Neural network methods for radiation detectors and imaging

S. Lin[1,2], S. Ning[2], H. Zhu[2], T. Zhou[3], C. L. Morris[1], S. Clayton[1], M. J. Cherukara [ID][4], R. T. Chen[2,5,6]* and Zhehui Wang[1]*

[1]Los Alamos National Laboratory, Los Alamos, NM, United States, [2]Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX, United States, [3]Center for Nanoscale Materials, Argonne National Laboratory, Lemont, IL, United States, [4]Advanced Photon Source, Argonne National Laboratory, Lemont, IL, United States, [5]Microelectronics Research Center, The University of Texas at Austin, Austin, TX, United States, [6]Omega Optics Inc, Austin, TX, United States
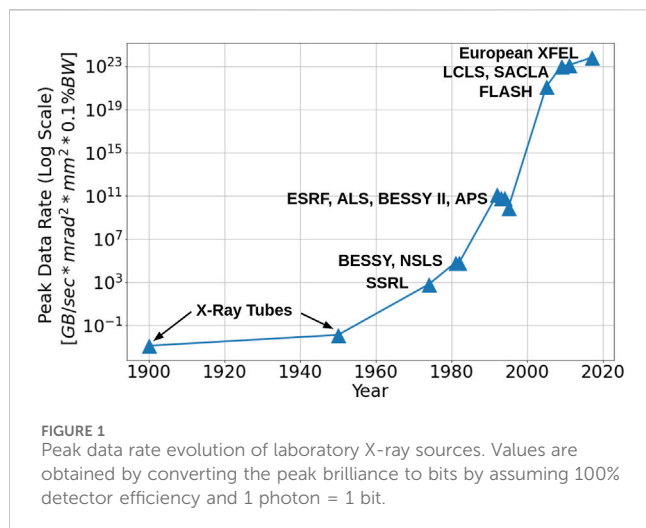
Recent advances in image data proccesing through deep learning allow for new optimization and performance-enhancement schemes for radiation detectors and imaging hardware. This enables radiation experiments, which includes photon sciences in synchrotron and X-ray free electron lasers as a subclass, through data-endowed artificial intelligence. We give an overview of data generation at photon sources, deep learning-based methods for image processing tasks, and hardware solutions for deep learning acceleration. Most existing deep learning approaches are trained offline, typically using large amounts of computational resources. However, once trained, DNNs can achieve fast inference speeds and can be deployed to edge devices. A new trend is edge computing with less energy consumption (hundreds of watts or less) and real-time analysis potential. While popularly used for edge computing, electronic-based hardware accelerators ranging from general purpose processors such as central processing units (CPUs) to application-specific integrated circuits (ASICs) are constantly reaching performance limits in latency, energy consumption, and other physical constraints. These limits give rise to next-generation analog neuromorhpic hardware platforms, such as optical neural networks (ONNs), for high parallel, low latency, and low energy computing to boost deep learning acceleration (LA-UR-23-32395).

KEYWORDS

neural networks, ONN, radiation detectors, radiographic imaging and tomography, AI, edge computing

1 Introduction

X-rays produced by synchrotrons and free electron lasers (XFELs), together with high-energy photons above 100 keV, which are often generated using high-current (kA) electron accelerators and lately high-power lasers, are widely used as radiographic imaging and tomography (RadIT) tools to examine material properties and their temporal evolution [1–3]. Spatial resolution (δ) down to atomic dimensions is possible by using diffraction-limited X-rays, $\delta \sim \lambda/2$, corresponding to Abbe's diffraction limit for X-ray wavelength λ [4,5]. The overall object size that X-rays can probe readily reaches a length (L) greater than 1 mm, which is limited by the X-ray attenuation length and is X-ray energy dependent. In room-temperature water, for example, $L = 0.19, 1.2, 5.9,$ and 14.1 cm for $1/e$-attenuation length of 10 keV, 20 keV, 100 keV, and 1 MeV X-rays, respectively. The temporal resolution has now approached a few femtoseconds by using XFELs, where an XFEL experiment can be repeated for many hours in a pump-probe configuration [6,7]. In other words, the spatial

FIGURE 1
Peak data rate evolution of laboratory X-ray sources. Values are obtained by converting the peak brilliance to bits by assuming 100% detector efficiency and 1 photon = 1 bit.

dynamic range (i.e., for 10 keV X-rays, L ~ 1 mm) is $2L/\lambda > 10^7$ and temporal dynamic range is $> 10^{18}$. Such ultra-wide-dynamic-range abilities of X-ray and photon techniques to connect elementary atomic and molecular processes, which are described by ultra fast (sub-nanosecond) quantum physics, with emergent macroscopic material properties and functions, which are usually treated classically through continuum approximations, make them extremely valuable in a wide range of applications. A few applications include medicine (i.e., new drug discovery), high-energy density battery development, and applications in materials exposed to high-temperature, high radiation, and other harsh or 'extreme' conditions. Additional applications include the optimization of chemical catalysis and the development of new superconductors and other quantum materials for information technology, accelerated computing, and artificial intelligence (AI).

The enormous spatial and temporal dynamic ranges give rise to "big data" in X-ray imaging, tomography, and photon science. Theoretically, 1 mm^3 of water contains about 5.6×10^{-5} mol of water molecules ($N = 3.3 \times 10^{19}$). If the position of every molecule were recorded, the memory size would be $N\log_2 N$ ($\log_2 N$ is the bit length for a binary data system) or 2.2×10^{21} bits. In experiments, explosive data growth in X-ray and other forms of RadIT is built upon steady progress for more than 120 years in X-ray and radiation sources, detectors, computation, and lately data science. Figure 1 shows the evolution of the peak data rate due to the increasing X-ray source brilliance over the years [8]. The fourth generation synchrotrons such as APS-U [9] and PETRA IV [10] will have a significant reduction in emittance and a brilliance increase by a factor about 10^3 over the parameters of the third generation synchrotrons such as APS and PETRA III. XFELs, which are many of orders of magnitude brighter than synchrotrons, will run at a higher repetition rate up to 1 MHz [11]. The original LCLS, in comparison, operates at 120 Hz. However, the upgraded LCLS-II greatly increased the repetition rate to 1 MHz. High-speed detectors with frame rate frequencies above 1 MHz are commercially available. The combination of high-repetition-rate experiments with a mega-pixel and larger recording system leads to high data rates, exceeding 1 TB/s (1 TB = 10^{12} bytes), as we discuss further in Sec. 2.1.

Big data not only presents a significant challenge to data handling in terms of computing speed, computing power, short- and long-term computer memory, and computer energy consumption, which all together is called "computational resources", but also offer a transformative approach to process and interpret data, i.e., machine learning (ML) and AI through data-enabled algorithms. Such algorithms, including deep learning (DL) [12,13], are distinctive from traditional physics, statistical, and other forward-model- or domain-knowledge-driven algorithms. Traditional algorithms are based on the domain knowledge, such as physics and statistics, and applicable to both small or large ensembles of data. In contrast, data-driven models may only rely on data explicitly for model training (tuning), model validation and use, with no domain knowledge required. In practice, domain knowledge always helps, partly due to the fact that some aspects of data models, such as the model architecture and other hyper-parameters, are chosen pragmatically and do not depend on the data. The amount of data required for data model training depends on the number of model parameters such as weights, activation functions, the number of nodes, *etc.* It is not uncommon that a deep neural network (DNN) may contain billions of tunable free parameters, which require a commensurate amount of data for training. Hybrid approaches to ML and AI [14,15], which merge data and domain knowledge, are increasingly popular. Hybrid models not only supplement data-driven models with domain knowledge and reduce the amount of data required for training, but also accelerate the computational speed of traditional forward models by 10 to more than 100 times by bypassing some detailed and time-consuming computations [16–18].

We may differentiate two approaches to ML and AI by the computational resources involved and how the resources are distributed. In the *centralized* approach, data are collected from distributed locations or different data acquisition instruments through the internet. The data are then stored in a data center, and processed by high-performance computers or mainframes. Cloud computing and data centers are now widely used to process 'big data' in industry, healthcare, and research institutions. However, using cloud computing to process data generated at the network edge is not always efficient. One limiting factor is the limited network bandwidth for data transportation due to increasing data generation rates. For example, in 2017, CERN had to install a third 100 Gigabit per second fiber optic line to increase their network capacity and bandwidth [19]. Other factors include the scalability and privacy issues of data transmission to the cloud [20]. Through the cloud computing and data center approach, data generation and data processing tasks can be separated, which can mitigate the computation and data processing burden on people who generate data. In the *distributed* or *edge* approach, ML and AI, together with the computing hardware, are deployed at the individual device or instrument level. Distributed computing now pairs with distributed data. Through an internet of ML/AI-enhanced instruments, each ML/AI-enhanced instrument can be optimized for a specific purpose such as data reduction and real-time data processing. Shown later in Table 1, detection cameras used at various synchrotron and XFEL facilities can generate data at a rate of >1 GB/s and over >12.5 GB/s for state-of-the-art cameras. This results in high costs of memory storage as well as high energy costs

TABLE 1 A comparison of different camera data rates and specifications for individual integration modules for each detector. Additional details and examples may be found in [1]. Note that this table tabulates select cameras to illustrate the data rates and their uses in light sources or X-ray measurements. The state of the art is >10 Gpixel/s (>12.5 GB/s assuming 10-bit data) in continuous mode imaging. The burst mode imaging is > 1 Tpixel/s (>1250 GB/s assuming 10-bit data) [218].

Detector (camera)	Facility (particle/ photon)	Det. Mode (Direct/ inDirect)	Array format (voxel size, μm^3/pixel size, μm^2)	Frame-rate (fps/Hz)	Data bits	Data rate (GB/s)
AGIPD [39]	Eu-XFEL	D	512×128^a	16 k/6.5 M^b	14	1.85
	(12.4 keV)		($200^2 \times 500$)			
CS-PAD [219]	LCLS	D	194×370^c	120	14	0.02
	(8.3 keV)		($110^2 \times 500$)			
DSSC [220]	Eu-FXEL	D	1024×1024	1–5 M	8	1050–5240
	(0.5–20 keV)		($204 \times 236 \times 450$)			
ePix100 [221]	LCLS	D	384×352^d	120	14	0.03
	(8.3 keV)		($50^2 \times 500$)	(≤ 240)		
ePix10k	LCLS	D	384×352^e	120	14	0.03
	(8.3 keV)		($100^2 \times 500$)	($\leq 10^3$)		
EIGER2 (Dectris)	APS & others	D	1028×512^f	2.25 k	16	2.37
			($75^2 \times 450$)	(4.5 k)	(8)	
HEXITEC [222]	DIAMOND	D	80^2	6.3–8.9 k	14	0.07–0.1
	(2–200 keV)		($250^2 \times 1000^g$)			
Icarus [223]	NIF, Z	D	1024×512	≥ 250 M^h	10	163,840
	(0.7–10 keV)		($25^2 \times 25$)			
(Advanced hCMOS Sys.) JUNGFRAU [224]	SwissFEL/SLS	D	1024×512^i	2.2 k	16	2.31
	(0.25 ^{j}-12 keV)					
MM-PAD [225]	CHESS	D	128^2	10 k/100 M^k	14	0.3–2867
	(> 20 keV)^l		($150^2 \times 500$)			
SOPHIAS	SACLA	D	891×2157	60	12	0.17
			($30^2 \times 500$)			
HPV-X2 [226]	APS & others	inD	400×250	7.8 k/5 M^m	10	0.98–625
(Shimadzu)	(10–40 keV)		(32^2)			
Kraken [227]	NNSS	inD	800×800	20 M^n	12	19,200
			(30^2)			
MX170-HS	LCLS	inD	3840^2	2.5^o	16	0.07
(Rayonix)	(8–12 keV)		(44^2)			
PI MAX 4	APS	inD	1024^2	26^p	16	0.05
(Teledyne)	(10–40 keV)		(12.8^2)			

^{a}AGIPD is deployed as mega-pixel/voxel cameras through tiling.
^{b}Burst mode for 352 stored frames.
^{c}CS-PAD is deployed as tiled 2, 8, and 32 modules with up to 2.3 M voxels.
^{d}ePix100 is deployed as tiled 4 modules with about 0.5 M voxels.
^{e}ePix10K replaces CS-PAD, and is deployed as a single, or tiled 16 modules with about 2.2 M voxels.
^{f}Eiger2 is deployed as a single, or tiled modules with more than 10 M voxels.
^{g}Also 2 mm CdZnTe.
^{h}In burst mode for 4 frames.
^{i}Array size of individual modules. Multiple modules can be tiled to create larger detector configurations [228]. For example, the JUNGFRAU 4M consists of eight modules.
^{j}Can resolve single photons down to 800 eV or lower by combining the readout chip with LGAD sensors [229].
^{k}In burst mode for 8 frames.
^{l}When using 750 μm CdTe as sensor.
^{m}In burst mode for 128 stored frames; or 10 M Hz frame rate and 256 stored frames possible by reducing the number of pixels by half.
^{n}In burst mode for 8 frames. Read noise 157 e^-, Full Well 4.0 × 105 e^-. Buttable to larger array 2 × 2.
^{o}Higher frame rate can be obtained through pixel binning, at 10 × 10 binning, the frame rate increases to 120 Hz.
^{p}Higher frame rate can be obtained through pixel binning, at 4 × 4 binning, the frame rate increases to 95 Hz.

for data transmission and memory access during data processing. The large volumes, varieties, and generation rate of X-ray data motivate automated processing and reduction in light sources, such as synchrotrons and XFELs, to reduce the memory requirement, minimize latency related to data transmission and processing, and lower energy and power consumption. Later in the paper, Section 3.1 will discuss more details.

