements TM—Pipeline coding EP—Pipeline coding and proof-of-concept testing.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

References

Arimatsu, K., Tsumura, K., Ichikawa, K., Usui, F., Ootsubo, T., Kotani, T., et al. (2017). Organized Autotelescopes for serendipitous event survey (OASES): Design and performance. *Publ. Astronomical Soc. Jpn.* 69, 68. doi:10.1093/pasj/psx048

Arimatsu, K., Tsumura, K., Usui, F., Shinnaka, Y., Ichikawa, K., Ootsubo, T., et al. (2019). A kilometre-sized Kuiper belt object discovered by stellar occultation using amateur telescopes. *Nat. Astron.* 3, 301–306. doi:10.1038/s41550-018-0685-8

Bailey, M. E. (1976). Can 'invisible' bodies be observed in the Solar System? *Nature* 259, 290–291. doi:10.1038/259290a0

Barbary, K. (2016). Sep: Source extractor as a library. *J. Open Source Softw.* 1, 58. doi:10.21105/joss.00058

Bertin, E., and Arnouts, S. (1996). SExtractor: Software for source extraction. *Astron. Astrophys. Suppl. Ser.* 117, 393–404. doi:10.1051/aas:1996164

Bickerton, S. J., Kavelaars, J. J., and Welch, D. L. (2008). A search for sub-km kuiper belt objects with the method of serendipitous stellar occultations. *Astronomical J.* 135, 1039–1049. doi:10.1088/0004-6256/135/3/1039

Bickerton, S. J., Welch, D. L., and Kavelaars, J. J. (2009). Kuiper belt object occultations: Expected rates, false positives, and survey design. *Astron. J.* 137, 4270–4281. doi:10.1088/0004-6256/137/5/4270

Brown, A. G. A., Vallenari, A., Prusti, T., de Bruijne, J. H. J., Babusiaux, C., Biermann, M., et al. (2021). Gaia early data Release 3. Summary of the contents and survey properties. *Astron. Astrophys.* 649, A1. doi:10.1051/0004-6361/202039657

Chang, H.-K., King, S.-K., Liang, J.-S., Wu, P.-S., Lin, L. C.-C., and Chiu, J.-L. (2006). Occultation of X-rays from Scorpius X-1 by small trans-neptunian objects. *Nature* 442, 660–663. doi:10.1038/nature04941

Delsanti, A., and Jewitt, D. (2006). "The solar system beyond the planets,". *Mason. Solar system update*. Editors P. Blondel, and J. W., 267. doi:10.1007/3-540-37683-6_11

Harris, W. E. (1990). A comment on image detection and the definition of limiting magnitude. *Publ. Astronomical Soc. Pac.* 102, 949–953. doi:10.1086/132720

Jones, T. A., Levine, A. M., Morgan, E. H., and Rappaport, S. (2006). Millisecond dips in sco X-1 are likely the result of high-energy particle events. *Astronomer's Telegr.* 949, 1.

Lang, D., Hogg, D. W., Mierle, K., Blanton, M., and Roweis, S. (2010). Astrometry.net: Blind astrometric calibration of arbitrary astronomical images. *Astronomical J.* 139, 1782–1800. doi:10.1088/0004-6256/139/5/1782

Lehner, M. J., Wang, S.-Y., Alcock, C. A., Cook, K. H., Furesz, G., Geary, J. C., et al. (2012). "The transneptunian automated occultation survey (TAOS II),". *Ground-based Airborne Telesc. IV*. Editors L. M. Stepp, R. Gilmozzi, and H. J. Hall (Bellingham, WA, United States: of Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series), 8444, 84440D. doi:10.1117/12.925541

Lehner, M. J., Wen, C. Y., Wang, J. H., Marshall, S. L., Schwamb, M. E., Zhang, Z. W., et al. (2009). The Taiwanese-American occultation survey: The multi-telescope robotic observatory. *Publ. Astronomical Soc. Pac.* 121, 138–152. doi:10.1086/597516

Levison, H. F., and Stern, S. A. (2001). On the size dependence of the inclination distribution of the main kuiper belt. *Astron. J.* 121, 1730–1735. doi:10.1086/319420

Pass, E., Metchev, S., Brown, P., and Beauchemin, S. (2018). Pipeline for the detection of serendipitous stellar occultations by kuiper belt objects with the Colibri fast-photometry array. *Publ. Astronomical Soc. Pac.* 130, 014502. doi:10.1088/1538-3873/aa971f

Prusti, T., de Bruijne, J. H. J., Brown, A. G. A., Vallenari, A., Babusiaux, C., and Bailer-Jones, C. A. L. (2016). The Gaia mission. *Astron. Astrophys.* 595, A1. doi:10.1051/0004-6361/201629272

Roques, F., Doressoundiram, A., Dhillon, V., Marsh, T., Bickerton, S., Kavelaars, J. J., et al. (2006). Exploration of the kuiper belt by high-precision photometric stellar occultations: First results. *Astron. J.* 132, 819–822. doi:10.1086/505623

Roques, F., and Moncuquet, M. (2000). A detection method for small kuiper belt objects: The search for stellar occultations. *Icarus* 147, 530–544. doi:10.1006/icar.2000.6452

Roques, F., Moncuquet, M., and Sicardy, B. (1987). Stellar occultations by small bodies - diffraction effects. *Astron. J.* 93, 1549–1558. doi:10.1086/114438

Schlichting, H. E., Ofek, E. O., Sari, R., Nelan, E. P., Gal-Yam, A., Wenz, M., et al. (2012). Measuring the abundance of sub-kilometer-sized kuiper belt objects using stellar occultations. *Astrophys. J.* 761, 150. doi:10.1088/0004-637X/761/2/150

Schlichting, H. E., Ofek, E. O., Wenz, M., Sari, R., Gal-Yam, A., Livio, M., et al. (2009). A single sub-kilometre Kuiper belt object from a stellar occultation in archival data. *Nature* 462, 895–897. doi:10.1038/nature08608

Vida, D., Šegon, D., Gural, P. S., Brown, P. G., McIntyre, M. J. M., Dijkema, T. J., et al. (2021). The global meteor network - methodology and first results. *Mon. Not. R. Astron. Soc.* 506, 5046–5074. doi:10.1093/mnras/stab2008

Zhang, Z. W., Lehner, M. J., Wang, J. H., Wen, C. Y., Wang, S. Y., King, S. K., et al. (2013). The TAOS project: Results from seven years of survey data. *Astron. J.* 146, 14. doi:10.1088/0004-6256/146/1/14

frontiers | Frontiers in Astronomy and Space Sciences

TYPE Original Research
PUBLISHED 03 May 2023
DOI 10.3389/fspas.2023.952887

OPEN ACCESS

EDITED BY
David Leisawitz,
National Aeronautics and Space Administration, United States

REVIEWED BY
Amy Connolly,
The Ohio State University, United States
Carlos Alexandre Wuensche,
National Institute of Space Research (INPE), Brazil

*CORRESPONDENCE
A. J. Castro-Tirado,
ajct@iaa.es

SPECIALTY SECTION
This article was submitted to Astronomical Instrumentation, a section of the journal Frontiers in Astronomy and Space Sciences

RECEIVED 25 May 2022
ACCEPTED 16 January 2023
PUBLISHED 03 May 2023

CITATION
Hu YD, Fernández-García E, Caballero-García MD, Pérez-García I, Carrasco-García IM, Castellón A, Pérez del Pulgar C, Reina Terol AJ and Castro-Tirado AJ (2023), The burst observer and optical transient exploring system in the multi-messenger astronomy era. *Front. Astron. Space Sci.* 10:952887. doi: 10.3389/fspas.2023.952887

The burst observer and optical transient exploring system in the multi-messenger astronomy era

Y.-D. Hu[1], E. Fernández-García[1], M. D. Caballero-García[1], I. Pérez-García[1], I. M. Carrasco-García[2], A. Castellón[2], C. Pérez del Pulgar[2], A. J. Reina Terol[2] and A. J. Castro-Tirado[1,2]*, on behalf of a larger collaboration[1]

[1]Instituto de Astrofísica de Andalucía (IAA-CSIC), Granada, Spain, [2]Unidad Asociada al CSIC Departamento de Ingeniería de Sistemas y Automática, Escuela de Ingenierías, Universidad de Málaga, Málaga, Spain

The Burst Observer and Optical Transient Exploring System (BOOTES) was first designed as an asset of autonomous telescopes that started to be deployed in 1998, taking 24 years to be fully developed around the earth. Nowadays BOOTES has became a global network of robotic telescopes, being the first one present in all continents, as of 2022. Here we present the details of the network and review its achievements over the last 2 decades regarding follow-up observations of high-energy transient events. Moreover, considering the recent operations of neutrino and gravitational wave detectors, some hot-topic expectations related to robotic astronomy are discussed within the framework of multi-wavelength astrophysics.

KEYWORDS

robotic Astronomy, optical observations, BOOTES network, telescopes, multi-messenger

1 Introduction

The industrial revolution gave rise to technological advances, leading to the coinage of the word "robot" for the machines to replace workers in repeatable roles (Baruch, 1992). In science, robots quickly became popular, and one of the very first ones was a mobile robotic chemistry machine at Liverpool University which was designed to undertake repeatable chemical experiments in order to saving time and avoiding operating errors (Burger et al., 2020). In Astronomy, attempts to achieve some degree of automation were undertaken since the mid-20th century, especially regarding space satellites which can be treated as robotic systems because they can operate with self-power supply, command uploading and remote control (Ellery, 19 2003). These robotic systems can augment or replace human activity in space, such as the robotic arm on the international space station (ISS) which can install and replace equipment and perform external inspections of the station. In addition to their use in space, systems for ground-based telescopes have been developed to allow certain tasks to be executed automatically. Thus, four degrees of automation have been achieved over the last decades, ranging from automatic tasks and automated telescopes to remote instruments and robotic observatories, depending on their degree of automation and the level of human interaction (Baruch, 1992; Castro-Tirado, 2010). Since the 1980s, there have been several old telescopes that could be upgraded and operated robotically. The advancement of robotic telescopes became possible with the development of internet and computer technology, leading to the design and construction of several ground-based robotic telescopes, such as the BOOTES network discussed here.

2 The BOOTES network

The Burst Optical Observer and Transient Exploring System (BOOTES) network (Castro-Tirado et al., 1996; Castro-Tirado et al., 2012) is a worldwide robotic telescope network whose first prototype was proposed and designed under a Spanish-Czech collaboration framework. It achieved its first light in 1998. As the Spanish pioneer robotic observatory for optical transient searching and follow-ups, it has achieved multiple scientific goals, as detailed below.

As originally planned, the BOOTES network consists of seven stations eventually, four in the northern hemisphere and three in the southern hemisphere ensuring that there will always be at least one telescope covering the northern and southern parts of the night sky (Castro-Tirado et al., 2012). All stations are marked in **Figure 1**. Since the first light of BOOTES-1 in 1998 until the installation of BOOTES-7 in 2022, the BOOTES network has already deployed all its seven astronomical stations. These observatories are detailed in the following Table.