We will give an overview of DL methods for real-time radiation image analysis as well as hardware solutions for DL acceleration at the edge. We note that while not all scientific applications may require real-time image analysis, it is possible to offload some computing and preprocessing steps to an edge device. The edge device can preprocess the acquired data in real-time before sending the processed data to upstream processing centers for heavier computations. This paper is organized as follows. In Section 2, we discuss different radiation detectors and imaging devices, the resulting big data generation at photon sources, and the motivations for edge computing and DL. In Section 3, we present an overview of popular neural network architectures and several image processing tasks that have potential to be performed on edge devices. In addition, we discuss examples of DL-based methods for each. In Section 4, we present on overview of hardware solutions for DL acceleration and recent works that have applied them for computing at the edge. Lastly, Section 5 concludes this paper.

2 Experimental data generation at photon sources

Data science at light sources is centered around scientific data generation and processing. Scientific data at synchrotrons and XFEL sources consist of experimental data, simulation and synthetic data, and meta data, such as detector calibration data, material properties of objects and sensors, and point spread functions of the detectors. Methods (imaging modalities) and detectors to collect experimental data are driven by the light sources, which continue to improve in source brightness, repetition rate, source coherence, photon energy, and spectral tunability. Computing hardware and algorithms are used to process experimental data and for data visualization. Computing hardware and algorithms are also used to simulate the experiments and produce synthetic data as close to the experimental data as possible for experimental data interpretation. Diversity of the materials to be integrated and imaged, together with the photon source and detector improvement have demanded continued improvements in computing hardware and algorithms towards real-time data processing, reductions in data transmission over long distances, and reducing data storage volumes.
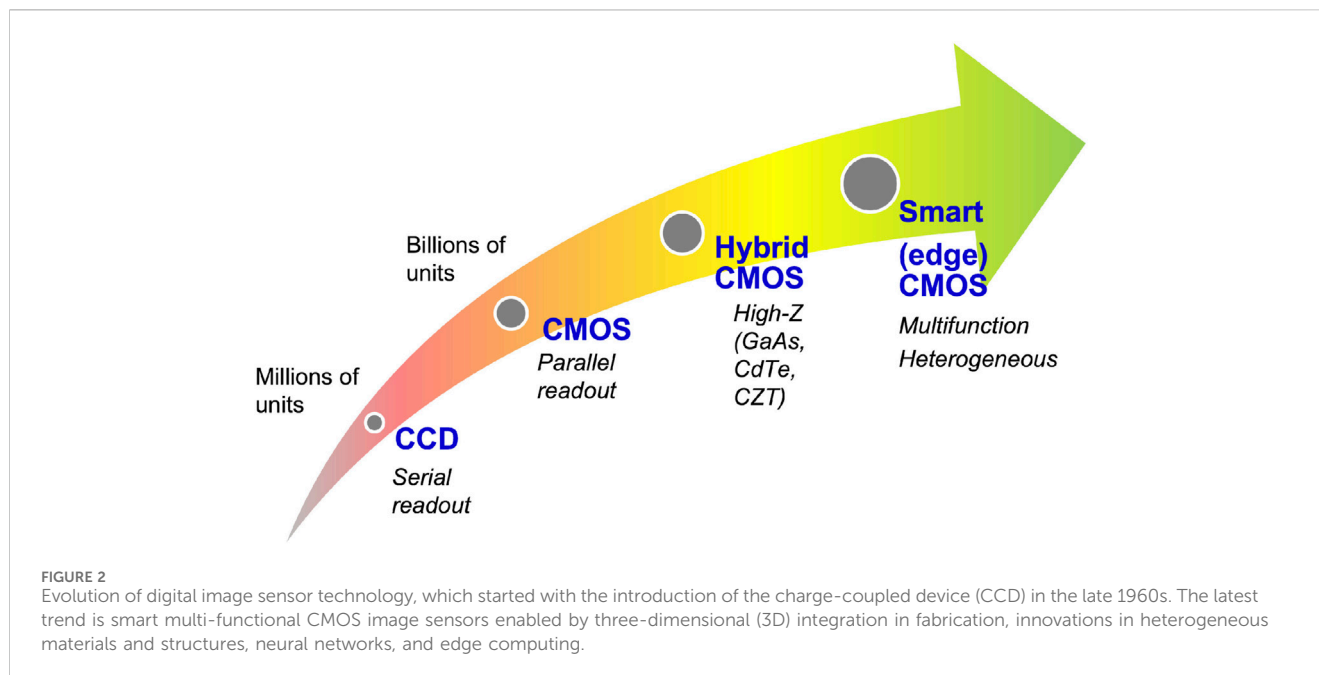
2.1 Radiation detectors and imaging for photon science

Complementary metal-oxide semiconductor (CMOS) pixelated detectors, including hybrid CMOS, are now widely used for X-ray photon science, replacing charge-coupled devices (CCDs) as the primary digital imaging technology, see Figure 2. CMOS technology is rapidly catching up to CCD cameras, with recent developments

such as Sony's STARVIS which can offer better sensitivity than traditional CCD sensors [21]. In addition, CMOS sensors are much cheaper than CCD sensors, making them more cost efficient while achieving matching performance. The latest trend is smart CMOS technology to enable edge computing and neural networks on CMOS sensors [22,23]; see Section 2.3 for more details.

CMOS sensors are used in many state-of-the-art radiation applications. For example, CMOS-based back-thinned monolithic active pixel sensors (MAPS) are the state-of-the-art detectors used for cryo-electron microscopy applications. MAPS detectors are CMOS sensors that combine the photodetectors and readout electronics on the same silicon layer, while backthinning reduces the electron scattering within pixels. MAPS detectors are also being developed for high-energy physics [24], cryogenic electron microscopy (cryo-EM), cryo-ptychography, integrated differential phase contrast (iDPC), and liquid cell imaging applications [25]. Meanwhile, hybrid CMOS detectors such as the AGIPD, ePix, and MM-PAD (see Table 1 for more detectors) are popularly used at facilities for photon science applications. Hybrid detectors are composed of a sensor array and pixel electronics readout layer that are interconnected through bump bonding, while the sensor frontend can be fabricated using different semiconductor materials. The thickness and material properties of the sensor array is dependent on the active absorbing layer design requirements and given X-ray energy to obtain high quantum efficiency. For example, high-Z sensors use materials with high atomic numbers such as Gallium Arsenide (GaAs), Cadmium Telluride (CdTE) and Cadmium Zinc Telluride (CZT) [26]. The hybrid design architecture allows for independent optimization of the quantum efficiency of the sensor array and pixel electronics functionality to meet imaging and measurement performance requirements [27]. Currently, hybrid CMOS detectors are the most widely used image sensors for high energy physics experiments [24]. Another family of image sensor is called the low-gain avalanche detector (LGAD), a silicon sensor fabricated on thin substrates to deliver fast signal pulses to achieve enhanced time resolution [28], as well as to increase the X-ray signal amplitudes and the signal-to-noise ratio to achieve single photon resolution [29]. As a result, LGADs are popularly used in experiments that require fast time resolution and good spatial resolution such as 4D tracking [30] and for soft X-ray applications in low energy diffraction, spectro-microscopy and imaging experiments such as the resonant inelastic X-ray scattering experiments [29]. In summary, radiation pixel detectors aim to capture incident photons and convert the accumulated charges in the pixel into an output image. We also mention that CMOS image sensors, including hybrid CMOS, may also be extended to neutron imaging by converting incident neutrons to visible photons through neutron capture reactions [31].

The particle nature of photons motivates digitized detectors for photon counting. Hybrid CMOS detectors are one of the most popular detectors that use the photon counting mode of operation, where individual photons are detected by tuning the discriminator threshold and the energy value of each incident photon is recorded as electronic signals. However, several factors complicate photon counting implementation in high-luminosity X-ray sources. The intensity of the sources can be too high to count individual photons one by one. The amount of X-ray photon-induced charge in CMOS detectors, which is the basis of X-ray photon counting, is not

FIGURE 2
Evolution of digital image sensor technology, which started with the introduction of the charge-coupled device (CCD) in the late 1960s. The latest trend is smart multi-functional CMOS image sensors enabled by three-dimensional (3D) integration in fabrication, innovations in heterogeneous materials and structures, neural networks, and edge computing.

constant for the same X-ray energy. Furthermore, the detectors suffer from a charge sharing issue when a photon interacts on the border between neighboring pixels. The source energy is not monochromatic, especially in imaging applications. Inelastic scattering of mono-chromatic X-rays can result in a broad distribution of X-ray photon energies after scattering by the object. When an optical camera is used together with a X-ray scintillator, the energy resolution of individual X-rays based on the photon detection is worse than direct detection when the X-ray directly deposits its energy in a silicon photo-diode. See Table 1 for a comparison of different direct and indirect detection cameras and their data rates. Note that Table 1 tabulates the specifications of select cameras to illustrate their data rates as well as their uses in light sources or X-ray measurements. To overcome the issue of the too high photon flux rate, hybrid detectors are developed to operate under a charge integration mode, where the signal intensity is obtained by integrating over the exposure time. The current generation of hybrid CMOS detectors are capable of different modes of operation (i.e., photon counting and charge integration) for direct photon detection [32].

2.2 Imaging modalities

X-ray microscopy uses X-ray lenses, zone plates, mirrors and other optics to modulate the X-ray field to form images [33]. As the X-ray intensities generated by synchrotrons and XFELs continue to increase, the advances in computational imaging modalities and lens-less X-ray modalities are increasingly used in synchrotrons and XFELs. In some cases, lens-less modalities may be preferred to avoid damages to X-ray lenses and mirrors. Lens-less modalities may also avoid aberration, diffraction due to imperfect X-ray lens, defects in zone plates, and other optics. The simplest lens-less X-ray imaging setup is radiography or projection imaging, pioneered by Röntgen. Röntgen's lens-less radiographic imaging modality directly

measures attenuated X-ray intensity due to absorption. Synchrotrons and XFELs also allow a growing number of phase contrast imaging, see Ref. [1] and references therein. Other modalities include in-line holography [34] and coherent diffractive imaging [35]. Additional phase and intensity modulation using pinholes, coded apertures, and kinoforms are also possible. Combinatorial X-ray modalities have also been introduced. For example, X-ray ptychography microscopy combines raster scanning X-ray microscopy with coherent diffraction imaging [36]. Compton scattering, usually ignored in the synchrotron and XFEL setting, may offer some additional information about the samples and potentially reduce the dose required [37]. The versatility of modalities requires different off-line and real-time data processing techniques. Background reduction is a common issue for all X-ray modalities. Real-time data processing, including energy-resolving detection, is highly desirable to distinguish different sources of X-rays since the detector pixel may simultaneously collect X-ray photons from different sources of X-ray attenuation and scattering.

2.3 Real time in-pixel data-processing

When an X-ray photon is detected directly or indirectly through the use of a scintillator, charge-hole pairs are created through photo-to-electric conversion, or the photoelectric effect, within pixels of a camera or a pixelated array. CCD cameras, CMOS cameras, and LGAD arrays are now available for synchrotron and XFEL applications. Unlike a CCD camera, a CMOS image sensor collects charge and stores it in capacitors in pixels in parallel. Parallel charge collection and capacitor voltage digitization, which turns analog voltage signals into digitized signals, allow CMOS image sensors to operate at a much higher frame rate than CCDs. Charge and voltage amplification, in LGADs and sometimes in CMOS image sensors, are also used to improve

signal-to-noise ratio. Any source of charge or voltage modulation not related to the photoelectric effect is a potential source of noise. The photoelectric effect itself can lead to so-called Poisson noise due to the probabilistic process of photo-to-electric conversion. Other sources of noise include thermal noise or dark current, salt'n'pepper noise (due to charge migration in and out of pixel defects and traps), and readout noise.

Automated real-time in-pixel signal and data processing are therefore required in CMOS and other pixelated array sensors for noise rejection and noise reduction for charge and voltage amplification controls, and for charge sharing corrections. Figure 3 illustrates a generic approach on in-pixel neural network processing for optimized and real-time data processing. Common approaches process the data by transmitting it to a separate processor and storing the data in memory. However, the data transmission and memory access actions are known to be among the most power hungry in imaging systems [38]. As a result, it is desirable to optimize the end-to-end processes of sensing, data transmission, and processing tasks. One solution is to utilize in-pixel processing to directly extract features of the input pixels which can significantly reduce system bandwidth and power consumption of data transmission, memory management, and downstream data processing. In recent years, a number of works have been proposed to implement image sensors with in-pixel neural network processing; see [22,23] and references therein. This motivates real-time image processing for image sensors for various image processing tasks including noise removal. If uncorrected, noise can corrupt the image information and make it hard for post processing or misleading for data interpretation. Charge and voltage amplification may lead to nonlinear distortion between the X-ray flux and voltage signal. When the X-ray flux is too high, the so-called plasma effect may also need correction. Charge-sharing happens when an X-ray photon arrives at a pixel border and the electron-hole pairs created are spread across multiple neighboring pixels.

By using transistor circuits, correlated double sampling (CDS) is an extremely successful example in noise reduction. Adaptive gain control circuits have been implemented in the AGIPD high-speed camera [39,40]. While real-time pixel-level signal processing by novel transistor circuits is important, there is also room for novel data-processing approaches that do not require hardware modifications to the pixels. As a recent example [41], a physics-informed neural network was demonstrated to improve spatial resolution of neutron imaging. Other novel applications of neural networks and their integration with hardware, see Figure 3, may offer new possibilities in noise reduction and image corrections. Integrated hardware and software (neutral networks are emphasized here) approaches for optimal performance also need to take into account of the complexity of the workflow [42–44], or computational cost, power consumptions, constrained by the frame rate and other metrics. For example, the computational cost of an $n \times n$ matrix is $O(n^3)$ [45].

3 Deep learning for image processing

In recent years, deep learning (DL) has contributed significantly to the progress in computer vision, especially in different areas of image processing tasks including but not limited to image denoising, segmentation, super-resolution, and classification. DL is a sub-field of ML and AI that utilize neural networks (NNs) and their superior nonlinear approximation capabilities to learn underlying structures and patterns within high-dimensional data [12]. In other words, DL aims to learn multiple levels of representation, corresponding to a hierarchy of features or concepts, where higher-level features are defined from lower-level ones and lower-level ones can help build up higher-level features.

For DL algorithms to extract underlying features and to obtain accurate predictions, it is important to understand the workflow of the DL process. In general, the DL process can be broken into several stages: i) data acquisition, ii) data preprocessing, iii) model training, testing, and evaluation, iv) model deployment and monitoring [46,47]. The first step to ML and DL problems is to collect large amounts of data from sources including but not limited to sensors, cameras, and databases. Next, the collected data needs to be preprocessed into useful features as inputs into the DL model. At a high level, the preprocessing step aims to prepare the raw data (e.g., data cleaning, outlier removal, data normalization, etc.) and to allow data analysts to preform data exploration (i.e., identifying data structure, relevance, type, and suitability). The preprocessed data is split for model training, testing, and evaluation. The appropriate DL training algorithm, model, and ML problem are dependent on the nature of the application. The model is trained on the training dataset to tune the model hyperparameters and is evaluated using unseen data, also known as the test dataset. This process is reiterated until a desired accuracy performance or stopping criteria has been achieved. Last, the trained model is deployed and monitored for further retraining and redeployment. See [46,48,49] for comprehensive details on the basics of DL.

3.1 Centralized and decentralized learning

Recall that ML and AI, and thus DL, can be differentiated into two approaches, namely, centralized and decentralized approaches. In the centralized approach, the data collected from network edge detectors are transmitted to and stored in a data center and then processed by high-performance computers. Very large traditional, ML, or hybrid algorithms can be deployed in the data center, which also requires correspondingly large memory, energy and power consumption. Estimated global data center electricity consumption in 2022 was 240–340 TWh [50], or around 1%–1.3% of global final electricity demand from data centers and data transmission networks [51]. To put this value in a better perspective, it is estimated that Bitcoin alone consumes around 125 TWh per year [52] and that the combination of Bitcoin and Ethereum consumed around 190 TWh (0.81% of the world energy consumption) in 2021 [53]. Furthermore, centralized approaches and large ML models are commonly executed by a large team of people. A Meta AI research team recently introduced the model called Segment Anything Model (SAM) and a dataset of more than 1 billion masks on 11 Million images [54]. Nvidia unveiled Project Clara at its recent GTC conference, showing early results using DL post-processing to dramatically enhance existing, often grainy and indistinct echocardiograms (sonograms of the heart). Clara motivates acceleration in research being done on several fronts that exploits explosive growth in DL computational capability to
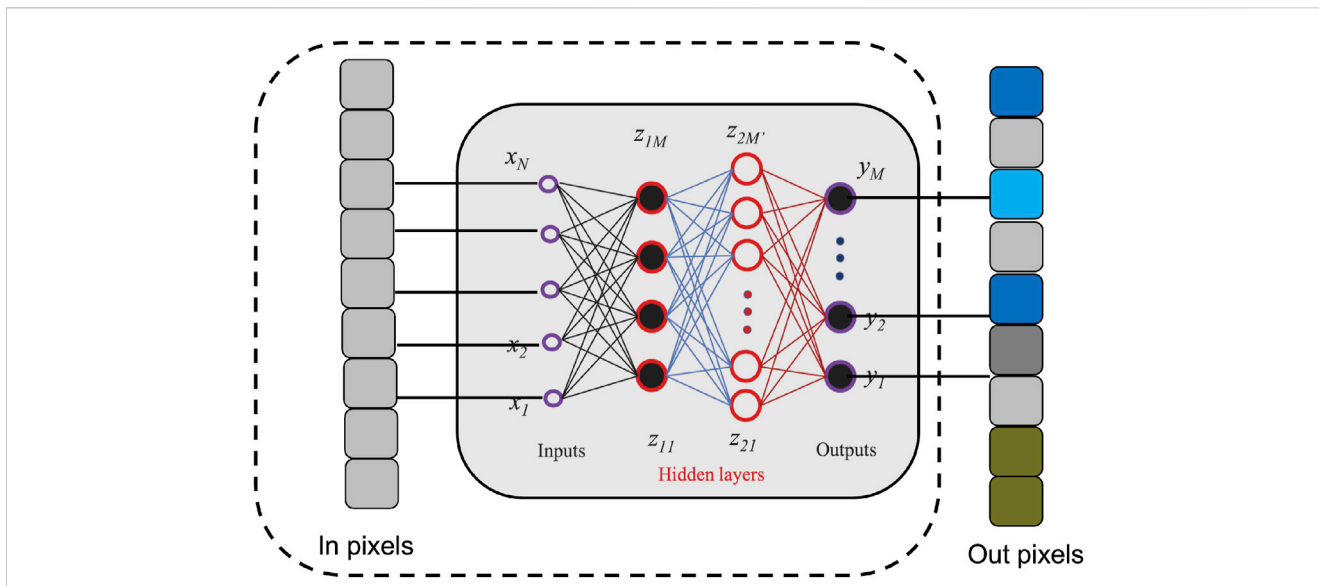
FIGURE 3
A generic illustration of in-pixel neural network processing for optimized and real-time end-to-end data processing and reductions. The neural network is directly implemented on the imaging sensor. For this specific example, the network illustrated is a fully connected neural network (FCNN). The network takes in the sensor pixel values as inputs x (in pixels) then feeds the input into hidden layers z with the number of neurons per layer denoted by M. The processed pixels (out pixels) is the output of the neural network denoted by y.

perform analysis that was previously impossible or far too costly. One technique is called 3D volumetric segmentation that can accurately measure the size of organs, tumors or fluids, such as the volume of blood flowing through arteries and the heart. Nvidia claims that a recent implementation, an algorithm called V-Net, "would've needed a computer that cost $10 million and consumed 500 kW of power [15 years ago]. Today, it can run on a few Tesla V100 GPUs" [55,56]. This claim accentuates the rapid advancements made in the hardware industry to accommodate DL computational requirements. For example, a work by [57] implemented and trained V-Net for 48 h on a workstation equipped with an NVIDIA GTX 1080 with 8 GB of video memory.

However, data processing using cloud computing and data centers are inefficient due to factors including limited network latency, scalability, and privacy [20]. To address these challenges, edge computing, or the distributed approach, offloads computing resources to the edge devices to improve network latency, to enable real-time services, and to address data privacy challenges by directly analyzing data generated by the source. In addition, edge computing will help reduce the high costs of memory storage as well as high energy costs for data transmission and memory access during data processing. For example, CERN's Large Hadron Collider (LHC) and non-LHC experiments generate over 100 petabytes of data each year and CERN's main data center had an energy consumption of about 37 GWh over the year of 2021 [58]. Assuming that the average cost of electricity is $0.15/kWh, then the cost of using 37 GWh is $5.55 million. CERN's new data center in Prévessin aims to have a power usage effectiveness (PUE) below 1.1 (ideal PUE is 1.0), and in future data centers, CERN aims to implement ML based approaches to key computing tasks to help reduce the amount of computing resources and energy consumption [58].

There are some other successes using ML and AI in areas such as HEP experiments (i.e., Higgs boson discovery) and electron microscopy. The discovery of the Higgs boson was a major challenge in HEP and can be setup as a classification problem. Many ML methods such as decision trees, logistic regression, and DL algorithms have been applied to solve the signal separation problem [59,60]. Meanwhile, ML and AI in electron microscopy are proposed to enable autonomous experimentation. Specifically, the automation of routine operations including but not limited to probe conditioning, guided exploration of large images, optimized spectroscopy measurements, and time-intensive and repetitive operations [61]. Edge ML and edge AI have already attracted a lot of attention in medicine. The fusion of DL and medical images creates dramatic improvements [56]. The concept is similar to techniques like high-dynamic range (HDR) photography, digital remastering of recordings or even film colorization in that one or more original sources of data are post-processed and enhanced to bring out additional detail, remove noise or improve aesthetics.

3.2 Neural network architectures

This section provides an overview of different popular deep neural network (DNN) architectures used for image processing tasks. These widely used architectures include but are not limited to convolutional neural networks (CNNs), long short-term memory (LSTM), encoder-decoder networks, and generative adversarial networks (GANs). Due to space limitations, other DNN architectures such as transformers [62], restricted boltzamann machines [63], and extreme learning [64] will not be covered here.