2.1 The BOOTES-network construction

2.1.1 The seven BOOTES astronomical stations

The BOOTES-1 observatory (B1) is located at Estación de Sondeos Atmosféricos in Centro de Expermentación de El Arenosillo which belongs to Instituto Nacional de Técnica Aerospacial (CEDEA-INTA) in Mazagón, Huelva, Spain (Jelínek et al., 2016). It contains three domes not far from each other, BOOTES-1A (B1A), BOOTES-1B (B1B) and BOOTES-1C (B1C). B1A is equipped with two wide-field CCD cameras ($4,096 \times 4,096$ pixels^2) in the same mount, one attached to a 400 mm f/2.8 lens which covers a $5° \times 5°$ field of view (FOV) and another connected with a 135 mm f/2 lens covering a $15° \times 15°$ FOV. In the B1B dome, there is a 0.3 m diameter Schmidt-Cassegrain reflector telescope mounted on a Paramount mount which covers a 15' ×15′ FOV. Both cameras are working with a clear filter which can be transformed to the R-band under the assumption of no intrinsic colour evolution of the optical counterpart. B1C is the dome which was used to store the Spanish-Polish collaboration project "Pi of the sky" which was a system of robotic telescopes with a wide field of view containing 4 units (16 CCD cameras) since 2010. After their retirement in 2020, this dome was re-furnished with a new mount pier and now hosts a 28 cm wide field of view camera operating since mid-2022. The BOOTES-2/TELMA observatory (B2) is located at the Instituto de Hortofruticultura Subtropical y Mediterránea La Mayora, which belongs to the Consejo Superior de Investigaciones Científicas and Universidad de Málaga (IHSM/UMA-CSIC) in Algarrobo-Costa (Málaga, Spain) which started its scientific operation in 2002 (Jelínek et al., 2016). A 0.3 m diameter Schmidt-Cassegrain telescope was first deployed with an attached wide-field camera similar to the one installed at B1B. The idea was to get B2 operating in two different modes: the stand-alone observation and the parallel stereoscopic mode. The latter mode allows for simultaneous observation together with the B1B telescope, which is located at a distance of 250 km. This setup allows for the discrimination of near-Earth detected objects up to a distance of 10^6 km. In 2008, a new high-speed slewing fast-camera and fast-filters telescope prototype was purchased in order to upgrade the 0.3 m telescope to quickly follow-up astronomical transients. The new telescope installed was a 0.6 m aperture Ritchey-Chretien one with f/8 and its optical tube truss made of carbon fibre making it a lightweight instrument with an overall weight of about 70 kg (see **Figure 2**). The equatorial mount NTM-500 from the Astelco company was chosen because it had the ability to achieve speeds up to 30 deg/s and accelerations up to 10 deg/s^2 according to the manufacturer. Its pointing accuracy is less than 5″, and the tracking accuracy is less than 1" per hour once a proper pointing model is achieved. A wide temperature range of −20°C–40°C is suitable for its operation. Meanwhile, the Andor iXon X3 EMCCD 888 was attached to the telescope in order to capture images on the $1,024 \times 1,024$ pixels CCD detector, which has a pixel size of 13-μm and a full resolution frame rate of 9 fps, thus providing a FOV of 10' ×10'. With such an ultra-light telescope, the mount has the capability to achieve fast slewing speeds and accelerations to reach any part of the sky in less than 8 s.

The BOOTES-3/YA (B3) observatory was the first observatory of the BOOTES network in the southern hemisphere. It was deployed in February 2009 at Vintage Lane, Blenheim (New Zealand, 27 m.a.s.l. Tello et al., 2012). Due to unavoidable environmental factors, it was moved from the Northern part of the South Island to the Southwest side of the South Island, with the site lying at the National Institute of Water and Atmospheric Research (NIWA) in Lauder, nearby Otago, since 2014. The same telescope and mount as that at BOOTES-2 were chosen thus making B2 the continuing prototype for the BOOTES network since then (see **Figure 2**). Hence, BOOTES-3 contains a 0.6 m aperture, f/8 beam Ritchey–Chretien telescope atop an Astelco NTM-500 mount. A variety of filters, from clear to SDSS u' g' r' i' to WFCAM/VISTA Z and Y can be attached to an Andor $1,024 \times 1,024$ pixel^2 CCD camera, thus able to provide multi-wavelength photometric observations within a 10' ×10′ FOV. The BOOTES-4/MET (B4) observatory is located at the Lijiang Astronomical Observatory in Lijiang, China and is operated by the Chinese Academy of Sciences. It was the second observatory of its kind in the northern hemisphere outside of Spain, and the first Chinese robotic astronomical observatory. It has been in operation since February 2012. This astronomical site is located approximately 120° east of the BOOTES-1 observatory, resulting in a time difference of around 8 h. In addition to this, there is also a 0.6-m Ritchey-Chretien telescope with an Andor camera mounted on an Astelco mount at the Cassegrain focus of the telescope. Similar to B2 and B3, the FOV of this telescope is 10' ×10′ with a filter set comprising a clear filter plus SDSS filter set (u' g' r' i') and both WFCAM/VISTA Z and Y filters.

The BOOTES-5/JGT (B5) observatory is the third observatory in the northern hemisphere and is located at the National Astronomical Observatory in Sierra de San Pedro Mártir (Bajo Califomia, México Hiriart, 2014; Hiriart et al., 2016). This astronomical site was chosen due to its longitude, which is approximately 120° west from the B1 observatory. It began operations in November 2015 and is equipped with the same equipment as the BOOTES-3 and BOOTES-4 stations. As shown in **Figure 3**, the three BOOTES stations in the northern hemisphere (B1, B4, B5) are located at roughly the same latitude with evenly divided longitudes. This placement secures that the BOOTES network can monitor the northern sky at all times with at least one observatory always available.

The BOOTES-6 (B6) observatory was the second astronomical station in the southern hemisphere and is located at Boyden Observatory, Maselspoort (Bloemfontein, South Africa). With a similar setup to B2, B3, B4, and B5 stations, it was deployed in late 2021.

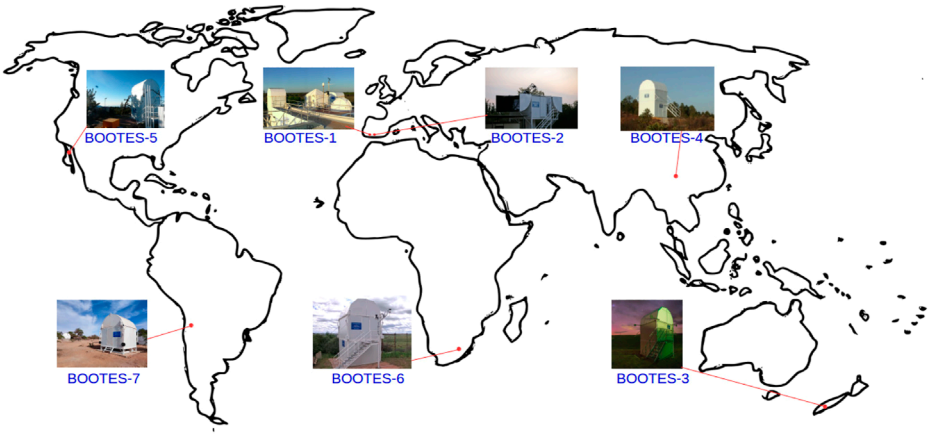


FIGURE 1
The BOOTES network map at present (2022), where all BOOTES network astronomical stations worldwide are marked with red points.

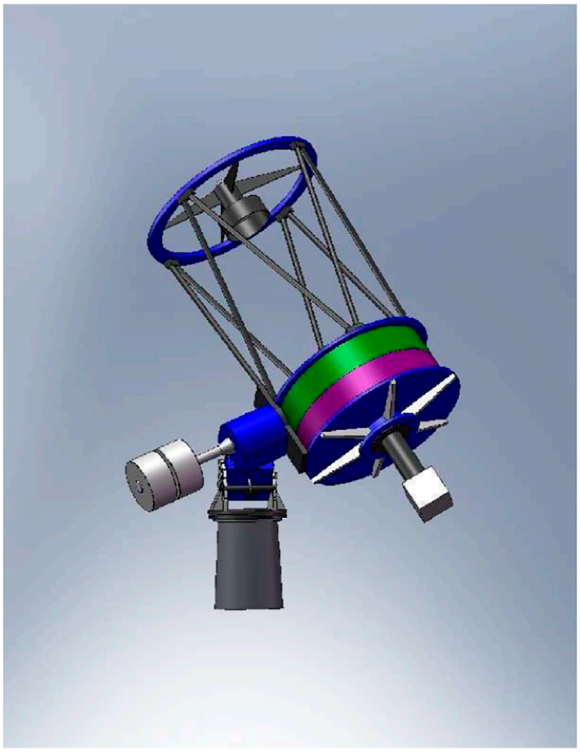


FIGURE 2
The ultra-light weight telescope concept was first used at the B2 station being the station we replicated in the rest of the BOOTES stations worldwide (Castro-Tirado et al., 2012).

Finally, the BOOTES-7 (B7) observatory is located in San Pedro de Atacama (Chile) and was deployed in late 2022. It is equipped with the same telescope, camera, and filter set as the other observatories of the network in order to ensure complete coverage of the southern sky, just as the network of observatories in the northern hemisphere already do, as shown in **Figure 3**.

The information for all BOOTES network sites is listed in **Table 1**, and the information for the telescopes is listed in **Table 2**.

2.1.2 The enclosures

The enclosures at the BOOTES observatories consist of two-halves with an overlapping roof that can open or close in a shell fashion under the action of electric motors. The overlapping direction of the two-halves is set to the upwind direction of the site's prevailing wind. Additionally, the two-halves are designed to open fully, allowing the telescope to access any part of the sky with an airmass of less than 5.8. The two-halves of the enclosures at the BOOTES observatories each have two electric motors connected through hinges and gears that provide the necessary torque to start and finish observations. The B2 dome, however, uses two oil pressure pumps and extra mechanical construction instead of motors. These motors/pumps are controlled automatically by the system, but they can also be activated manually during upgrading and commissioning inside. At each station, a weather station is mounted on a metallic tower near the dome position as an important part of the security system (see the description in the section below). The weather station includes a meteorological camera and precipitation and cloud sensors that work together to determine wind, cloud cover, rain, and humidity conditions. Besides, the outside dome surveillance camera for checking the dome's situation is also mounted at the same place. Another surveillance camera is installed inside the dome. The measured parameters from both units constitute the selection criteria for the control system to open/close the dome in less than 30 s.

2.1.3 The CASANDRA very wide-field cameras

In addition to a 0.6 m telescope, each BOOTES station is equipped with an all-sky camera named Compact All-Sky Automated Network Developed for Research in Astronomy (CASANDRA, Castro-Tirado, 2011). A $4096 \times 4,096$ pixels^2 CCD camera attached to a 16 mm f/2.8 lens provides a 180° FOV which can detect bright stars with a magnitude <8 mag near the horizon and approximately 10 mag at the zenith. The working mode of this camera is programmed to take images every minute which are used to monitor sky conditions and detect astronomical events, such as meteors.

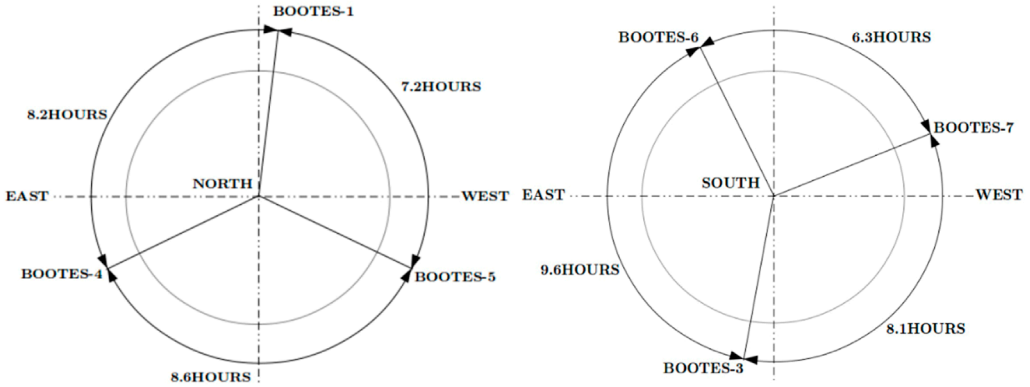


FIGURE 3
The location of BOOTES sites in the northern hemisphere (B1, B4, B5) and the southern hemisphere (B3, B6, B7) ensures that at least one telescope can monitor both the northern and southern skies at any night time (monsoon permitting) (Hiriart, 2014).

TABLE 1 BOOTES network sites location.