3.2.1 Convolutional neural networks (CNNs)

CNNs are one of the most widely used architectures in DL, especially for image processing tasks, due to their inherent spatial invariance property. The built-in convolutional layers allow the

FIGURE 4
Basic neural network architectures for **(A)** CNNs, **(B)** LSTMs, **(C)** encoder-decoders, and **(D)** GANs.
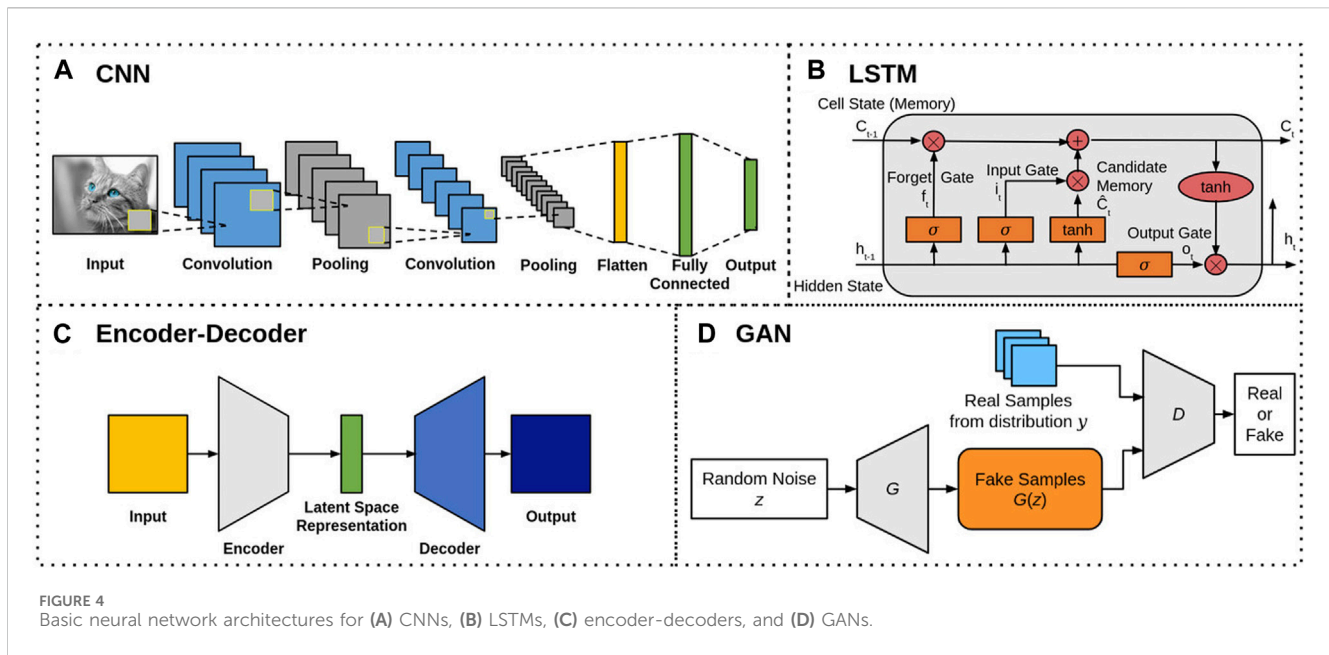
network to naturally reduce the high dimensionality of the input data, i.e., images, without information loss. Figure 4A shows the basic architecture of CNNs, which usually consists of 3 types of layers: i) convolutional layers, ii) pooling layers, and iii) fully connected layers. The convolutional layer uses various kernels to convolve the entire input image, including intermediate feature maps, and generate new feature maps. There are 3 major advantages of the convolutional operation [65]: i) the number of parameters is reduced by using weight sharing mechanisms, ii) the correlation among neighboring pixels are easily learned through local connectivity, and iii) the location of objects are fixed due to spatial invariance. Generally following a convolutional layer, the pooling layer is used to further reduce the dimensions of feature maps and network parameters. The average pooling and max pooling methods are commonly used, and their theoretical performances have been evaluated by [66,67], where max pooling is shown to achieve faster convergence and improved CNN performance. Lastly, the fully connected layer follows the last pooling or convolutional layer to convert the 2D feature maps into a 1D vector for additional feature mapping, i.e., labels. A few of the well known CNN models are the AlexNet [68], VGG [69], GoogLeNet [70], and ResNet [71], where all the models were top 3 finishers in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). Discussed later in Section 3.3, CNNs are popularly used in many image processing tasks such as image restoration (e.g., denoising, deblurring, and super resolution), segmentation, classification, and 3D reconstruction. A few examples from photon sciences include but are not limited to denoising synchrotron computed tomography images, deblurring neutron images, segmentation of inertial confinement fusion radiographs, and 3D reconstruction of coherent diffraction imaging.

3.2.2 Long short-term memory (LSTM)

LSTMs [72] are a special type of recurrent neural network (RNN) that is commonly used to process sequential datasets, such as audio recordings, videos, and time-series data. Figure 4B

shows the basic structure of a LSTM block, which consists of 3 gates (the forget gate f_t, the input gate i_t, and output gate o_t), as well as the candidate memory (new information) $\tilde{C}_t$, that regulate the stored memory and information flow within the block. Note that the subscript t denotes the variable state at time t. In summary, the forget gate decides what information to remove from the cell state, the input gate decides what information to update in the cell state by selecting the candidate memory values, and the output gate decides what information to output in the current cell state. The multiple gate architecture of LSTMs is specifically designed to capture long-term dependencies in the data as well as to avoid the vanishing gradient problem of vanilla RNNs [72]. The vanishing gradient problem in RNNs results from the backpropagation of gradients through time, which can result in very small gradient values and short-term memory behaviors. Meanwhile, the gating mechanisms of LSTMs control the flow of gradients through time during backpropagation, and thus effectively addresses the vanishing gradient problem and allows the network to learn long-term memory behaviors. Other LSTM architectures are derived from the basic architecture in Figure 4B such as LSTM without a forget gate, LSTM with peephole connections, the gated recurrent unit (GRU), and other variants [72]. Due to their ability to process sequential data, LSTMs can be used to process time-series experimental data such as videos, where video-based processing techniques can be applied.

3.2.3 Encoder-decoders

Encoder-decoder neural networks, also known as sequence-to-sequence networks, are a type of network that learns to map the input domain to a desired output domain [13]. As shown in Figure 4C, the network consists of two main components: an encoder network which uses an encoder function $h = f(x)$ to compress the input x into a latent-space representation h, and a decoder network $y = g(h)$ that produces a reconstruction y from h. The latent-space representation h prioritizes learning the important aspects of the input x which are useful in reconstructing the output y.

A special case of encoder-decoder models, autoencoders are networks in which the input and output domains are the same. These networks are popularly used in DL applications involving sequence-to-sequence modeling such as natural language processing [73], image captioning [74], and speech recognition [75]. In image processing, encoder-decoder networks are popularly used for image denoising, segmentation, compression, and 3D reconstruction. For example, one popular encoder-decoder model for image segmentation is U-Net [76]. Discussed later in Section 3.3, U-Net is used for image segmentation of inertial confinement fusion images and modified versions of the U-Net architecture are used in many works for image processing tasks. A few examples include but are not limited to the denoising and super resolution of synchrotron and X-ray computed tomography images.

3.2.4 Generative adversarial networks (GANs)

GANs [77] are increasingly popular DL frameworks for generative AI models. Classical GANs consist of 2 different networks, a generator and a discriminator, as shown in Figure 4D. The generator network G aims to generate data $G(z)$ that is indistinguishable from the real data by learning a mapping from an input noise distribution z to a target distribution y of the real data. Meanwhile, the discriminator network D takes as input the real and generated data, and aims to correctly classify them as "real" or "fake" (generated). The GAN learning objective takes on a game-theoretic approach as a two player minimax game between G and D. Let $\mathcal{L}$ denote the loss function and the GAN objective as $\min_G \max_D \mathcal{L}(G, D)$. Intuitively, D aims to minimize its own classification error, which maximizes $\mathcal{L}(G, D)$. Meanwhile, G aims to maximize the classification error of D, which minimizes $\mathcal{L}(G, D)$. This adversarial loss function allows both models to be trained simultaneously and in competition with each other. Other GAN architectures are derived from the basic architecture in Figure 4D such as conditional GANs, GANs with inference models, and adversarial autoencoders [78]. Similar to CNNs, GANs are popularly used in many image processing tasks such as image restoration, compression, and 3D reconstruction. For example, a GAN-based image denoising method was proposed to denoise low-dose X-ray computed tomography images. In addition, GANs can be used to generate synthetic data similar to experimental data. A recent work [79] uses a GAN-based model to generate synthetic inertial confinement fusion radiographs.

3.3 Image processing techniques

This section provides an overview of several image processing tasks that have potential to be performed on edge devices. In addition, this section surveys different works that have applied the DL-based image processing techniques to radiographic image processing.

3.3.1 Restoration

Image restoration is the process of adjusting the quality of digital images such that the enhanced image can facilitate further image analysis. Common enhancement operations include histogram-based equalization, brightness, and contrast adjustment. However, these operations are very elemental and advanced operations are

necessary to further improve the perceptual quality. These advanced operations include image denoising, deblurring, and super-resolution (SR); see [80–83] for examples of images before and after processing.

3.3.1.1 Denoising

One of the fundamental challenges in image processing, image denoising aims to estimate the ground-truth image by suppressing internal and external noise factors such as sensor and environmental noise, as discussed in Section 2.3. Sources of noise include but are not limited to Poisson noise due to photo-electric conversion, camera thermal noise or dark current, salt'n'pepper noise, camera readout noise, and shot noise for low-dose X-ray imaging conditions. Conventional methods including but not limited to adaptive nonlinear filters, Markov random field (MRF), and weighed nuclear norm minimization (WNNM), have achieved good performance in image denoising [84], however, they suffer from several drawbacks [85]. Two major drawbacks are the need to manually set parameters as the proposed methods are non-convex and the high computational cost for the optimization problem for the test phase. To overcome these challenges, DL methods are applied for image denoising problems to learn the underlying noise distribution. Various neural network architectures, such as CNNs, encoder-decoders, and GANs, have been proposed for image denoising in recent years; see [84] for details.

An example application that uses image denoising is in X-ray computed tomography (CT). X-ray CT imaging is a common noninvasive imaging technique that allows for reconstructing the internal structure of objects by using 3D reconstruction from 2D projection images; see Section 3.3.6 on 3D reconstruction. The spatial resolution of XFEL-based and synchrotron-based X-ray CT images can range from tens of microns to a few nanometers, while higher resolutions can be obtained by using higher radiation doses. However, some experiments may require short exposure times or low radiation dosage to avoid damaging the sample. The low-dose image conditions results in noisy 2D projection images, which in turn impacts the quality of the 3D reconstructed image. To address this issue [86], developed a GAN-based image denoising method called TomoGAN. TomoGAN is a conditional GAN model where the generator G conditionally uses the noisy reconstruction as input and outputs enhanced (denoised) reconstructions. Furthermore, the generator network architecture adopts a modified U-Net [76] architecture, popularly used for image segmentation. Meanwhile, the discriminator D is trained to classify reconstructions of the enhanced reconstructions and reconstructions of normal dose projections [86]. Evaluates the effectiveness of TomoGAN on two experimental (shale sample) datasets. TomoGAN outperforms conventional methods in noise reduction and reports a higher structural similarity (SSIM) value. In addition, TomoGAN is demonstrated to be robust to images with dynamic features from faster experiments, e.g., collecting fewer projections and/or using shorter exposure times.

Denoising has also been applied to synchrotron radiation CT (SR-CT) in a recent work by [87], which developed a CNN-based image denoising method called Sparse2Noise. Similar to the previous work for TomoGAN, this work presents a low-dose imaging strategy and utilizes paired normal-flux CT images (sparse-view) and low-flux CT image (full-view) to train

Sparse2Noise. In addition, Sparse2Noise also adopts a modified U-Net architecture for its performance of removing image degradation factors such as noise and ring artifacts. The Sparse2Noise network takes as input the normal-flux CT images into the modified U-Net architecture and outputs the enhanced image. During training, the network is trained in a supervised fashion using the low-flux CT images. The loss function to update the network weights is defined to minimize the difference between the enhanced image and the reconstructed low-flux CT image [87]. Evaluates the effectiveness of Sparse2Noise on one simulated and two experimental datasets. Furthermore, Sparse2Noise is compared to simultaneous iterative reconstruction technique (SIRT), unsupervised deep imaging prior (DIP), and supervised training algorithms Noise2Inverse [88] and Noise2Noise [89]. For the simulated dataset, Sparse2Noise outperforms all methods by achieving the highest SSIM and peak signal to noise ratio (PSNR) values, and in terms of removing image degradation factors such as noise and ring artifacts. For the experimental datasets, Sparse2Noise also achieves the best performance in terms of noise and ring artifact removal. Most importantly, however, Sparse2Noise can achieve excellent performance for low-dose experiments (0.5 Gy per scan).

3.3.1.2 Deblurring

Image deblurring aims to recover a sharp image from a blurred image by suppressing blur factors such as lack of focus, camera shake, and target motion. Some blur factors are application specific such as multiple Coulomb scattering and chromatic aberration in proton radiography [90]. A blurred image can be modeled mathematically as $B = K*I + N$, where B denotes the blurred image, K the blur kernel, I the sharp image, N the additive noise, and $*$ the convolution operation. The blur kernel K is typically modeled as a convolution of blur kernels that are spatially invariant or varying [82]. Conventional methods aim to solve the inverse filtering problem to estimate K, however, this is an ill-posed problem as the sharp image I needs to be estimated as well. To address this issue, prior-based optimization approaches, also known as maximum *a posteriori* (MAP)-based approaches, have been proposed to define priors for K and I [91]. While these approaches are shown to achieve good results for image deblurring, deep learning approaches can further improve the accuracy of the blur kernel estimation or even skip the kernel estimation process altogether by using end-to-end methods. Various neural network architectures, such as CNNs, LSTMs, and GANs, have been proposed for image deblurring; see [82,91] for details.

One example application that uses image deblurring is in neutron imaging restoration (NIR), a non-destructive imaging method. However, the neutron images suffer from noise and blur artifacts due to the neutron source and the digital image system. The low quality of raw neutron images limits their applications in research, and thus image denoising and deblurring techniques are necessary to produce sharp images. To address these issues [92], proposes a fast and lightweight neural network called DAUNet. DAUNet consists of three main blocks: a feature extraction block (FEB), multiple cascaded attention U-Net blocks (AUB), and a reconstruction block (RB). First, DAUNet takes as input a degraded neutron image and feeds it into the FEB to extract

important underlying features. Next, the AUB inputs the extracted feature maps into a modified U-Net with an attention mechanism, which allows U-Net to focus on harder to address features such as texture and structure information, and outputs a restored image. Last, the RB block outputs the enhanced image by reconstructing the restored image. To evaluate DAUNet, its performance is compared with several popular DNN image restoration methods such as DnCNN [93] and RDUNet [94]. Due to the lack of available neutron imaging datasets, the networks are trained on X-ray images that are similar to the neutron imaging principle; specifically, the X-ray images are obtained from the SIXray dataset [95], where 4699 and 23 images are used as the training and test set respectively. In addition, seven clean neutron images are added to the test set. Results show that DAUNet can effectively improve the image quality by removing noise and blurring artifacts, while achieving quality close to the large network with faster running times and a smaller number of network parameters.

3.3.1.3 Super-resolution (SR)

Image SR is the process of reconstructing high-resolution images from low-resolution images. It has been widely applied in many real-world applications, especially in medical imaging [96] and surveillance [97], where the spatial resolution of captured images are not sufficient due to limitations such as hardware and imaging conditions. A variety of DL-based methods for SR have been explored, ranging from CNN-based methods (e.g., SRCNN [98]) to more recent GAN-based methods (e.g., SRGAN [99]). In addition to utilizing different neural network architectures, DL-based SR algorithms also differ in other major aspects such as their loss functions and training approaches [83,100]. These differences result from various factors that contribute to the degradation of image quality including but not limited to blurring, sensor noise, and compression artifacts. Intuitively, one can think of the low-resolution image as the output of a degradation function with an input high-quality image. In the most general case, the degradation function is unknown and an approximate mapping is learned through deep learning. These degradation factors influence the design of loss function, and thus training approaches. A detailed discussion of the various loss functions, SR network architectures, and learning frameworks is out of scope for this paper; however, see [100] for details.

An example application that applies super resolution is for X-ray CT imaging. As mentioned earlier, CT imaging has many factors that impact the resulting image quality such as radiation dose and slice thickness. In addition, 3D image reconstruction may require heavy computational power due to the number of slices or projection views taken, where thicker slices results in lower image resolution, and slower operational speed, which increases with the number of slices. To address this issue, it is desirable to obtain higher-resolution (thin-slice) images from low-resolution (thick-slice) ones [101]. Develops an end-to-end super-resolution method based on a modified U-Net. The network takes as input the low-resolution image and outputs the high-resolution one. The network is trained on slices of brain CT images obtained from a 65 clinical positron emission tomography (PET)/CT studies for Parkinson's disease. The low-resolution images are generated as the moving average of five high resolution slices and the

ground-truth image is taken as the middle slice. The performance of the proposed method is compared with the Richardson-Lucy (RL) deblurring algorithm using the PSNR and normalized root mean square error (NRMSE) metrics. The results show that the proposed method achieves the highest PSNR and lowest NRMSE values compared to the RL algorithm. In addition, the noise level of the enhanced images are reported to be lower than that of the ground-truth.

Super resolution has also been applied to transmission and cryogenic electron microscopy (cryo-EM) imaging applications for sub-pixel electron event localization [25,102]. In transmission electron microscopy, electron events are captured using pixelated detectors as a 2D projection track of the energy deposition [102]. Conventional reconstruction methods, such as the weighted centroid method and the furthest away method (FAM), require an event analysis procedure to extract electron track events. However, these classical algorithms are unable to separate overlapping electron event tracks, and do not take into consideration the statistical behavior of the electron movement and energy deposition. To address this issue [102], used a U-Net-based CNN to learn a mapping from input electron track image to an output probability map that indicates the probability of the point of entry for each pixel. The network is trained using a labeled dataset generated through Monte Carlo simulations, and tested on simulated data and experimental data from a pnCCD [103]. The performance of the proposed CNN model is compared with FAM. The results show that the proposed method achieves superior localization performance compared to FAM by reducing the distribution spread of the Euclidean distance on the simulated dataset, while achieving a modulation transfer function closer to the ground truth on the experimental dataset. For cryo-EM, a CNN model was applied in a similar manner for electron event localization, but with a slightly different dataset. Cyro-EM experiments popularly use MAPS detectors to directly detect electron events, where each captured electron results in a pixel cluster on the captured image. In [25], a CNN model is designed to output a sub-pixel incident position given an input pixel cluster image and the corresponding time over threshold values.

3.3.2 Segmentation

Image segmentation is the process which segments an image or video frames into multiple regions or clusters, where each pixel can be represented by a mask or be assigned a class [104]. This task is essential in a broad range of computer vision problems, especially for visual understanding systems. A few applications that utilize image segmentation include but are not limited to medical imaging for organ and tumor localization [105], autonomous vehicles for surface and object detection, and video footage for object and people detection and tracking [106]. Numerous techniques for image segmentation have been proposed throughout the years, ranging from early techniques based on region splitting or merging such as thresholding and clustering algorithms, to newer algorithms based on active contours and level sets such as graph cuts and Markov random fields [104,107]. Although these conventional methods have achieved acceptable performance for some applications, image segmentation still remains a challenging task due to various image degradation factors such as noise, blur, and contrast. To address these issue, numerous deep learning methods have been developed and have been shown to achieve remarkable performance. This is due to the powerful feature learning capabilities of DNNs, which allows DNNs to have reduced sensitivity to image degradation factors compared to the conventional methods. Popular neural network architectures used for DL-based segmentation includes CNNs, encoder-decoder models, and multiscale architectures; see [107] for details. Two popular DNN architectures used for image segmentation problems are U-Net [76] and SegNet [106]; see [107] for examples of images before and after processing.

Image segmentation is an important step in analyzing X-ray radiographs from, for example, inertial confinement fusion (ICF) experiments [79]. ICF experiments typically use single or double shell targets which are imploded as the laser energy or laser-induced X-rays rapidly compress the target surface. X-ray and neutron radiographs of the target provide insight to the shape of target shells during the implosion. Contour extraction methods are used to extract the shell shape to conduct shot diagnostics such as quantifying the implosion and kinetic energy, identifying shell shape asymmetries, and determining instability information [79]. Uses U-Net [76], a CNN architecture for image segmentation, to output a binary masked image of the outer shell in ICF images. The shell contour is then extracted from the masked image using edge detection and shape extraction methods. Due to the limited number of actual ICF images, a synthetic dataset consisting of 2000 experimental-like radiographs is used to train the U-Net. In addition, the synthetic dataset provides ground-truth ICF image-mask pairs, which are required to train U-Net. The trained U-Net is tested on experimental images and has successfully extracted the binary mask of high-signal-to-noise ratio ICF images as shown in Figure 5.

Another example of X-ray image segmentation is for the Magnetized Linear Inertial Fusion (MagLIF) experiments at Sandia National Laboratory's Z-facility [108]. The MagLIF experiments compresses a cylindrical beryllium tube, also known as a liner, filled with pure deuterium fuel using a very large electric current on the order of O ($20MA$). Before compression, the deuterium fuel is pre-heated and an axially oriented magnetic field is applied. The electric current causes the liner to implode and compresses the deuterium fuel in a quasi-adiabatic implosion. The magnetic field flux is also compressed which aids in the trapping of charged fusion particles at stagnation. X-ray radiographs are taken during the implosion process for diagnostics and to analyze the resulting plasma conditions and liner shape. To better analyze the implosion, a CNN model is proposed to segment the captured X-ray images into fuel strand and background. The CNN is trained using synthetically generated and augmented dataset of 10,000 X-ray images and their corresponding binary masks. The trained CNN is tested on experimental images where the results generally demonstrate excellent fuel-background segmentation performance. The worst segmentation performance is due to factors such as excessive background noise and X-ray image plate damage.

3.3.3 Image classification and object detection

Image classification, a fundamental problem in computer vision, aims to assign labels or categories to images or specific regions in images. It is known to form the basis of other computer vision tasks including segmentation and object detection. Traditional approaches to solve the classification problem typically use a two stage approach, where handcrafted features extracted from the
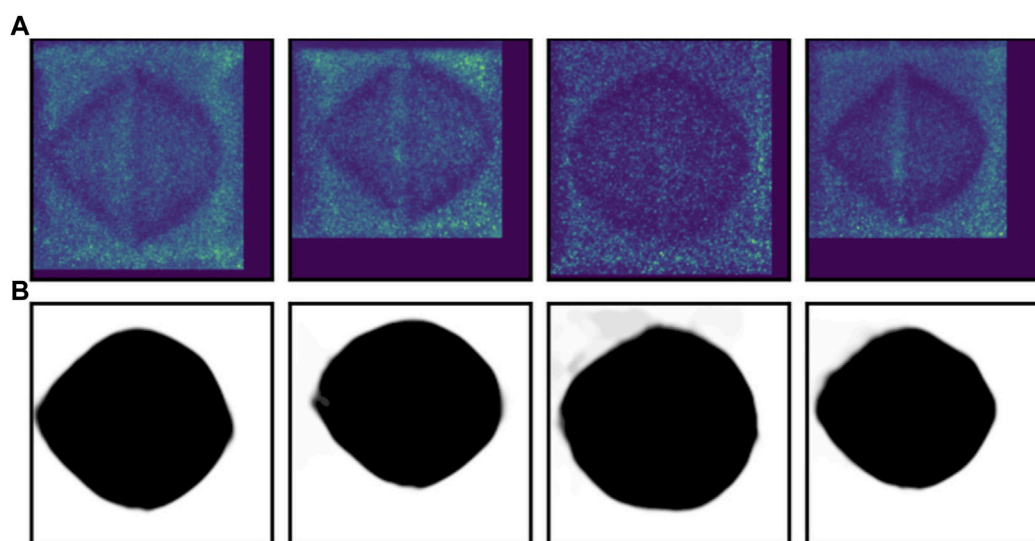
FIGURE 5
Results of image segmentation using U-Net on experimental ICF images **(A)** and the corresponding output masks **(B)**. Reproduced with permission from [79].

image are used to train a classifier. The traditional approaches suffer from low classification accuracy due to the heavy dependence on the design of the handcrafted features. DL approaches can easily overcome this challenge by exploiting neural network layers for automated feature extraction, transformation, and pattern analysis. CNNs are the most popular neural network architecture used for image classification [109,110] due to their capability of reducing the high dimensionality of images without information loss, as discussed in Section 3.2.1. In addition, recall in Section 3.2.1, the CNN architectures AlexNet, VGG, GoogLeNet, and ResNet were top 3 finishers in the ILSVRC. The ILSVRC is an annual software contest where algorithms compete to correctly classify images in the ImageNet database.

Object detection builds upon image classification by estimating the location of the object in an input image in addition to classifying the object. As a result, the workflow for traditional detection algorithms can be broken down into informative region selection, feature extraction, and classification. For informative region selection, a multiscale sliding window (bounding box) is used to scan the image to determine regions of interest. Feature extraction is used on the selected region, which is then used for object classification. However, traditional methods are time consuming and robust algorithms are difficult to design. For example, a large number of candidate sliding windows need to be considered or the algorithm may return bad regions of interest. In addition, the imaging conditions can vary significantly due to factors such as lighting conditions, backgrounds, and distortion effects. Again, DL algorithms can overcome these challenges due to their capability of learning complex features using robust training algorithms [111–113]. Popular DL object detection models are Fast R-CNN [114,115] which jointly optimizes the classification and bounding box regression tasks, You Only Look Once (YOLO) [116] which uses a fixed-grid regression, and Single Shot MultiBox Detector (SSD) [117] which improves upon YOLO using multi-reference and multi-resolution techniques.