Site	Latitude	Longitude	Plus codes	ASL (m)	Site
BOOTES-1	37°05′58.2″N	6°44′14.89″W	8C9M37X8+2C	50	Mazagón
BOOTES-2	36°45′24.84″N	4°02′33.83″W	8C8QQX44 + GQ	70	Algarrobo-Costa
BOOTES-3	45°02′22.92″S	169°41′0.6″E	4V6FXM6M + QP	360	Lauder
BOOTES-4	26°41′42.8″N	100°01′48.24″E	7PR2M2WJ + FF	3200	Lijiang
BOOTES-5	31°02′39″N	115°27′49″W	85362GVP + JG	2860	Baja California
BOOTES-6	29°02′20″S	26°24′13″E	5G28XC63 + FF	1383	Maselespoort
BOOTES-7	22°57′09.8″S	68°10′48.7″W	2RWC + V7X	2440	Atacama

2.1.4 The COLORES imaging spectrograph

Spectroscopic observations can offer crucial insights into the nature of the sources, but fast follow-up observations are hindered by the time-consuming process of changing instruments and other factors on medium/large telescopes. This can create significant difficulties in obtaining spectroscopic observations of rapidly fading transient events during their early stages. Consequently, spectroscopic instruments mounted on robotic telescopes benefit from quicker reaction times and more efficient scheduling, which enhances the ability to detect fast-evolving transients. In this context, we want to highlight the COmpact LOw REsolution Spectrograph (COLORES Rabaza et al., 2013; Jelínek, 2014; Caballero-Garcia et al., 2014) we developed and mounted on the BOOTES-2 station. COLORES is a prototype for a light-weight spectrograph mounted on a small-size telescope and it has been functioning well so far. It is a Faint Object Spectrograph and Camera (FOSC) with a single optical path and wheels at multi-position installed. The slits, filters and grisms are inlaid in the wheels into the collimated space to prevent any changes in the focus plane. The idea is to make the instrument work in both imaging and spectroscopy modes, with the central object in imaging mode being the same as the object whose spectrum is observed in spectroscopic mode. As shown in **Figure 4**, three designed wheels are staggered vertically to the parallel beam, i.e. aperture, grism and filter wheels. Each one has 8 holes with a size of 2.54 cm ×2.54 cm per item, which can be replaced without dismounting the entire wheel. The aperture wheel is placed in the front of the collimator which is located at the Cassegrain focal plane. Here, 5 slots are occupied by the slits with widths of 25 μm, 50 μm, 75 μm and 100 μm and length of 9.3 arcmins (to suit the different atmospheric seeing conditions) and also a blank plate. Meanwhile, one empty slot is used for imaging mode and the other two slots for a future equipment upgrade. The grism wheel follows the collimator and has four grisms currently available, i.e. GTK19, P-SF68, SF2 and N-BK7, to cover the wavelength range of 3,800–11,500 Å with a spectral resolution of 15–60 Å under the combination of a fixed slit and one grism. The filter wheel is in between the grism wheel and the camera which has 7 slots used for clear, SDSS g' r' i', Bessel R, WFCAM/VISTA Z and Y filter together with an empty place for the light to pass in spectroscopic mode. During the direct imaging mode and acquisition image mode, the aperture wheel and the grism wheel remain fixed with their empty slots. When the telescope switches to the spectroscopic mode, the filter wheel is set to an empty slot and the chosen slit and grism are engaged. The wavelength calibration is achieved through calibrating the standard lamp spectrum. In front of the aperture wheel, two tubes in COLORES have been placed opposite each other and oriented perpendicular to the light path. There are two standard lamps, i.e. Krypton and

TABLE 2 Features of the BOOTES network of telescopes.

Site	B1A	B1B	B2	B3/4/5/6/7	CASANDRA
Type[a]	Ph	SC	RC	RC	Ph
Lens	400 mm + 135 mm	-	-	-	16 mm
Mirror	-	30 cm	60 cm	60 cm	-
Focus[b]	-	Ca	Ca	Ca	-
Focal ratio	f/2.8 + f/2	f/10	f/8	f/8	f/2.8
CCD	4096x4096	512x512	1024x1024	1024x1024	4096x4096
Pixel size	9 μm	16 µm	13 µm	13 µm	9 µm
Angular Resolution	4.39"+13.2′′	2′′	0.59′′	0.59′′	2.2′
FOV	5 °×5°+15 °×15°	17′x17′	10′x10′	10′x10′	180°
Filter[c]	C	C	g'r'i' C R ZY	u'g'r'i' C ZY	C
Mount	Paramount ME	Paramount MX+	Astelco NTM-500	Astelco NTM-500	-
Camera[d]	MG4	A887	A888	A888	MG4
Spectrograph	-	-	COLORES	-	-

[a]Photolens (Ph), Schmidt-Cassegrain (SC), Ritchey-Chretien (RC).
[b]Cassegrain focus (Ca).
[c]Clear (C).
[d]Moravian G4-16000 (MG4), Andor iXon EMCCD DV887 (A887), Andor iXon X3 EMCCD, 888 (A888).

HgAr lamps, installed in one of the tubes controlled by the electronic program. Another tube contains a 45°-tilted flat mirror which can be inserted into the optical path to reflect the standard lamp light to the camera or removed from the main light path by means of a motorized precision slide.

The COLORES imaging spectrograph was installed in B2 in 2012 (Jelínek, 2014). This low-resolution spectrograph is capable of providing preliminary estimates of the distance to bright cosmological gamma-ray bursts and rough estimates of the chemical abundance information with a resolution of 15–60 Å. For the time being, COLORES is only installed in B2, but in the future there will be a similar model installed in one of the Southern Hemisphere BOOTES observatories. Consequently, the BOOTES network with such light-weight imaging spectrographs shall provide spectroscopic observations for fast transients located at any position in the sky. Note that all the information about the telescopes mentioned in this section is listed in **Table 2**.

3 The BOOTES control system

The control system is an important part of a robotic telescope network, responsible for ensuring that all the telescopes are operating correctly. When BOOTES was first established, the control system was known as the optical transient monitor (OTM, Páta et al., 2001) which was used for the first two wide-field prototype telescopes. Then, a Linux-based platform called Remote Telescope System - second version (RTS2, Kubánek et al., 2006), was used to control all telescopes with the instrument driver programmed in C++ language. Currently, the BOOTES network has been upgraded to a user-friendly, ASCOM platform-based system that contains an interface for communicating with the telescope server host on the Windows operating system. Every BOOTES station has a copy of this system installed on its host server.

This system is composed of three sub-systems: 1) Targets manager 2) Object executor and 3) Dome controller.

The targets manager is running on the BOOTES site host server which runs a program to receive alerts automatically and supports a web-browser interface for communication with telescope users. Mainly, the targets manager is continuously listening to the output from the Gamma-ray Coordinates Network (GCN)/Transient Astronomy Network (TAN)[1] which distributes the locations of transients detected by various spacecraft (*Swift, Fermi, MAXI, INTEGRAL, IPN,* etc) and ground-based multi-messenger detectors (LIGO/Virgo, IceCube, HAWC, etc). Once a new transient position is received, the DakotaVoEvent module reacts to this alert by making it a target of opportunity (ToO) and assigning it the highest priority in the observation if its detectability and time window are suitable for the current site. By adding this higher priority target to the pending list, the ongoing observation will be aborted and the telescope will point to the ToO event. If there are not any new ToO objects, the telescope will be running with a prioritized observation list. As a backup, the BOOTES network has also installed a central server which communicates with all sites by sending any non-ToO and non-scheduled but interesting targets (with specific priority values) to a suitable observatory site based on their detectability. This method is generally used during the follow-up of objects within a large field.

The second part of the control system is the object executor, which is a program scripted in VisualBasic language, that drives the telescope mount and calls the Maxlm DL module to operate the CCD camera with a specific filter. The focuser module, FocusMax, is also steered with this software during the observation. Meanwhile, the calibrated universal time and geographic information are also

1 https://gcn.gsfc.nasa.gov/

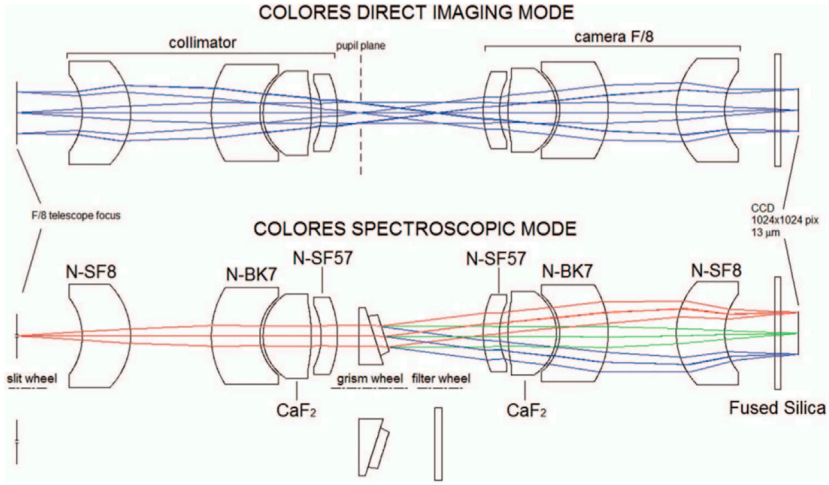


FIGURE 4
The optical path configuration of COLORES in direct imaging and spectroscopic modes (Rabaza et al., 2013).

provided by the GPS module. All of these communications are carried out using the common object model standard through the application programming interface.

The dome controller is responsible for opening and closing the dome depending on environmental conditions. It plays a crucial role as others in this system to protect the telescope from bad weather, such as storms, and ensures that observations can be performed safely. This part of the system gathers information (such as cloud rate, temperature, wind speed and humidity) from the weather station, the outer all-sky camera, the inner all-sky camera and a forecast report from the internet. It then sends this information to the database to determine if the weather conditions are suitable for opening the dome and conducting observations. The database provides quick and frequent feedback to close the dome in case the weather conditions will get worst. As a final safeguard, a rain sensor is also included which takes priority to close the dome in the event that the other weather monitoring systems fails to detect rain. During the observation, the MaxmlDL program gathered data and saved it as a standard fit image and the DS9 software transferred it to a companion . jpg format file as a snapshot, both of which are generated simultaneously and are kept in the host server. After the observation ends, the authorized scientific user can view the snapshot and/or download the fit files immediately via the user-specific link located on the main page[2] of the HTTP server at each station. Briefly, a sketch is made as shown in **Figure 5** to depict the process operating within the control system described in this section. In the sketch, the rectangle represents the hardware and the ellipse represents the software involved. Meanwhile, the arrow lines indicate the direction of the data flow and the instructions set.

Normally, there are two working modes of initialisation in the BOOTES system: the ToO mode, which has a higher priority, and the monitor mode. The ToO mode is activated when a new alert is received on the host server and the top priority target is assigned. If the object's position is attainable by the telescope at the time, the ToO mode will interrupt the current observational plan. Otherwise, this object will be moved into the pending monitor list until the next observation time window comes. The monitor mode only operates in the non-ToO periods. In this mode, the telescope executes the planned objects in order of their priorities from the scientific community. Depending on the triggering instrument, the received coordinates of the new event from GCN have an error region ranging of a couple of arcminutes (such as BAT/*Swift*) to several degrees (such as LIGO/Virgo). This error region can be enhanced by the object identification through other follow-up observations. For example, the X-ray observations of gamma-ray bursts can locate the event within several arcseconds. For the new ToO observation, the BOOTES network takes images to cover the entire error region. If there is not any counterpart confirmed by other facilities, the obtained images from the BOOTES network are used to find the new object through the difference imaging method, where a late-time image or resampling images from the Pan-STARRS and the 2MASS catalogs within the same field of view will be used to calculate the residual image. During the monitor mode operations, the difference imaging method can also be used to detect any kind of optical flashes in the image. Once the optical counterpart is identified, the telescope points to the updated position and continues monitoring it. As a consequence of being a global network, the time difference between different sites in the northern and southern hemispheres allows for long-term monitoring of any kind of object.

4 Scientific goals and results of the BOOTES network

With the capability of fast slewing and quick reaction, the scientific goals of the global BOOTES network were first set as below:

a) Observations of GRB optical counterparts: from the prompt emission to the afterglow. Simultaneous multi-wavelength detections of optical counterparts to GRBs have been obtained in some cases with white band magnitudes in the range of ~5–10. These observations provide important results on the central engine

2 http://bX.bootestelescopes.net/(With "X" ranging from 1a to 7 depending on the given BOOTES telescope).

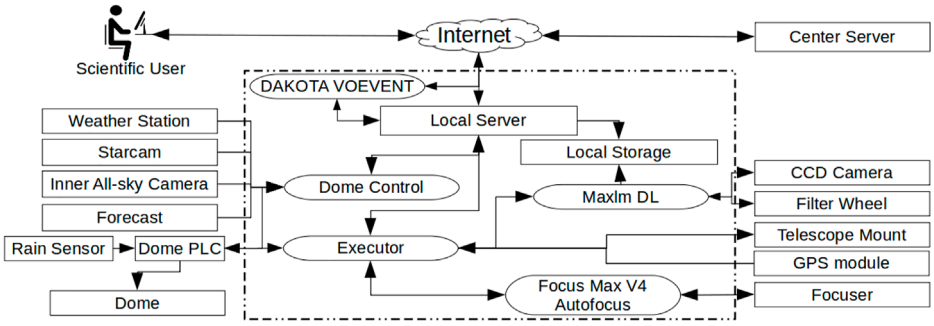


FIGURE 5
The control system is sketched in each observatory site. The dashed square contains the host server, which runs all modules of the telescope. The arrows mark the direction of the data flow and/or control commands.

of the violent emitters which can be executed and monitored by BOOTES telescopes due to their fast slewing capability.

b) The detection of optical transients of astrophysical origin. These events could be related to new astrophysical phenomena, perhaps associated with Fast Radio Burst (FRB), Neutrino sources, Gravitational waves (GWs), Quasars (QSOs), Active Galactic Nuclei (AGN) and Tidal Disruption Events (TDEs).

c) Ground-based support for the space missions, including the ESA's International Gamma-Ray Laboratory (*INTEGRAL*) and the NASA's Neil Gehrels Swift Observatory (*Swift*) satellites and also future missions, such as the Space Variable Objects Monitor (*SVOM*), to monitor high energy sources.

d) Monitoring astronomical galactic objects which include meteors, asteroids, comets, variable stars and novae, etc.

Since its first operation in 1998, there have been fruitful scientific results and outreach projects achieved during its 24 years of existence. Here the main topics in the field of astrophysical transients are listed in the following sections, as well as the BOOTES network's contributions to public outreach.

4.1 Gamma-ray bursts

Gamma-ray bursts (GRBs) are the most energetic explosion phenomena in the Universe which have a duration from several seconds even up to thousands of seconds in their gamma-ray emission (Klebesadel et al., 1973; Gehrels et al., 2009). Normally, they can be classified into two categories (Kouveliotou et al., 1993): short GRBs (SGRB, < 2s; Berger, 2014) and long GRBs (LGRB, >2s; Woosley and Bloom, 2006) based on their temporal scale. Their gamma-ray emission is followed by a longer-lived fading emission (detected from the X-ray to the radio domain) which is called the "afterglow". This afterglow is produced when the relativistic fireball interacts with the surrounding medium to generate external shocks which can be used to pinpoint and study the GRBs and their host galaxies' properties (Piran, 1999). Furthermore the afterglow can be detected even months after the burst. Nowadays, the LGRBs have bright afterglows that can be studied in great detail. It has been found that LGRBs are associated with Type Ib/c supernova explosions which indicate that the progenitors are collapsed massive stars (Woosley and Bloom, 2006). SGRBs have been linked with the merger of compact objects, such as neutron star binaries (BNS) or neutron star-black hole binaries (NS-BS) (Narayan et al., 1992). The short GRB 170817A was the first electromagnetic counterpart coincident with the gravitational wave event GW 170817A which unambiguously confirmed that BNS mergers are at least part of the mechanism that produces SGRBs (Abbott et al., 2017). Furthermore, in some events, it was found that the magnetar giant flare and underlying supernova components are also related to SGRBs which indicates that this mysterious phenomenon has multi-faceted characteristics intrinsically (Castro-Tirado et al., 2021b). Since GRBs were first detected by the Vela satellite in 1967 (Klebesadel et al., 1973), more than half a century has passed since the GRB field is still the frontier of multi-messenger astronomy. With generations of instruments involved in this research field, the method of GRBs' localization is described below. The space detectors in the high-energy band, such as *Swift* and *Fermi* can localise the burst position with GRBs' prompt emission and then circulate this result to the astronomical community all over the world through the Gamma-ray Coordinates Network (GCN). Ground facilities can follow up their position in order to capture their afterglow. Their optical counterparts play an important role in providing key information about the bursts themselves, such as color index and redshift, as well as the physical properties of their circumburst medium and host galaxies. However, observing the afterglows of GRBs remains challenging due to their short temporal scale, fast-decreasing brightness and faint afterglows. To address this, the BOOTES framework has implemented a strategy for GRB follow-up observations that aims to minimize the time delay between receiving the position and starting the observation, based on its autonomous reaction to triggers. Once the trigger is received by the telescope server, the narrow/wide-field telescopes slew to the burst position automatically as soon as a couple of seconds if the position is reachable. Meanwhile the all-sky camera will keep monitoring the sky in case that the burst has a very bright counterpart in the optical.

Since the first BOOTES network operations, one of the main goals has been to observe all triggered bursts in order to detect their optical afterglows. However, their intrinsic faintness limited their detection rate. Totally, there have been 196 reports published on GCN Circulars for early time observations of 182 GRBs based on BOOTES results (Jelínek et al., 2010; Jelínek et al., 2016). For 48 of these events we have found/confirmed their optical afterglows (see **Supplementary Appendix S1A**). Other observations have not reported in this manner due to the timeliness of their observations but are published elsewhere.

The count plot of the circular number (see in **Figure 6** left panel) clearly shows that the early detection rate has increased with the completeness of the BOOTES network construction and the following upgrades. The automation makes it possible to reach the burst position in 3 s after trigger and observations can provide detections as deep as 21 mag. In some cases, follow-up observations are executed by several telescopes from different sites. Though the all-sky camera only provides an upper limit of ~10 mag, the observations can be performed simultaneously/semi-simultaneously during the prompt emission phase which can be used to detect the early phase of the event similar to the naked-eye burst, e.g. GRB080319B (Racusin et al., 2008). This proves the capability of the BOOTES network to search for and provide the early observations of the optical afterglows of GRBs. These continuing observations can be used to constrain the afterglow evolution models. For example, in the case GRB080603B (see **Figure 6** right panel), the optical light curve can be fitted with a broken power-law showing a smooth transition between two decay epochs from $\alpha_1 = -0.55 \pm 0.16$ to $\alpha_2 = -1.23 \pm 0.22$ (Jelínek et al., 2012). Thanks to the spectral indices measured, it can be suggested that this burst is a case of a stellar wind profile expansion in a slow cooling regime.

4.2 Fast-radio bursts

Fast radio bursts (FRBs) are a new type of high energy transients discovered in the 21st century (Lorimer et al., 2007). They were first named "Lorimer bursts" because of his contribution to the first FRB event detection with the Parkes telescope in Australia in 2007. This phenomenon has a characteristic time-scale of a millisecond duration in its MHz-GHz radio emission with a high dispersion measure value (Thornton et al., 2013). With the accumulation of new detections, it has been found an isotropic sky distribution instead of the high latitude region distribution which suggests a cosmological origin. Since their first report, a number of radio facilities have been conducted to search for FRBs, such as the Canadian Hydrogen Intensity Mapping Experiment (CHIME), the Deep Synoptic Array (DSA) and the Five-hundred-meter Aperture Spherical radio Telescope (FAST). With their joint efforts, the FRBs detection rate has increased and it has been found that there are two main types of bursts, i.e. repeating/non-repeating, among which most of them are the non-repeating cases. Since their millisecond duration, it is difficult to take follow-up observations except for the repeating ones. The first repeating event, FRB121102A, was identified with the Arecibo Observatory and it was found to have a 157-day cycle which enables the precise localization of the burst place (Spitler et al., 2014). It has been found that the host galaxy is a low-metallicity star-forming dwarf galaxy at a redshift of z = 0.193. Recently, CHIME published its new catalogue (CHIME/FRB Collaboration et al., 2020; CHIME/FRB Collaboration et al., 2021) on FRBs hunting which included 535 events and 61 bursts from 18 recorded positions while other 474 events are one-off bursts. With these repeating FRBs, observations in other wavelengths become possible, spanning from optical, X-ray to gamma-ray, in order to place constraints on their radiation mechanism. There are several ways to search for the optical or high-energy counterparts of FRBs, such as using the same procedure as for triggering a GRB observation, monitoring burst fields with wide field telescopes, or targeting the repeating FRBs directly. Through these efforts, we are learning more about bursts, their multi-band counterparts, and the host galaxies they belong to, which is helping us to better understand this mysterious phenomenon.

As the BOOTES network is composed of several narrow and wide-field cameras (Castro-Tirado et al., 2012), this has proven to be a good platform for optical FRB counterpart searching. Like for the GRB follow-up observations mode, observing campaigns in other wavelengths can trigger the BOOTES telescopes to react simultaneously/semi-simultaneously for obtaining images in the burst active phase. On 28 April 2020, a bright FRB from the Galactic magnetar SGR 1935 + 2154 was captured with the CHIME radio telescope and STARE2 (Survey for Transient Astronomical Radio Emission 2) radio array (CHIME/FRB Collaboration et al., 2020; Bochenek et al., 2020). Meanwhile, the FAST telescope observed the same position in four sessions but a burst happened between the third and fourth sessions (Lin et al., 2020). During the third 1-h session, the magnetar became very active and emitted 29 bursts as observed by the *Fermi* satellite. Following the previous high-energy detection trigger of this magnetar, the BOOTES network responded it globally and there were simultaneous images taken from the BOOTES-3 station during its emitting episode. See **Figure 7** for the timeline of the multi-wavelength observations. The multi-burst phase was observed in gamma-ray, optical and radio. In a series of images in the Z-band obtained during the burst and in the simultaneous 60 s exposure frames we got a limiting magnitude of 17.9 mag. Considering the extinction correction, this limit corresponds to a peak flux of $F_{\mu,opt} \leq 4.4$kJy for 1 ms combined with a radio counterpart flux of $F_{\mu,FRB} \geq 1.5$MJy for 1 ms that lead to the flux ratio between fast optical bursts (FOB) and fast radio bursts should follow $\tau \leq 10^{-3}$. This stringent limit gives the first meaningful constraint on the FOB model parameters. Similarly, other facilities also follow up FRBs, such as the Zwicky Transient Facility (ZTF Andreoni et al., 2020), the Ground-based Wide Angle Camera (GWAC, Xin et al., 2021) and Apache Point Observatory (APO, Kilpatrick et al., 2021). However, the non-detection of these elusive optical counterparts presents a challenge for the BOOTES network in the future.

4.3 Gravitational waves electromagnetic counterparts

Massive stars are the most interesting and mysterious objects in the sky since their late evolution could be the best laboratory to produce catastrophic phenomena during the birth of black holes and neutron stars and due to their interaction with surrounding objects (Hughes, 2009). Therefore, they constitute an ideal natural environment to search for new physics. Theoretically, most of the energy in the two compact objects' merger process is released through the electromagnetic radiation that is produced together with the gravitational waves (GW) propagation containing part of the potential energy. This could be a new probe to investigate the information about their host and the characteristics of the objects (e.g. mass, spin-orbit, etc.). The first GW detector was proposed and designed in the 1960s with large cylinders of aluminium (Weber, 1968). After about half a century of research, the new generation of GW detectors used the laser interferometry method and there are already several main detectors built, such as the Laser Interferometer Gravitational-wave Observatory (LIGO, Livingston and Hanford at America) and Virgo interferometer (Virgo, Pisa at Italy). More detectors are under construction to join observation in the near future, such as

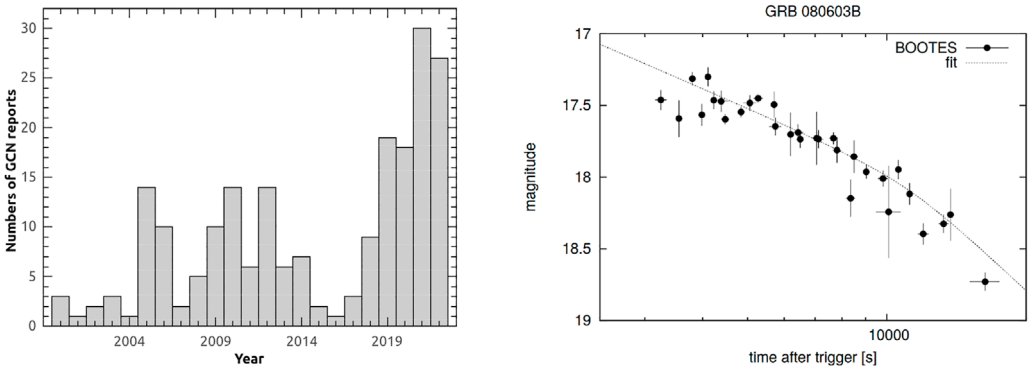


FIGURE 6
Left panel: BOOTES network GCN reports statistics until 2022. Right panel: GRB080603B optical observations with the BOOTES network with a smooth broken power-law that can fit well the afterglow (Jelínek et al., 2012).