3.3.4 Compression

Image compression is the process of reducing the file size, in bytes, without reducing the quality of the image below a threshold. This process is important in order to save memory storage space and to reduce the memory bandwidth to transmit data, especially for running image processing algorithms on edge devices. The fundamental principle of compression is to reduce spatial and visual redundancies in images, by exploiting inter-pixel, psycho-visual, and coding redundancies. Conventional methods commonly leverage various quantization and entropy coding techniques [118]. Popularly used conventional methods for lossy and lossless compression includes but are not limited to JPEG [119], JPEG2000, wavelet, and PNG. While conventional methods are widely used for both image and video compression, their performance is not the most optimal for all types of image and video applications. DL approaches can achieve improved compression results due to several factors. DNNs can learn non-linear mappings to capture the compression process as well as extract the important underlying features of the image through dimensionality reduction. For example, an encoder network or CNN can extract important features into a latent feature space for compact representation. In addition, DNNs can implement direct end-to-end methods using networks such as encoder-decoders to directly obtain the compressed image from an input sharp image. Furthermore, once a DNN is trained, the inference time is much faster. For DL-based image compression methods, the most commonly used neural network architectures are CNNs, encoder-decoders, and GANs [118].

3.3.5 Sparse sampling

A closely related process to image compression is sparse sampling. While compression aims to reduce the file size, sparse sampling, also known as compressed sensing (CS), aims to efficiently acquire and reconstruct a signal by solving underdetermined linear systems. It has been shown in CS theory that a signal can be

recovered from sampling fewer measurements than required by the Niquist-Shannon sampling theorem [120]. As a result, both memory storage space and data transmission bandwidth can be reduced. In conventional methods, CS algorithms need to overcome two main challenges: the design of the sampling and reconstruction matrices. Numerous methods have been proposed including but no limited to random and binary sampling matrices and reconstruction methods using convex-optimization and greedy algorithms [121]. However, these conventional methods suffer from long computational times or low quality reconstruction. DL approaches allow for fast inference (reconstruction) times for a trained network, as well as learning non-linear functions for higher quality signal reconstruction [121,122]; see [123] for examples of images before and after processing.

Neural network (NN) models that learn to invert X-ray data have also been shown to significantly reduce the sampling requirements faced by traditional iterative approaches. For example, in ptychography, traditional iterative phase retrieval methods require at least 50% overlap between adjacent scan positions to successfully reconstruct sample images as required by Nyquist-Shannon sampling. In contrast, Figure 6B shows image reconstructions obtained from PtychoNN when sampled at 25× less than required for conventional phase retrieval methods [124]. Figure 6A shows the probe positions and intensities, there is minimal overlap between probes. Through use of inductive bias provided through online training of the network [125], PtychoNN is able to reproduce most of the features seen in the sample even when provided extremely sparse data. Figure 6C shows the same region reconstructed using an oversampled dataset and traditional iterative phase retrieval. Furthermore [125], demonstrated live inference performance during a real experiment using an edge device and running the detector at its maximum frame rate of 2 kHz.

In the previous example, DL is used to reduce sampling requirements but not to alter the sampling strategy. In other words, the scan proceeds using a conventional acquisition strategy, but using fewer points along that trajectory than traditionally required. In contrast, active learning approaches are being developed that use data-driven priors to direct the acquisition strategy. Typically, this is treated as a Bayesian optimization (BO) problem using Gaussian processes (GPs). This method has been applied to a variety of characterization modalities including scanning probe microscopy [126], X-ray scattering [127], and neutron characterization [128]. A downside to such approaches is that the computational complexity typically increases as $O(N^3)$ with the action space [129], making real-time decision a challenge. To address these scaling limitations which are critical especially in fast scanning instruments, recent work has demonstrated the use of pre-trained NNs to make such control decisions [130,131]. Figure 7 shows the workflow and results from the Fast Autonomous Scanning Toolkit applied to a scanning diffraction X-ray microscopy measurement of a WSe_2 sample. Starting from some quasi-random initial measurements, FAST generates an estimate of the sample morphology, predicts the next batch of 50 points to sample from, triggers acquisition on the instrument, analyzes the image after the next set of points has been acquired and continues the process until the improvement in sample image is minimal. Figures 7B, A, C, and E show the predicted image after 5%, 15%, and 20% sampling while Figure 7 B, D, and F shows the points preferentially selected by the AI. The AI has learned to prioritize

acquisition where the expected information gain is maximum, e.g., around contrast features on the sample.

3.3.6 3D reconstruction

Image-based 3D reconstruction is the process of inferring a 3D structure from a single or multiple 2D images, and is a common topic in the fields of computer vision, medical imaging, and virtual reality. This problem is well known to be an ill-posed inverse problem. Conventional methods attempt to formulate a mathematical formula for the 3D to 2D projection process, use prior models, 2D annotations, and other techniques [132,133]. In addition, high quality reconstruction typically requires 2D projections from multiple views or angles, which may be difficult to calibrate (i.e., cameras) or time consuming to obtain (i.e., CT) depending on the application. DL techniques and the increasing availability of large datasets motivates new advances in 3D reconstruction by address challenges found in conventional methods. The popular networks used for image-based 3D reconstruction are CNNs, encoder-decoder, and GAN models [132]; see [132] for examples on 2D to 3D reconstruction.

X-ray phase information is now available for 3D reconstruction in the state-of-the-art X-ray sources such as synchrotrons and XFELs. In contrast to iterative phase retrieval methods that incorporate NNs through a DIP or other means, single-shot phase retrieval NNs provide sample images from a single pass through a trained NN. The inference time on a trained NN is minimal and such methods are hundreds of times faster than conventional phase retrieval [134,135]. Figures 8A, B compare AutoPhaseNN and traditional phase retrieval for 3D coherent image reconstruction, respectively [136]. AutoPhaseNN is trained to invert 3D coherently scattered data into sample image in a single shot. Once trained AutoPhaseNN is $>100 \times$ faster that iterative phase retrieval with some reduction in accuracy. The prediction from AutoPhaseNN can also be used to seed phase retrieval, i.e., provide an initial estimate which can be refined by a few iterations of phase retrieval. This combined approach of NN + phase retrieval is shown to be both faster and more accurate than iterative phase retrieval.

A recent work by Scheinker and Pokharel [137] developed an adaptive CNN-based 3D reconstruction method for coherent diffraction imaging (CDI), a non-destructive X-ray imaging technique that provides 3D measurements of electron density with nanometer resolution. The CDI detectors record only the intensity of the complex diffraction pattern of the incident object. However, all phase information is lost in this detection method, and thus results in an ill-posed inverse Fourier transform problem to obtain the 3D electron density. Conventional methods encounter many challenges including expert knowledge, sensitivity to small variations, and heavy computation requirements. While DL methods currently cannot completely substitute conventional methods, they can speed up the 3D reconstruction speed given an initial guess, and can be fine-tuned using conventional methods to achieve better performances. For CDI 3D reconstruction, Scheinker and Pokharel [137] proposes a 3D CNN architecture with model-independent adaptive feedback agents. The network takes in 3D diffracted intensities as inputs and outputs a vector of spherical harmonic coefficients, which describe the surface of the 3D incident object. The adaptive feedback agents take as input the
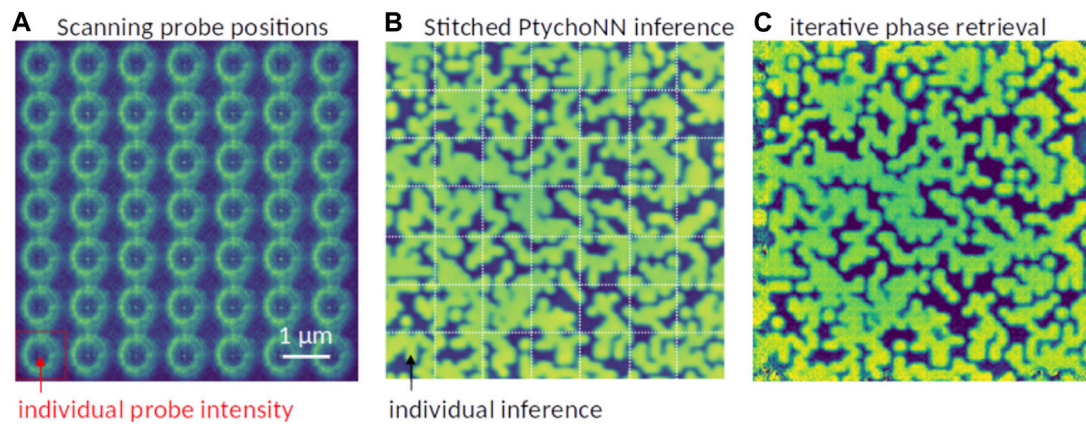
FIGURE 6
Sparse-sampled single-shot ptychography reconstruction using PtychoNN. **(A)** Scanning probe positions with minimal overlap. **(B)** Single-shot PtychoNN predictions on 25 × sub-sampled data compared to **(C)** ePIE reconstruction of the full resolution dataset.
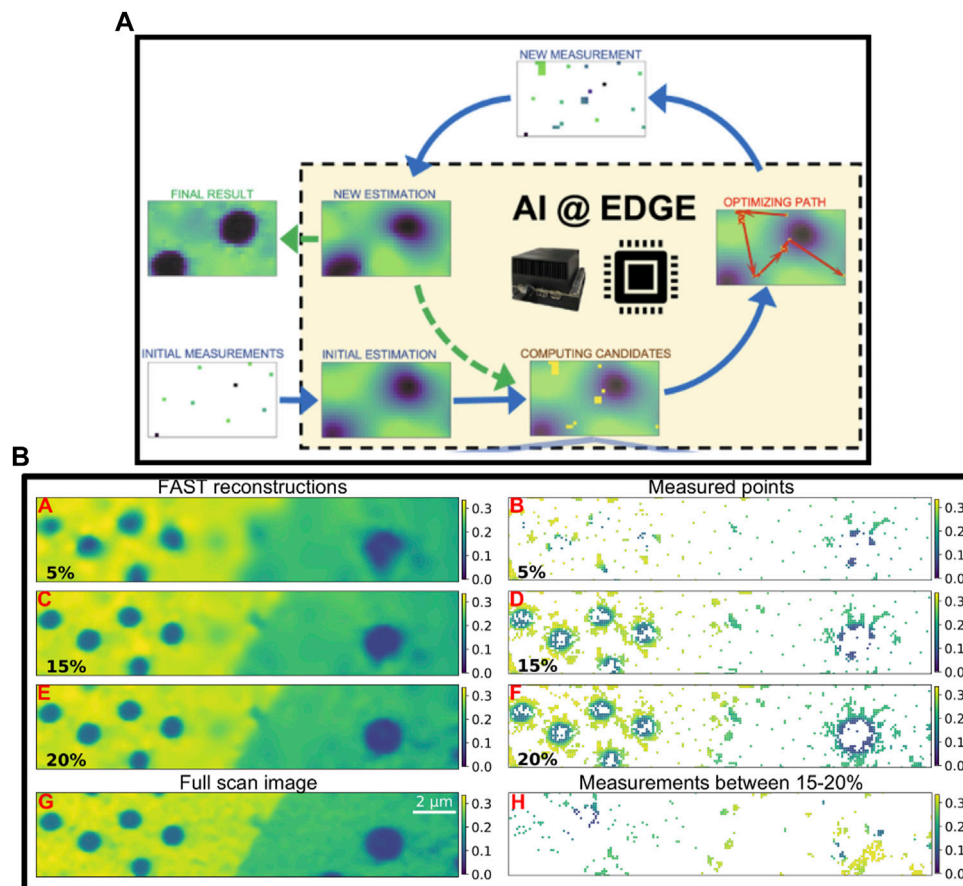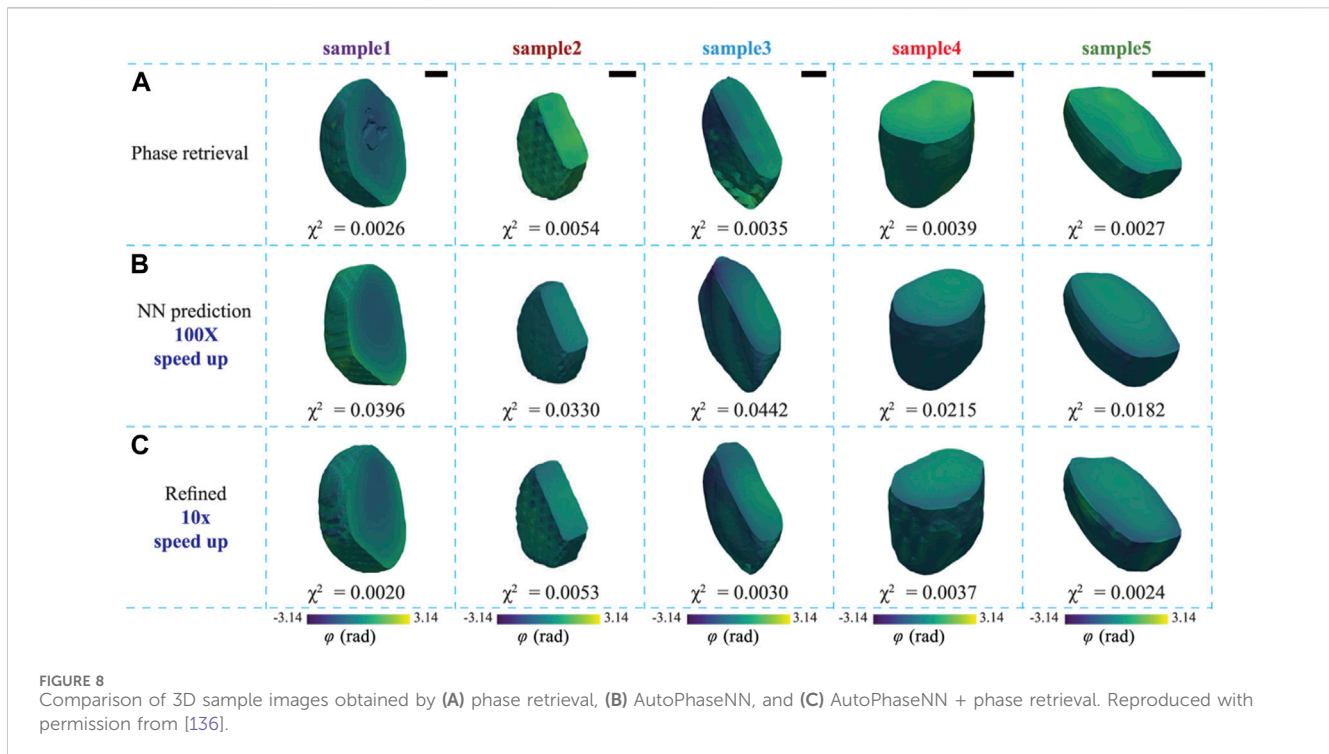