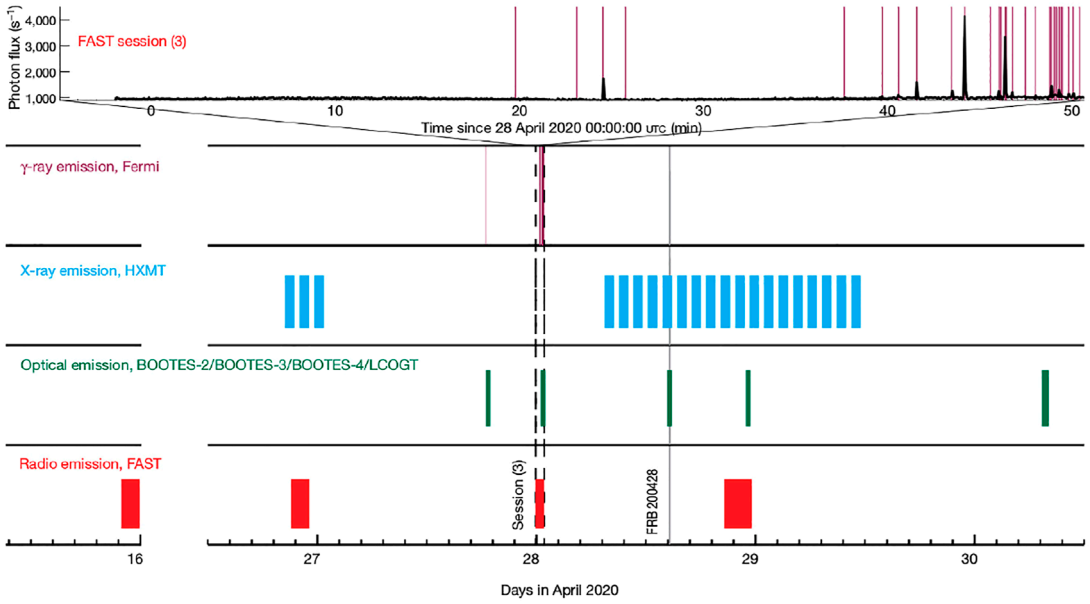


FIGURE 7
The timeline of SGR 1935 + 2,154 observations with Fermi, HXMT, BOOTES, LCOGT and FAST, i.e. from radio to gamma-ray bands (Lin et al., 2020).

the Indian Initiative in Gravitational-wave Observations (IndIGO). During the LIGO-Virgo scientific operation, the first GW event was detected on 14 September 2015, which was confirmed to be generated from two ~30 solar mass black holes merging at a distance of 410^{+160}_{-180}Mpc (Abbott et al., 2016). Together with the GW signal, the gravitational potential released in the form of electromagnetic (EM) radiation, reached earth simultaneously which could be used to constrain their related physical processes. Optical observations, which have well-developed techniques in the visible wavelength range, can be particularly useful in searching for the EM counterpart of GW events. Furthermore, the optical follow-up observation of the optical counterparts can provide multi-colour evolution to distinguish between different physical models. Along with the observations at other wavelengths, this will push multi-messenger astronomy ahead. Currently, the first discovery of an EM counterpart of a GW signal was found to be associated with a short gamma-ray burst, i.e. GRB170817 (Abbott et al., 2017). Due to the transient's short time scale, a telescope with fast localization and rapid follow-up capabilities was key for studying its properties. Since the BOOTES network is a robotic telescope system, it is well-suited for studying this type of transient event.

Since the LIGO-Virgo completed their construction, they have already made three joint scientific observing runs: 12 September 2015–19 January 2016 (O1), 30 November 2016–25 August 2017 (O2) and 1 April 2019–27 March 2020 (O3) (The LIGO Scientific Collaboration et al., 2021). The localization uncertainty was of several thousands deg^2 until Virgo's joint later which decreased the error region down to 28 deg^2 in the case of GW170817 (Abbott et al., 2017). Facing bigger error regions, the strategy used in the BOOTES network is shown in **Figure 8**. The BOOTES Network has both wide and narrow FOV telescopes, which can observe the new events in different ways. When a new GW alert is

received, the whole region is observed using a mosaic approach with the wide-field camera B1A. If that alert is identified as being related to neutron stars (NS), then the galaxy candidates in the error region are given higher priority follow-up observation using the narrower FOV telescopes.

During O1, the BOOTES network observed the event GW150914 with the CASANDRA camera at the BOOTES-3 station (Abbott et al., 2016). The image was taken simultaneously and no prompt optical counterpart was found. Due to the poor weather conditions, it only provided an upper limit of 5 mag. While this value sets a wide constraint, the BOOTES observation provided the earliest image corresponding to the first GW alert and demonstrated that its wide field coverage is sufficient for searching for GW counterparts. During O2, the milestone event of GW170817 was detected in images taken from the BOOTES-5 telescope. The magnitude recorded was $r = 18.2 \pm 0.45$ which is brighter than the predicted flux of optical afterglows but is consistent with other contemporaneous measurements (Zhang et al., 2018). During O3, there have been a total of 72 alerts, with 16 of them being NS-related merger events. The BOOTES network followed up on 55 of these events, including 13 NS-related mergers. Although the 76% of these alerts triggered BOOTES telescopes, there was not any new object detected in these images thus giving only a typical 3-σ upper limit of 20 mag (Hu et al., 2021).

4.4 Neutrino burst and blazar monitoring

The detection of neutrinos is another important aspect of multi-messenger astronomy which contains pieces of information essential for the understanding of high-energy events since these elementary electrically neutral particles have been detected arising from some nearby astrophysical sources (Learned and Mannheim, 2000). They only interact through the weak force which makes them very difficult to be detected. On the other hand, they can point out the source position directly. Tracking the Cherenkov radiation photons is the way to capture these events nowadays. The IceCube observatory in Antarctica is the largest neutrino detector currently. It consists of 86 strings, each connecting 60 digital optical modules, which are distributed in the Antarctic ice within one square kilometre area evenly, i.e. a km^3 cubic of ice as its volume providing the detectable energy range from 0.1 TeV to beyond 1 EeV (Aartsen et al., 2013). Similar ideas to design the neutrino detectors are also used such as ANTARES which is located in the Mediterranean Sea. For the time being, neutrinos from SN 1987A in the Large Magellanic Cloud were detected by the Kamiokande experiment which was the first astrophysical source of neutrinos ever identified (Schaeffer et al., 1987). The second detection of astrophysical origin was in the direction of the flaring active galactic nucleus AGN TXS 0506 + 056 (IceCube Collaboration et al., 2018), a BL Lac object. Although several models have predicted the association between neutrino and other catastrophic events in the Universe, no direct coincident evidence supporting this point has been made so far.

The BOOTES network also responds to neutrino events as one of its scientific targets and uses a similar method to the one used for GRB follow-up observations. Once an alert of this kind of event is received, the BOOTES telescopes point towards their origin directly to search for any new candidates that may be related to the neutrino source. For example, the second neutrino event detection, i.e. IceCube-201114A, was found to be related to the blazar NVSS J065844 + 063711 which was included in the *Fermi*-LAT fourth source catalogue with the name of 4FGL J0658.6 + 0636. The BOOTES-2 station monitored it for several nights in order to check its short-time variability (de Menezes et al., 2022). Besides, long-term optical monitoring of variability in blazars was attempted for the object S5 0716 + 714 (Wang et al., 2019) which is a candidate of high energy neutrino and high energy radiation. The BOOTES-4 station continued monitoring this source in multiple bands for several years and was the first facility to discovery its long-term variation pattern: a strong flatter when brighter (FWB) trend at a low flux state and then a weak FWB trend at a higher flux state, which was interpreted as the acceleration and cooling mechanisms of different electron's populations in the relativistic jet.

4.5 Outreach

In addition to the scientific results mentiond above, the BOOTES network also plays an important role in the public science education. By increasing scientific literacy and opportunities among young people, it helps to foster scientific vocations and provides opportunities for interested individuals to get involved in real research projects.

4.5.1 GLORIA

In 2009, the use of a worldwide network of robotic telescopes for educational purposes was proposed to the European Union, dubbed the GLObal Robotic telescopes Intelligent Array (GLORIA, Mankiewicz, 2013). This intelligent array devoted a fraction of the available observing time of existing telescopes for public use and it was the first large-scale robotic telescope network with free access. This project was initially funded in October 2011 under the support of the European Union within the Seventh Framework programme (EU FP-7) for research and technological development including demonstration activities and lasted 3 years. Finally, twelve institutions from seven countries (Spain, Chile, Czech Republic, Ireland, Italy, Poland and Russia) participated in this project and made use of 18 telescopes working in different scientific fields. As part of the Spanish contribution, three of the BOOTES Network telescopes (BOOTES-1, BOOTES-2, and BOOTES-3) were part of GLORIA. By using web 2.0 technology, anyone could freely access and obtain nearly real-time images of the night sky using a working telescope. There were two modes of observational application: the online mode (Sun observation, interactive night sky observation, scheduled night sky observation) and the offline mode (archival images in the database of GLORIA or from other databases including the European Virtual Observatory). The main goals of this project were to increase the number of telescopes and the number of scientists and citizens interested in astronomy, in order to expand the scope of research in these areas. By making knowledge freely accessible through this project, public motivation to engage in science education was enhanced and barriers to access were reduced. During the project, 5 astronomical events (4 eclipses and a transit of Venus) were broadcasted through the GLORIA network with associated educational activities in schools of the partner countries. Furthermore, scientific research was also conducted using GLORIA, including the observation of the active star DG CVn (Caballero-García et al., 2015) and the eccentric eclipsing binaries in our galaxy (Zasche et al., 2018).

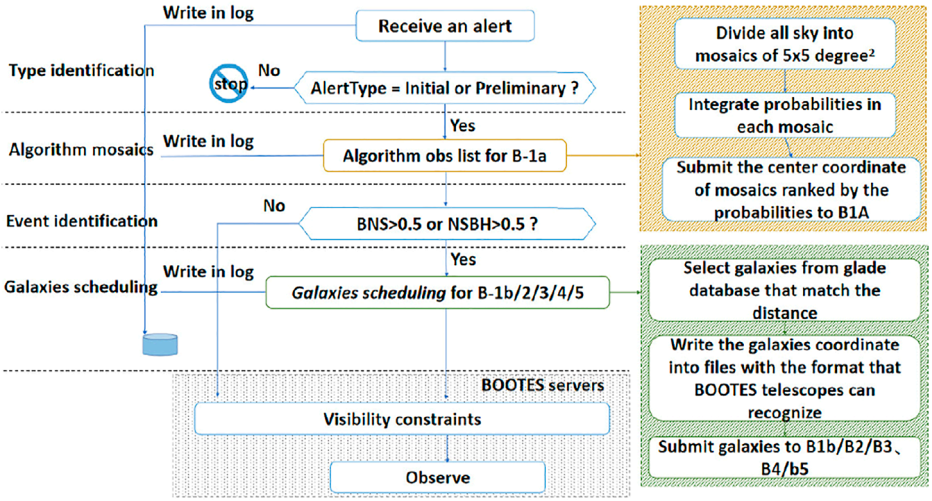


FIGURE 8
The strategy used in the BOOTES network for the GW observation.

4.5.2 ScienceIES

Facing the fact of the decreasing number of students in Science, Technology, Engineering and Mathematics at Spanish Universities in the early 2000s, an educational project at the Andalusian level started to bring high school students to work together with scientists in Andalusia (the southernmost autonomous community in Peninsular Spain) in 2010. This project named "Proyecto de Iniciación a la Investigación e Innovación en Secundaria en Andalucía (PIIISA)" and was then dubbed with ScienceIES (Castro-Tirado et al., 2021a). It is a distinctive education way to teach natural sciences including Astronomy to high-school students (at age of 15–17). This project provides the chance to join together high school students, their teachers and scientists to solve a practical problem and undertake the experiment in schools or laboratories. Students involved in these proposed projects are required to write evaluations, take interviews and complete surveys under the supervision of scientists. Normally, during the middle of every academic year, introductory lectures for ScienceIES are held at university laboratories or research centers, spaced 3 days apart. These experiments and/or observations needed in this project are carried out at the same place to obtain the preliminary result which are subsequently evaluated by their high-school teacher and supervisor after being presented in public. Finally, research projects from each province in Andalusia that receive high ranking scores will be presented in the form of oral talks or posters at a 1-day conference (held in May) in the main town at the regional level. The BOOTES network has been donating a portion of its telescope observing time to various astronomical research projects in PIIISA since 2013. Previous projects have included investigations of X-ray binaries, the local group of galaxies, and meteor storms.

5 Robotic telescope networks worldwide

Robotic telescopes have demonstrated many advantages over conventional telescopes, including fast reaction times, long-term monitoring capability, unmanned operations, lower costs, and more. It is clear that a worldwide network of robotic telescopes, comprising multiple sites, will enhance these advantages and improve observational efficiency. Like the BOOTES Network of Robotic Telescopes, there are several projects that involve deploying a network of robotic telescopes. Some of these projects have already been decommissioned. The following is a list of these projects.

- ROTSE-III (Robotic Optical Transient Search Experiment)[3]: A network which consisted of four 0.45 m diameter robotic Cassegrain telescopes distributed in different counties. Since 1998, ROTSE was devoted to searching optical transients and it operates with a wide FOV (1.85° × 1.85°) camera onto f = 1.9 telescopes without filters (Akerlof et al., 2003).
- LCOGT (Las Cumbres Observatory Global Telescope)[4]: It is a network composed of 25 telescopes, including 2 × 2-m, 13 × 1-m and 10 × 0.4-m, where the 2-m telescopes were operated by RoboNet (Tsapras et al., 2009) and purchased by LCO. They are distributed in 7 sites worldwide with the Ali Observatory station (the only one in Asia, still under construction in 2023). Optical imagers and spectrographs are designed and used in this network. It is operated by using a software scheduler which continuously optimizes the observing schedule of each telescope to monitor any target in the night sky (Pickles et al., 2014).
- MASTER (Mobile Astronomical System of Telescope-Robots)[5]: This network has 9 sites worldwide (just missing an station in Oceania to achieve complete coverage) which all are installed with a two-tube aperture system. Each tube is a 0.4-m telescope equipped with a 4k ×4k CCD camera with a scale of 1.85"/pixel and a universal photometer with Johnson-Cousins (BVRI) and polarizer filters. Its main scientific goals are not limited by the

3 http://www.rotse.net/
4 https://lco.global/
5 http://observ.pereplet.ru/

prompt optical emission of GRBs but the discovery of uncataloged objects (Gorbovskoy et al., 2013).

- SONG (Stellar Observations Network Group)[6]: It is a Danish-led project to design and construct a global network of 1-m class robotic telescopes in 8 nodes to undertake long-time monitoring. So far three nodes have been built at Teide Observatory, Mt. Kent Observatory and Delingha Observatory. Lucky-imaging camera and a high-resolution spectrograph have been mounted in these telescopes to suit the requirement of the main observational goals (asteroseismology and gravitational microlensing; Grundahl et al., 2009; Andersen et al., 2016).
- TRTN (Thai Robotic Telescope Network)[7]: A network of several 0.6 m and 0.7 m telescopes at 5 sites worldwide which is used for astronomical research, education and public outreach activities. Both telescopes have a set of Johnson-Cousins (BVRI) filters and a 2k ×2k CCD camera to provide photometric observation (Soonthornthum, 2017).
- TAROT (Télescopes à Action Rapide pour les Objets Transitoires)[8]: It is a French-led network which consist of 2×0.25-m (FOV $1.8° \times 1.8°$), 1×0.18-m ($4° \times 4°$) telescopes at three sites worldwide in addition to the 1-m Zadko telescope (FOV 20" ×20") at Gingin Observatory. They can be used to take photometric images with Sloan filters for multiple science goals, such as resident space objects, GRBs (Boer et al., 2017).
- TAT (Taiwan Automated Telescope) Network: This network is planned to install several robotic telescopes worldwide. Each site has installed a 9 cm Questar telescope to provide a variable FOV but it has been set to 0.62° square currently. Images in UBVRI filters can be obtained which are dedicated to photometric measurements of stellar pulsations to study the stellar structure and evolution (Chou et al., 2010).
- MicroObservatory: It is a worldwide network that comprises five robotic 0.91 m tall, 15 cm aperture, reflecting telescopes. Each unit is equipped with photometric filters (BVRI) and one neutral density filter which can provide a FOV of ~1 square-degree. This project was dedicated to public education but also can be used in the scientific field, such as transiting exoplanets monitor (Fowler et al., 2020).
- SPECULOOS (Search for habitable Planets EClipsing ULtra-cOOl Stars): This is a network of 1m-class robotic telescopes searching for transiting terrestrial planets around the nearest and brightest ultra-cool dwarfs. It has two main nodes, one in each hemisphere. The southern one consists of four 1-m telescopes and the northern one has the same plan but just has 1 telescope installed so far. A thermoelectrically-cooled camera with a near-IR-optimized deeply depleted 2k×2k CCD detector, which is sensitive from 350 nm (near-UV) to 950 nm (near-IR), has been installed in every site to cover 12' ×12' sky. Currently, Sloan-g'r'i'z' filters, a special exoplanet filter "I + z" and a blue-blocking filter are installed in the filter wheel in order to observe redder objects, such as mid-to-late M dwarfs (Delrez et al., 2018; Murray et al., 2020).
- ARTN (Arizona Robotic Telescope Network): This network aims to upgrade and integrate multiple small to medium size telescopes (1–3 m) to have multiple capabilities, including photometry, spectroscopy, wide-field optical imaging, rapid response and monitoring and making the sky survey. Nowadays, Steward Observatory's 1.55 m Kuiper telescope and Vatican Advanced Technology Telescope are in the first phase of the upgrade (Weiner et al., 2018).
- Skynet[9]: It is a robotic telescope network spanning four continents which can schedule targets through the control web interface. At present, it has not only optical telescopes whose size ranges from 35 to 100 cm but also a radio telescope, the Green Bank Observatory's 20-m diameter radio telescope. Hence, the multi-color optical image and the timing/mapping observations in the radio band are available to support professional astronomers and public education use in various astronomy research fields (Martin et al., 2019; Reichart, 2022).
- SMARTNet (the Small Aperture Robotic Telescope Network): It was planned to initiate a robotic telescope network worldwide with a minimum of six stations and there are three stations finished construction already. Regarding its multiple objectives, a two-telescope setup with different apertures is favoured which includes a 20 cm (FOV $3.5° \times 3.5°$) plus a 50 cm (FOV 36' ×36') telescopes to provide a deep survey of faint objects and a fast survey of bright objects, respectively. Later, the wide field telescope upgrade to a 25 cm diameter telescope (FOV $2.3° \times 2.3°$). The obtained optical images will are used to survey the Geostationary Orbits (GEO) objects and label them (Fiedler et al., 2018; Herzog et al., 2021).
- OWL-Net (Optical Wide-field patroL Network): This network is composed of 0.5-m wide field optical telescopes operated in a robotic manner and distributed in six sites in the northern hemisphere. Each telescope has installed a 4k ×4k CCD camera with Johnson (BVRI) filters and a FOV of $1.1° \times 1.1°$. Although its main objective is to monitor Korean Low earth Orbits (LEO) and GEO satellites to maintain their orbital information, the astronomical mode is also accessible for photometric images (Park et al., 2018).

In addition to these worldwide robotic telescopes networks, there are also several single-site robotic telescopes networks, such as the Ground Wide Angle Camera Network (GWAC; Han et al., 2021), and the MINiature Exoplanet Radial Velocity Array (MINERVA; Swift et al., 2015). Currently, the 2-m class robotic telescope is the biggest size of robotic telescopes, such as the Faulkes Telescopes owned by LCOGT, and most telescopes in these networks are of small size. From a construction perspective, the majority of these robotic networks were newly designed and built. However, the upgrade of existing telescopes is also an option, such as with the ARTN. In terms of research objectives, both multi-objective networks, such as LCOGT and Skynet, and single-objective networks, such as TAT and SMARTNet, have been established. Regarding time domain astronomy, an alert system is necessary for the observation of transients observation by providing a Target of Opportunity (ToO) on robotic telescopes, such as MASTER and TAROT. From the perspective of the observation mode, all of these networks have imaging capabilities, some with multi-colour capability or even polarization mode, such as MASTER, but fewer have spectroscopic

6 https://phys.au.dk/song

7 https://trt.narit.or.th/

8 https://web.tarotnet.org/

9 https://skynet.unc.edu/

observation mode, such as LCOGT and SONG. Near-IR observations are even fewer in number, only SPECULOOS. From the user perspective, some networks are used exclusively for scientific research, such as OWL-Net, but some networks are also involved with public education programs, such as MicroObservatory and Skynet. While most of these robotic systems are still expanding their deployment to additional stations.

Among them, the BOOTES network is a Spanish-lead network of medium-size (0.6m) small aperture robotic telescopes with the following capabilities.

- multi-band imaging and optical spectroscopy,
- multi-objective (GRBs, FRBs, GWs, etc.),
- ToO reactive (with long-period monitoring observation capability),
- for multiple purposes (both scientific and public science education) observations.

The BOOTES Network is composed of both wide and narrow field cameras in each individual astronomical station.

Regarding the transient automatic follow-up observations, the MASTER network shares several overlapping scientific objectives with the BOOTES project. They both can quickly react to new events' triggers and have their own unique geographical locations (with no overlap between them). With bigger diameter telescopes apertures, the BOOTES network can provide deeper limiting magnitudes whilst the tube design of the MASTER network makes it suitable for polarization measurements. In addition, the BOOTES network can provide low-resolution spectroscopy for bright objects. Currently, the MASTER network has two observatories in the southern hemisphere. With the new station BOOTES-7 deployed in 2022, the BOOTES Network has became the first Global Network of Robotic Telescopes present on all continents.

6 Conclusions and future prospects

The Burst Optical Observer and Transient Exploring System (BOOTES) is the first Global Network of Robotic Telescopes present on all continents. Nowadays, it is composed of 7 stations already installed in six countries, i.e. Spain, New Zealand, China, Mexico, South Africa and Chile. This network makes use of both wide-field and narrow-field cameras. An ultra-lightweight telescope concept proposed for the BOOTES Network has allotted to deploy six copies in the above-mentioned countries which provide the capability of fast slewing. With the complementary all-sky camera, the BOOTES network can monitor any night sky in both the northern and southern hemispheres. The BOOTES telescopes are equipped with weather information detectors and a self-designed control system based on the ASCOM platform. This allows the telescopes to automatically receive alerts and schedule observations in a matter of seconds. It is also possible to conduct remote observations through HTTP communication.