FIGURE 7
FAST framework for autonomous experimentation. **(A)** shows the workflow that enables real-time steering of scanning microscopy experiments. **(B)** shows reconstructed images at 5%, 15% and 20% sampling along with the corresponding locations from which they were sampled. In addition, the full-grid pointwise scan and corresponding points sampled between 15% and 20% is also shown. Reproduced with permission from [131].

spherical harmonics to adaptively adjust the intensities, positions, and decay rates of a collection of radial basis functions. The 3DCNN is trained using a synthetic dataset consisting of 500,000 training set

of 49 sampling coefficients as well as the spherical surface and volume of each in order to perform a 3D Fourier transform. An additional 100 random 3D shapes and their corresponding 3D

FIGURE 8
Comparison of 3D sample images obtained by **(A)** phase retrieval, **(B)** AutoPhaseNN, and **(C)** AutoPhaseNN + phase retrieval. Reproduced with permission from [136].

Fourier transforms are used to test the adaptive model-independent feedback algorithm, with the CNN output as its initial guess. Last, the robustness of the trained 3DCNN is tested on the experimental data of a 3D grain from a polycrystalline copper sample measured using high-energy diffraction microscopy. Results show that the 3DCNN provides an initial guess that captures the average size and a rough estimate of the shape of the grain. The adaptive feedback algorithm uses the 3DCNN initial guess to fine-tune the harmonic coefficients to match and converge the generated and measured diffraction patterns of the grain.

4 Hardware solutions for deep learning

DNNs have been implemented for many imaging processing tasks ranging from enhancement to generation as discussed above. To achieve good performance, these algorithms use very deep networks which can be very computationally intensive during training and inference in their own ways. During training, DNNs are fed large amounts of data and a large number of computations must be performed to update network weights to achieve accurate predictions. For example, AlexNet [68] took five to 6 days to train on two NVIDIA GTX 580 graphical processing units. As a result, powerful computing hardware is needed to accelerate DNN training. Meanwhile, during inference, larger networks require more computing power and memory storage space, and thus results in higher energy consumption and latency to obtain predictions in real-time. For very large networks such as AlexNet, a single forward pass may require millions of multiply and accumulate (MAC) operations, thus making DNNs both computationally and energy costly. For real-time data processing in imaging devices, DNN algorithms need to be executed with low

latency, limited energy, and other design constraints. Hence, there is a need to develop cost and energy efficient hardware solutions for DL applications.

Interestingly, neural network algorithms are known to have at least two types of inherent parallelism, namely, model and data parallelism [138]. Model parallelism refers to the partitioning of the neural network weights for MAC operations for parallel execution as there are no data dependencies. Data parallelism refers to processing the data samples in batches rather than a single sample at a time. Hardware accelerators can exploit these characteristics by implementing parallel computing paradigms. This section presents different hardware accelerators used for DL applications. Note that the best hardware solution is dependent on the application and corresponding design requirements. For example, edge computing devices such as cameras and sensors may require small chip area with limited power consumption.

4.1 Electronic-based accelerators

The electronic-based hardware solutions for DL are broad, ranging from general purpose processors such as central processing units (CPUs) and graphical processing units (GPUs), field-programmable gate arrays (FPGAs), to application-specific integrated circuits (ASICs). The circuit architecture design typically follows either temporal or spatial architectures [139] as shown in Figures 9A, D. The architectures are similar in using multiple processing elements (PEs) for parallel computing, however, there are differences in control, memory, and communication. The temporal architecture features a centralized control for simple PEs, consisting of only arithmetic logic units (ALUs), which can only access data from the centralized memory. Meanwhile, the spatial

architecture features a decentralized control scheme with complex PEs, where each unit can have its own local memory or register file (RF), ALU, and control logic. The decentralized control scheme forms interconnections between neighboring PEs to exchange data directly, allowing for dataflow processing techniques.

4.1.1 Temporal architectures: CPUs and GPUs

CPUs and GPUs are general purpose processors that typically adopt the temporal architecture as shown in Figures 9B, C. Modern CPUs can be realized as vector processors, which adopt the single-instruction multiple-data (SIMD) model to process a single instruction on multiple ALUs simultaneously. In addition, CPUs are optimized for instruction-level parallelism in order to accelerate the execution time of serial algorithms and programs. Meanwhile, modern GPUs adopt the single-instruction multiple threads (SIMT) model to process a single instruction across multiple threads or cores. Different from CPUs, GPUs are made up of more specialized, parallel, and smaller cores than CPUs to efficiently process vector data with high performance and reduced latency. As a result, GPU optimization relies on software defined parallelism rather than instruction-level parallelism [140]. Both the SIMD and SIMT execution models for CPUs and GPUs, respectively, allow for parallel MAC operations for accelerated computations.

Nonetheless, CPUs are not the most used processor for DNN training and inference. Compared to GPUs, CPUs have a limited number of cores, and thus a limited number of parallel executions. For example, one of Intel's server-grade CPUs is the Intel Xeon Platinum 8280 processor which can have up to 28 cores, 56 threads, 131.12 GB/s maximum memory bandwidth, and 2190 Giga-floating point operations per second (GFLOPS) for single-precision compute power. In addition, AMD's server-grade EPYC 9645 features 96 cores, 192 threads, and a memory bandwidth of 460.8 GB/s. In comparison, NVIDIA's GeForce RTX 2080 Ti is a desktop-grade GPU with 4352 CUDA cores, 616.0 GB/s memory bandwidth, and 13450 GFLOPS single-precision compute power. Furthermore, a recently released NVIDIA RTX 4090 desktop-grade GPU features 16,834 CUDA cores, 1008 GB/s memory bandwidth, and 82.85 TFLOPS single-precision compute power. Therefore, GPUs outperform CPUs in terms of parallel computing.

For DL at the edge, the hardware industry has developed embedded platforms for AI. One popular platform is the NVIDIA Jetson for next-generation embedded computing. The Jetson processor features a heterogeneous CPU-GPU architecture [141] where the CPU accelerates the serial instructions and the GPU accelerates the parallel neural network computation. Furthermore, the Jetson is designed with a small form factor, size, and power consumption. A broad survey by [142] presents different works using the Jetson platform for DL applications such as medical, robotics, and speech recognition. Several surveyed works have used the Jetson platform to implement imaging processing tasks including segmentation, object detection, and classification.

Also using the NVIDIA Jetson platform, a work by [143] investigates the performance of the Jetson TX2 for edge deployment for TomoGAN [86], an image denoising technique using generative adversarial networks (GANs) for low-dose X-ray images. The training and testing datasets consist of 1024 pairs of images of size 1024 × 1024 with each image pair consisting of a noisy imag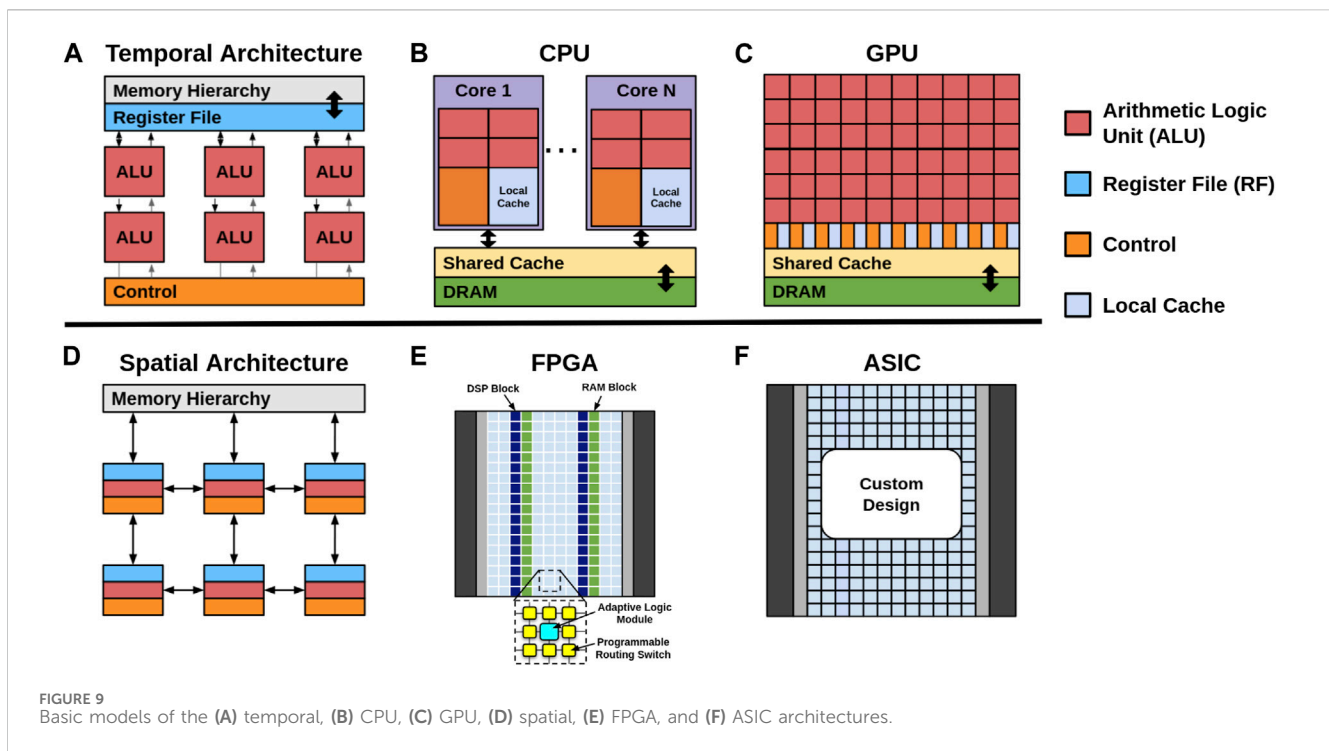e and its corresponding ground truth. The pre-trained TomoGAN network is deployed and tested on the Jetson TX2 and a laptop with an Intel Core i7-6700HQ CPU @2.60GHz with 32GB RAM. The laptop CPU achieves an average inference performance of 1.537 s per image, while the TX2 achieves an inference performance of 0.88 s per image, approximately 1.7× faster than the laptop CPU.