Taking full advantage of the BOOTES Global Network of Robotic Telescopes, its scientific goals are set to quickly react to high-energy transients alerts (GRBs, FRBs, GWs, etc.) and also to support scientific space missions. In regards to the BOOTES Network's observations of astronomical transients, each BOOTES telescope automatically reacts to incoming triggers by slewing to the source position using different strategies depending on the object's type. The BOOTES telescopes can react as fast as in 8 s to start gathering the observations (down to 20 mag, such as in the case of GRBs observations). With the gradual completion of the network, the detection rate of transients has increased. The rapid follow-up observations from early to late-times executed by BOOTES have been used to constrain the GRB's models. In case of FRBs, for the time being, there is still no clear optical detection related to FRBs themselves. For neutrino events, the upper limits obtained so far can set meaningful constrains on the existing models. Electromagnetic counterpart searches for GWs in the past GW detectors scientific runs on BOOTES network were executed perfectly. The optical counterpart to the milestone event, dubbed GW170817, was detected by BOOTES-5 (the only Spanish installation doing so). Even though there has not been any electromagnetic counterpart related to a GW detected in the O3, 76% of triggers have been followed up, providing a typical upper limit of 20 mag which can be used to constrain model parameters.

On top of this, the BOOTES network has contributed to public outreach through the EU FP-7 funded GLORIA project, which allows citizens to view the night sky for free through web 2.0. It also allows astronomers to gather additional scientific data. Currently, astronomy projects at the high-school level, such as ScienceIES in Spain, are giving teenagers the opportunity to experience real scientific research and obtain scientific results under the guidance of professional researchers.

The complete BOOTES network will make the transients' follow-up more efficient. The increasing number of follow-up observations regarding GRBs, FRBs, neutrino sources and the possible electromagnetic counterparts related to the fourth LIGO/Virgo scientific run (O4) in 2023 are expected to provide outstanding results. Robotic telescope networks are eager to contribute to these efforts and help shed light on the Cosmos. Additional details on the BOOTES Global Network of Robotic Telescopes will be published elsewhere (Castro-Tirado et al., 2023).

Data availability statement

The original contributions presented in the study are included in the article/**Supplementary Material**, further inquiries can be directed to the corresponding author.

Author contributions

Y-DH wrote the draft. EF-G contributed to the section of the control system. IP-G, IC-G, AC, CP, and ART developed the BOOTES control system. MC-G and AC-T gave the comments and improved the text. All co-authors maintained the system operation.

Acknowledgments

Y-DH acknowledges support under the additional funding from the RYC 2019-026465-I. MCG acknowledges support from the Ramón y Cajal Fellowship RYC 2019-026465-I. AJCT acknowledges support from the Spanish Ministry project PID 2020-118491 GB-I00 and Junta de Andalucia grant P20 010168. MCG and AJCT acknowledge financial support from the State Agency for Research of the Spanish

MCIU through the "Center of Excellence Severo Ochoa" award to the Instituto de Astrofísica de Andalucía (SEV-2017-0709). We also acknowledge to all colleagues of the scientific institutions hosting the BOOTES stations worldwide. Without them, the achievement of the BOOTES Network would have been much lower. Finally, the authors extend their thanks to the anonymous referees for their valuable comments that helped to improve the manuscript substantially.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fspas.2023.952887/full#supplementary-material

References

Aartsen, M. G., Abbasi, R., Abdou, Y., Ackermann, M., Adams, J., Aguilar, J. A., et al. (2013). First observation of PeV-energy neutrinos with IceCube. *Phys. Rev. Lett.* 111, 021103. doi:10.1103/PhysRevLett.111.021103

Abbott, B. P., Abbott, R., Abbott, T. D., Abernathy, M. R., Acernese, F., Ackley, K., et al. (2016). Localization and broadband follow-up of the gravitational-wave transient GW150914. *Astrophysical J. Lett.* 826, L13. doi:10.3847/2041-8205/826/1/L13

Abbott, B. P., Abbott, R., Abbott, T. D., Acernese, F., Ackley, K., Adams, C., et al. (2017). Multi-messenger observations of a binary neutron star merger. *Astrophysical J. Lett.* 848, L12. doi:10.3847/2041-8213/aa91c9

Akerlof, C. W., Kehoe, R. L., McKay, T. A., Rykoff, E. S., Smith, D. A., Casperson, D. E., et al. (2003). The ROTSE-III robotic telescope system, 115. *Publ. Astron. Soc. Pac.*, 132–140. doi:10.1086/345490

Andersen, M. F., Grundahl, F., Beck, A. H., and Pallé, P. (2016). "The SONG prototype: Efficiency of a robotic telescope," in Revista Mexicana de Astronomia y Astrofisica Conference Series, 54–58.

Andreoni, I., Lu, W., Smith, R. M., Masci, F. J., Bellm, E. C., Graham, M. J., et al. (2020). Zwicky transient facility constraints on the optical emission from the nearby repeating FRB 180916.J0158+65. *Astrophysical J. Lett.* 896, L2. doi:10.3847/2041-8213/ab94a5

Baruch, J. E. F. (1992). Robots in astronomy. *Vistas Astronomy* 35, 399–438. doi:10.1016/0083-6656(92)90002-N

Berger, E. (2014). Short-duration gamma-ray bursts. *Annu. Rev. Astronomy Astrophysics* 52, 43–105. doi:10.1146/annurev-astro-081913-035926

Bochenek, C. D., Ravi, V., Belov, K. V., Hallinan, G., Kocz, J., Kulkarni, S. R., et al. (2020). A fast radio burst associated with a Galactic magnetar. *Nature* 587, 59–62. doi:10.1038/s41586-020-2872-x

Boer, M., Klotz, A., Laugier, R., Richard, P., Pérez, J. C. D., Lapasset, L., et al. (2017). "Tarot: A network for space surveillance and tracking operations," in 7th European Conference on Space Debris.

Burger, B., Maffettone, P. M., Gusev, V. V., Aitchison, C. M., Bai, Y., Wang, X., et al. (2020). A mobile robotic chemist. *Nature* 583, 237–241. doi:10.1038/s41586-020-2442-2

Caballero-Garcia, M. D., Jelinek, M., Castro-Tirado, A., Hudec, R., Cunniffe, R., Rabaza, O., et al. (2014). *Initial follow-up of optical transients with COLORES using the BOOTES network*. arXiv e-prints , arXiv:1408.4370.

Caballero-García, M. D., Šimon, V., Jelínek, M., Castro-Tirado, A. J., Cwiek, A., Claret, A., et al. (2015). Early optical follow-up of the nearby active star DG CVn during its 2014 superflare. *Mon. Notices R. Astronomical Soc.* 452, 4195–4202. doi:10.1093/mnras/stv1565

Castro-Tirado, A. J., Cáceres, F. J. P., Castellón, A., Pérez del Pulgar, C. J., Cordón, A., Márquez, J., et al. (2021a). "The use of robotic telescopes in the science-IES project in Spain," in Revista Mexicana de Astronomia y Astrofisica Conference Series of Revista Mexicana de Astronomia y Astrofisica Conference Series, 151–154. doi:10.22201/ia.14052059p.2021.53.30

Castro-Tirado, A. J., Hudec, R., and Soldán, J. (1996). "The burst observer and optical transient exploring system (BOOTES)," *Gamma-ray bursts: 3rd huntsville symposium*. Editors C. Kouveliotou, M. F. Briggs, and G. J. Fishman (American Institute of Physics Conference Series), 384, 814–818. doi:10.1063/1.51596

Castro-Tirado, A. J., Fernández-García, E. J., Hu, Y.-D., Pérez-García, I., Pérez del Pulgar, C. J., Castellón, A., et al. (2023). in preparation.

Castro-Tirado, A. J., Jelínek, M., Gorosabel, J., Kubánek, P., Cunniffe, R., Guziy, S., et al. (2012). "Building the BOOTES world-wide network of robotic telescopes," in Astronomical Society of India Conference Series (Astronomical Society of India Conference Series) 7, 313–320.

Castro-Tirado, A. J., Østgaard, N., Göğüş, E., Sánchez-Gil, C., Pascual-Granado, J., Reglero, V., et al. (2021b). Very-high-frequency oscillations in the main peak of a magnetar giant flare. *Nature* 600, 621–624. doi:10.1038/s41586-021-04101-1

Castro-Tirado, A. J. (2011). Robotic astronomy and the BOOTES network of robotic telescopes. *Acta Polytech.* 51, 16. doi:10.14311/1308

Castro-Tirado, A. J. (2010). Robotic autonomous observatories: A historical perspective. *Adv. Astronomy* 2010, 1–8. doi:10.1155/2010/570489

CHIME/FRB Collaboration, Andersen, B. C., Bandura, K. M., Bhardwaj, M., Bij, A., Boyce, M. M., et al. (2020). A bright millisecond-duration radio burst from a Galactic magnetar. *Nature* 587, 54–58. doi:10.1038/s41586-020-2863-y

CHIME/FRB Collaboration, Amiri, M., Andersen, B. C., Bandura, K., Berger, S., Bhardwaj, M., et al. (2021). The first CHIME/FRB fast radio burst catalog. *Astrophysical J. Suppl. Ser.* 257, 59. doi:10.3847/1538-4365/ac33ab

Chou, D.-Y., Sun, M.-T., Fernandez Fernandez, J., Wang, L.-H., Jimenez, A., Serebryanskiy, A., et al. (2010). Taiwan automated telescope network. *Adv. Astronomy* 2010, 1–4. doi:10.1155/2010/125340

de Menezes, R., Buson, S., Garrappa, S., Gokus, A., Kadler, M., Cheung, T., et al. (2022). "Multi-Messenger observations of the Fermi-LAT blazar 4FGL J0658.6+0636 consistent with an IceCube high-energy neutrino," in 37th International Cosmic Ray Conference, 12-23 July 2021 Berlin, 955.

Delrez, L., Gillon, M., Queloz, D., Demory, B.-O., Almleaky, Y., de Wit, J., et al. (2018). "Speculoos: A network of robotic telescopes to hunt for terrestrial planets around the nearest ultracool dwarfs," in *Ground-based and airborne telescopes VII of society of photo-optical instrumentation engineers (SPIE) conference series*. Editors H. K. Marshall, and J. Spyromilio, 10700, 1070011. doi:10.1117/12.2312475

Ellery, A. (2003). Robotics: Robotics in the service of astronomy. *Astronomy Geophys.* 44, 323–325. doi:10.1046/j.1468-4004.2003.44323.x

Fiedler, H., Herzog, J., Hinze, A., Prohaska, M., Schildknecht, T., and Weigel, M. (2018). "Smartnet - evolution and results," in *IAC bremen*.

Fowler, M. J. F., Sienkiewicz, F. F., Zellem, R. T., and Dussault, M. E. (2020). *Observing transiting exoplanets with the MicroObservatory: 43 new transit light curves of the hot jupiter HAT-P-32b*. arXiv e-prints, arXiv:2007.13381.

Gehrels, N., Ramirez-Ruiz, E., and Fox, D. B. (2009). Gamma-ray bursts in the Swift era. *Annu. Rev. Astronomy Astrophysics* 47, 567–617. doi:10.1146/annurev.astro.46.060407.145147

Gorbovskoy, E. S., Lipunov, V. M., Kornilov, V. G., Belinski, A. A., Kuvshinov, D. A., Tyurina, N. V., et al. (2013). The MASTER-II network of robotic optical telescopes. First results. *Astron. Rep.* 57, 233–286. doi:10.1134/S1063772913040033

Grundahl, F., Christensen-Dalsgaard, J., Kjeldsen, H., Jørgensen, U. G., Arentoft, T., Frandsen, S., et al. (2009). "The stellar observations network group—The prototype," in *Solar-stellar dynamos as revealed by helio- and asteroseismology: Gong 2008/SOHO 21*. Editors M. Dikpati, T. Arentoft, I. González Hernández, C. Lindsey, and F. Hill (Astronomical Society of the Pacific Conference Series), 416, 579.

Han, X., Xiao, Y., Zhang, P., Turpin, D., Xin, L., Wu, C., et al. (2021). The automatic observation management system of the GWAC network. I. System architecture and workflow. *Publ. Astron. Soc. Pac.* 133, 065001. doi:10.1088/1538-3873/abfb4e

Herzog, J., Hofmann, B., Fiedler, H., Prohaska, M., and Schildknecht, T. (2021). "Software and hardware to improve a remote telescope station," in 8th European Conference on Space Debris, dlr146889.

Hiriart, D. (2014). "Continuous monitoring using BOOTES worldwide network," in Revista Mexicana de Astronomia y Astrofisica Conference Series, 87.

Hiriart, D., Valdez, J., Martínez, B., García, B., Cordova, A., Colorado, E., et al. (2016). "The BOOTES-5 telescope at san Pedro martir national astronomical observatory, Mexico," in Revista Mexicana de Astronomia y Astrofisica Conference Series, 114–117.

Hu, Y. D., Li, X. Y., Castro-Tirado, A. J., Fernandez-García, E. J., Castellón, A., Carrasco-García, I., et al. (2021). "The BOOTES network in the gravitational wave era," in Revista Mexicana de Astronomia y Astrofisica Conference Series, 75–82. doi:10.22201/ia.14052059p.2021.53.18

Hughes, S. A. (2009). Gravitational waves from merging compact binaries. *Annu. Rev. Astronomy Astrophysics* 47, 107–157. doi:10.1146/annurev-astro-082708-101711

IceCube Collaboration, Aartsen, M. G., Ackermann, M., Adams, J., Aguilar, J. A., Ahlers, M., et al. (2018). Neutrino emission from the direction of the blazar TXS 0506+056 prior to the IceCube-170922A alert. *Science* 361, 147–151. doi:10.1126/science.aat2890

Jelínek, M., Castro-Tirado, A. J., Cunniffe, R., Gorosabel, J., Vítek, S., Kubánek, P., et al. (2016). A decade of GRB follow-up by BOOTES in Spain (2003-2013). *Adv. Astronomy* 2016, 1–12. doi:10.1155/2016/1928465

Jelínek, M., Castro-Tirado, A. J., de Ugarte Postigo, A., Kubánek, P., Guziy, S., Gorosabel, J., et al. (2010). Four years of real-time GRB followup by BOOTES-1B (2005-2008). *Adv. Astronomy* 2010, 1–10. doi:10.1155/2010/432172

Jelínek, M., Gorosabel, J., Castro-Tirado, A. J., de Ugarte Postigo, A., Guziy, S., Cunniffe, R., et al. (2012). BOOTES observation of GRB080603B. *Acta Polytech.* 52, 34. doi:10.14311/1506

Jelínek, M. (2014). *Study of gamma-ray bursts with robotic telescopes*. Spain: University of Granada. Ph.D. thesis.

Kilpatrick, C. D., Burchett, J. N., Jones, D. O., Margalit, B., McMillan, R., Fong, W.-f., et al. (2021). Deep optical observations contemporaneous with emission from the periodic FRB 180916.J0158+65. *Astrophysical J. Lett.* 907, L3. doi:10.3847/2041-8213/abd560

Klebesadel, R. W., Strong, I. B., and Olson, R. A. (1973). Observations of gamma-ray bursts of cosmic origin. *Astrophysical J. Lett.* 182, L85. doi:10.1086/181225

Kouveliotou, C., Meegan, C. A., Fishman, G. J., Bhat, N. P., Briggs, M. S., Koshut, T. M., et al. (1993). Identification of two classes of gamma-ray bursts. *Astrophysical J. Lett.* 413, L101. doi:10.1086/186969

Kubánek, P., Jelínek, M., Vítek, S., de Ugarte Postigo, A., Nekola, M., French, J., et al. (2006). Status of robotics telescopes driven by RTS2 (BART, BOOTES, FRAM and Watcher). *Nuovo Cimento B Ser.* 121, 1501–1502. doi:10.1393/ncb/i2007-10295-8

Learned, J. G., and Mannheim, K. (2000). High-energy neutrino astrophysics. *Annu. Rev. Nucl. Part. Sci.* 50, 679–749. doi:10.1146/annurev.nucl.50.1.679

Lin, L., Zhang, C. F., Wang, P., Gao, H., Guan, X., Han, J. L., et al. (2020). No pulsed radio emission during a bursting phase of a Galactic magnetar. *Nature* 587, 63–65. doi:10.1038/s41586-020-2839-y

Lorimer, D. R., Bailes, M., McLaughlin, M. A., Narkevic, D. J., and Crawford, F. (2007). A bright millisecond radio burst of extragalactic origin. *Science* 318, 777–780. doi:10.1126/science.1147532

Mankiewicz, L. (2013). "Gloria - the GLObal robotic telescopes intelligent array for e-science," in *EAS publications series*. Editors A. J. Castro-Tirado, J. Gorosabel, and I. H. Park (EAS Publications Series), 61, 483–486. doi:10.1051/eas/1361079

Martin, J. R., Reichart, D. E., Dutton, D. A., Maples, M. P., Berger, T. A., Ghigo, F. D., et al. (2019). Skynet algorithm for single-dish radio mapping. I. Contaminant-Cleaning, mapping, and photometering small-scale structures. *Astrophysical J. Suppl. Ser.* 240, 12. doi:10.3847/1538-4365/aad7c1

Murray, C. A., Delrez, L., Pedersen, P. P., Queloz, D., Gillon, M., Burdanov, A., et al. (2020). Photometry and performance of SPECULOOS-South. *Mon. Notices R. Astronomical Soc.* 495, 2446–2457. doi:10.1093/mnras/staa1283

Narayan, R., Paczynski, B., and Piran, T. (1992). Gamma-ray bursts as the death throes of massive binary stars. *Astrophysical J. Lett.* 395, L83. doi:10.1086/186493

Park, J.-H., Yim, H.-S., Choi, Y.-J., Jo, J. H., Moon, H.-K., Park, Y.-S., et al. (2018). OWL-net: A global network of robotic telescopes for satellite observation. *Adv. Space Res.* 62, 152–163. doi:10.1016/j.asr.2018.04.008

Páta, P., Bernas, M., Castro-Tirado, A. J., and Hudec, R. (2001). "New version of optical transient monitor for BOOTES project," in *Gamma-ray bursts in the afterglow era*. Editors E. Costa, and F. Frontera, 412. doi:10.1007/10853853_113*J. Hjorth*

Pickles, A., Hjelstrom, A., Boroson, T., Burleson, B., Conway, P., De Vera, J., et al. (2014). "LCOGT network observatory operations," in *Observatory operations: Strategies, processes, and systems V of society of photo-optical instrumentation engineers (SPIE) conference series*. Editors A. B. Peck, C. R. Benn, and R. L. Seaman, 9149, 914912. doi:10.1117/12.2055215

Piran, T. (1999). Gamma-ray bursts and the fireball model. *Phys. Rep.* 314, 575–667. doi:10.1016/S0370-1573(98)00127-6

Rabaza, O., Jelinek, M., Castro-Tirado, A. J., Cunniffe, R., Zeman, J., Hudec, R., et al. (2013). Compact low resolution spectrograph, an imaging and long slit spectrograph for robotic telescopes. *Rev. Sci. Instrum.* 84, 114501–1145019. doi:10.1063/1.4827895

Racusin, J. L., Karpov, S. V., Sokolowski, M., Granot, J., Wu, X. F., Pal'Shin, V., et al. (2008). Broadband observations of the naked-eye γ-ray burst GRB080319B. *Nature* 455, 183–188. doi:10.1038/nature07270

Reichart, D. E. (2022). *Next-level, robotic telescope-based observing experiences to boost STEM enrollments and majors on a national scale*. arXiv e-prints , arXiv:2202.09257.

Schaeffer, R., Declais, Y., and Jullian, S. (1987). The neutrino emission of SN1987A. *Nature* 330, 142–144. doi:10.1038/330142a0

Soonthornthum, B. (2017). *The development of astronomy and emergence of astrophysics in Thailand", bookTitle="The emergence of astrophysics in Asia: Opening a new window on the Universe*. Cham: Springer International Publishing, 271–290. doi:10.1007/978-3-319-62082-4_11

Spitler, L. G., Cordes, J. M., Hessels, J. W. T., Lorimer, D. R., McLaughlin, M. A., Chatterjee, S., et al. (2014). Fast radio burst discovered in the Arecibo pulsar ALFA survey. *Astrophysical J.* 790, 101. doi:10.1088/0004-637X/790/2/101

Swift, J. J., Bottom, M., Johnson, J. A., Wright, J. T., McCrady, N., Wittenmyer, R. A., et al. (2015). Miniature exoplanet radial velocity array I: Design, commissioning, and early photometric results. *J. Astronomical Telesc. Instrum. Syst.* 1, 027002. doi:10.1117/1.JATIS.1.2.027002

Tello, J. C., de Ugarte Postigo, A., Sánchez-Ramírez, R., Jelínek, M., Gorosabel, J., Kubánek, P., et al. (2012). "Bootes 3: First two years of GRB follow-ups in New Zealand," in In Astronomical Society of India Conference Series (Astronomical Society of India Conference Series) 7, 79.

The LIGO Scientific Collaboration, the Virgo Collaboration, the KAGRA Collaboration, Abbott, R., Abbott, T. D., Acernese, F., et al. (2021). GWTC-3: Compact binary coalescences observed by LIGO and Virgo during the second part of the third observing run. arXiv e-prints arXiv:2111.03606

Thornton, D., Stappers, B., Bailes, M., Barsdell, B., Bates, S., Bhat, N. D. R., et al. (2013). A population of fast radio bursts at cosmological distances. *Science* 341, 53–56. doi:10.1126/science.1236789

Tsapras, Y., Street, R., Horne, K., Snodgrass, C., Dominik, M., Allan, A., et al. (2009). RoboNet-II: Follow-up observations of microlensing events with a robotic network of telescopes. *Astron. Nachrichten* 330, 4–11. doi:10.1002/asna.200811130

Wang, C.-J., Xiong, D.-R., and Bai, J.-M. (2019). Optical multi-color monitoring of the BL lacertae object S5 0716+714 with BOOTES-4. *Astrophysics Space Sci.* 364, 83. doi:10.1007/s10509-019-3569-1

Weber, J. (1968). Gravitational-wave-detector events. *Phys. Rev. Lett.* 20, 1307–1308. doi:10.1103/PhysRevLett.20.1307

Weiner, B. J., Sand, D., Gabor, P., Johnson, C., Swindell, S., Kubánek, P., et al. (2018). "Development of the Arizona robotic telescope network," in Observatory Operations: Strategies, Processes, and Systems VII of Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, 107042H. doi:10.1117/12.2314265

Woosley, S. E., and Bloom, J. S. (2006). The supernova–gamma-ray burst connection. *Annu. Rev. Astronomy Astrophysics* 44, 507–556. doi:10.1146/annurev.astro.43.072103.150558

Xin, L. P., Li, H. L., Wang, J., Han, X. H., Qiu, Y. L., Cai, H. B., et al. (2021). Constraints on optical emission of FAST-detected FRB 20181130B with GWAC synchronized observations. *Astrophysical J.* 922, 78. doi:10.3847/1538-4357/ac1daf

Zasche, P., Wolf, M., Uhlař, R., Cagaš, P., Juryšek, J., Mašek, M., et al. (2018). The first study of 54 new eccentric eclipsing binaries in our Galaxy. *Astronomy Astrophysics* 619, A85. doi:10.1051/0004-6361/201832793

Zhang, B. B., Zhang, B., Sun, H., Lei, W. H., Gao, H., Li, Y., et al. (2018). A peculiar low-luminosity short gamma-ray burst from a double neutron star merger progenitor. *Nat. Commun.* 9, 447. doi:10.1038/s41467-018-02847-3

Frontiers in
Astronomy and Space Sciences

Explores planetary science and extragalactic astronomy in all wavelengths

Advances the understanding of our universe - from planetary science to extragalactic astronomy, to high-energy and astroparticle physics.

Discover the latest Research Topics

See more →

Frontiers

Avenue du Tribunal-Fédéral 34
1005 Lausanne, Switzerland
frontiersin.org

Contact us

+41 (0)21 510 17 00
frontiersin.org/about/contact