A recent work by [144] investigates the classification accuracy of tuberculosis detection from chest X-ray images using MobileNet [145], ShuffleNet [146], SqueezeNet [147], and their proposed E-TBNet. In addition, they further investigate the inference time during testing of each network on the NVIDIA Jetson Xavier and a laptop with Intel Core i5-9600KF CPU and NVIDIA Titan V GPU. The dataset consists of 800 chest X-ray images scaled to size 512 × 512 × 3. The MobileNet network achieves the highest accuracy at 90% while their proposed E-TBNet achieves 85%. However, the inference time for E-TBNet is the fastest for all investigated networks with an inference time of 0.3 m and 3 m per image when deployed on the laptop with Titian GPU and Jetson Xavier, respectively. The slowest reported inference time for the Jetson Xavier is 6 m per image for the ShuffleNet. Although the inference time for the Xavier is an order of magnitude slower, classification inference can be achieved in real-time with smaller hardware footprint for edge deployment.

4.1.2 Spatial architectures: FPGAs and ASICs

Field-programmable gate arrays (FPGAs) and application-specific integrated circuits (ASICs) typically adopt the spatial architecture as shown in Figures 9E, F. FPGAs and ASICs are specialized hardware that are tailored for specific applications due to their design process. FPGAs can be configured to perform any function as it is made up of programmable logic modules and interconnecting switches as shown in Figure 9E. The FPGA software is used to directly build the logic and data flow directly into the chip architecture. On the other hand, ASICs are designed and optimized for a single application, and cannot be reconfigured. Nonetheless, the spatial architecture of FPGAs and ASICs makes them well suited for neural network computations as the mathematical operations of each layer are fixed and known *a priori*. As a result, FPGAs and ASICs can attain highly optimized performance.

As shown in Figure 9D, the spatial architecture consists of an array of PEs interconnected with a Network-on-Chip (NoC) design, allowing for custom data flow schemes. Although not shown in Figure 9D, the memory hierarchy consists of three levels. The lowest level consists of the RF in each PE, which is used to locally store data for inter-PE data movement or local accumulation operations. The middle level consists of a global buffer (GB) that holds the neural network weights and inputs to feed the PEs. The highest level is the off-chip memory, usually a DRAM, to store the weights and activations of the whole network. MAC operations need to be performed on large data sets. Hence, the major bottleneck is the high latency and energy costs of DRAM accesses. A comparison between DianNao and Cambricon-X, two CNN accelerators, show that DRAM accesses consume more that 80% of the total energy consumption [148]. In addition [149], reports that the energy cost of DRAM access is approximately 200× more than a RF access. Therefore, energy efficiency can be greatly improved through the reduction of DRAM accesses, commonly done by exploiting the idea of data reuse.

FIGURE 9
Basic models of the **(A)** temporal, **(B)** CPU, **(C)** GPU, **(D)** spatial, **(E)** FPGA, and **(F)** ASIC architectures.

The focus of data reuse is to utilize the data already stored in RFs and the GB as often as possible. This gives rise to the investigations of efficient data flow paradigms in both the spatial and temporal operations of PEs. For example, in fully connected layers, the input reuse scheme is popular since the input vector is dot multiplied by each row of the weight matrix to compute the layer output. For convolutional layers, the weight reuse scheme is popular as the weight kernel matrix is used for multiple subsets of the input feature map. In addition for convolutional layers, convolutional reuse can be applied by exploiting the overlapping region of the sliding window of kernel weights and the input feature map. Additional data reuse schemes are the weight stationary, output stationary, row stationary, and no local reuse schemes. A detailed discussion of the data reuse schemes is out of scope for this paper. However, for a comprehensive review, see details in [139,150,151]. In summary, optimizing the data flow is crucial for FPGAs and ASICs to attain high energy efficiency.

Nonetheless, it is important to note the challenges faced by FPGAs, and in turn ASICs, have in implementing DL networks. A few challenges include but are not limited to memory storage requirements, memory bandwidth, and large computational requirements on the order of Giga-operations per second (GOPS). For example, AlexNet requires 250 MB of memory with 32-bit representation to store 60 million model parameters and 1.5 GOPS for each input image [152], while VGG has 138 million model parameters and requires 30 GOPS per image [153]. Commercial FPGAs do not have enough memory storage space and thus requires external memory to store model parameters, which needs to be transmitted to the FPGA during computation. One way to address this issue, is to compress the neural network by reducing its size through methods such as compression and quantization [154,155]. For example, SqueezeNet [147] can be thought of a compressed AlexNet with 50× fewer model

parameters and < 0.5 MB model size. On the other hand, quantization reduces the number of data bits or transforms floating point data to fixed point data to reduce the computational burden. As a result, FPGA-based implementation of DL models will suffer a degree of accuracy loss.

The energy efficiency and massive parallelism of FPGA and ASIC-based accelerators make them desirable for edge computing. A recent work [156] develops a lightweight CNN architecture called SparkNet for image classification tasks. SparkNet features approximately 3× less parameters compared with the SqueezeNet, and approximately 150× less parameters than AlexNet. In addition, a comprehensive design is presented to map all layers of the network onto an Intel Arria 10 GX1150 FPGA platform with each layer mapped to a its own hardware unit to achieve simultaneously pipelined work, increasing throughput. SparkNet is tested on 4 benchmark image classification datasets, i.e., MINIST, CIFAR-10, CIFAR-100 and SVHN. The performance and average time for the Intel FPGA, NVIDIA Titan X GPU, and Intel Xeon E5 CPU to process 10,000 32 × 32 × 3 is reported. The FPGA-based accelerator achieves a processing time of 11.18 µs, which is 41× and 9× faster than the CPU and GPU, respectively. Furthermore, the FPGA average power consumption is 7.58 W with a performance of 337.2 Giga operations per second (GOP/s), making the FPGA more energy and computationally efficient compared to the CPU (95 W, 8.2 GOP/s) and GPU (250 W, 39.4 GOP/s).

Another recent work [157] uses FGPAs to deploy MobileNet for face recognition in a video-based face tracking system. The work further integrates the FPGA with CPUs and GPUs to build a heterogeneous system with a delay-aware energy-efficient scheduling algorithm to achieve reduced execution time, latency, and energy cost. The face tracking experiment is run using an Intel Gold 5118 CPU, NVIDIA Tesla P100 GPU, and the Intel Arria 10 GX 900 and Intel Stratix 10 GX1100 FGPAs. The reported

experimental results evaluate the computing speed and power efficiency of the FPGA-based accelerator compared to the CPU and GPU, as well as the efficiency of the combined detection system with CPU/GPU/FPGA. The FPGA accelerators achieve a computational speed that is approximate to or better than the GPU, while achieving superior power efficiency in GOP/s/W. The difference in performance of the FPGAs is due to their hardware specifications, where the hardware richer Intel Stratix will out perform the Intel Arria. Lastly, the experimental results report that the CPU/GPU/FPGA system can achieve optimal performance in comparison to using only one or a combination of two different accelerators. This is due to the energy efficient scheduling algorithm to optimally pipeline tasks to the different accelerators. The idea of utilizing a heterogeneous system and scheduling algorithm to improve computational and energy efficiency can be explored to address challenges of edge computing.

The Google Edge Tensor Processing Unit (TPU) platform [158] is a general purpose ASIC designed and built by Google for inference at the edge. One example product is the Dual Edge TPU which features an area footprint of 22×30 mm^2, peak perfrmance of 8 trillion operations per second (TOPS), and power consumption of 2 TOPS/W. Other hardware options are available for ASIC prototyping and deployment for edge devices. A survey by [159] presents works that use the Edge TPU platform for DL applications such as image classification, object detection, and image segmentation.

The previously discussed work [143], which deployed TomoGAN on the NVIDIA Jetson platform for X-ray image denoising, also deployed it on the Edge TPU. The work presents a quantized model of TomoGAN to address limitations of the Edge TPU, such as output size. A fine tuning model is also presented to improve the output quality of the quantized model. The Edge TPU's average inference time is 0.554 s per image, which is faster than the Jetson TX2 inference time of 0.88 s per image. In addition, the power consumption is reduced to 2 W compared to Jetson TX2's 7.5 W.

In addition, there is interest in the development of software and firmware for modular and scalable implementation of energy efficient algorithms on FPGA platforms. One such example is for high-speed readout systems for pixel detectors. Oak Ridge National Laboratory (ORNL), through the support of the Department of Energy (DOE) in High Energy Physics (HEP) and Nuclear Physics (NP), is leading the design of a new generic readout system for pixel detectors based on the successful first-generation system, the CARIBOu 2.0 [160]. The CARIBOu 2.0 system, shown in Figure 10, will be the proposed architecture for the platform. The concept of the system is to provide a generic framework for the readout of ASIC detectors for research and development and scalable to larger detector arrays. CARIBou 2.0 shares knowledge and code to provide the community with a convenient platform that maximizes reusability and minimizes overhead when developing such systems. ORNL will initially implement the readout firmware and software specific to the Timepix4 or to commercial CMOS image sensors, SMALLGAD, Photon-to-Digital Converters (PDCs), and the interconnect for the assemblies. The hardware platform is based on Xilinx Ultrascale + FPGA, that provides resources for CPU and FPGA side data processing at high speed. Using the resources of this modern FPGA, software and firmware will be developed to flexibly implement data processing and reduction, edge computing,
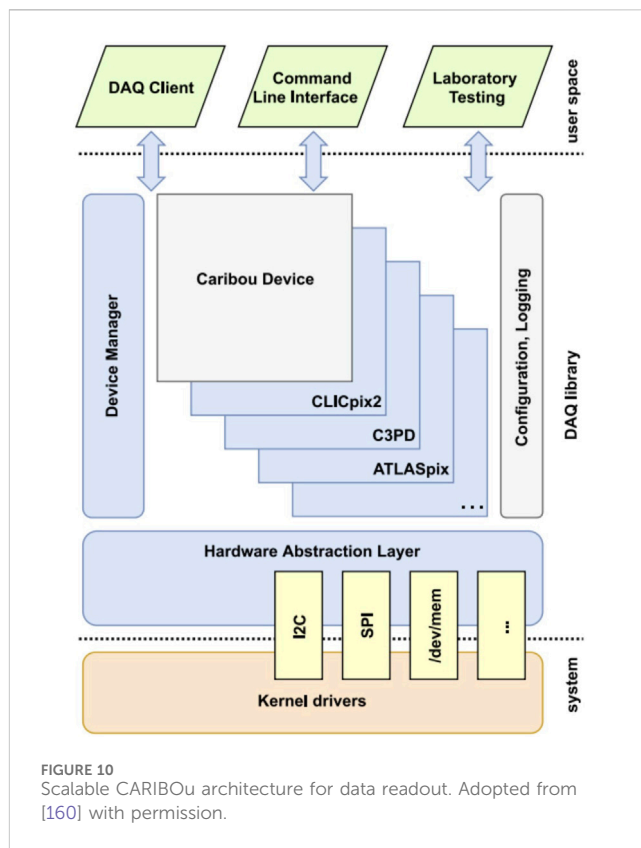

FIGURE 10
Scalable CARIBOu architecture for data readout. Adopted from [160] with permission.

by using conventional and ML algorithms running in the FPGA. For larger data rates, firmware will be developed to move the data to a FELIX card, which can handle up to 24 CARIBOu 2.0 systems and transmit data via a high-speed network interface to a data center or process them locally via GPU and CPU in the FELIX host machine. As a result, the system can be scaled up to the readout of large smart sensor stack arrays.

Furthermore, as advancements in ASIC technology have enabled greater integration of digital functionalities for scientific applications, there has been growing interest in incorporating compression capabilities directly within ASIC detectors to enhance data processing speed. ASIC architectures capable of frame rates approaching 1 MHz have been designed, providing a viable solution for enhancing the speed of various diffraction techniques employed at X-ray light sources, including those relying on coherent imaging methodologies like ptychography [161]. Developing ASIC compression strategies that exploit the structure in detector data enables high compression performance while requiring lower computational complexity than commonly used lossless compression methods like LZ4 [161].

4.1.3 Summary and limitations

We have presented an overview of 4 different electronic-based accelerators and a few works applying them to DL applications at the edge. Figure 11A shows that there is a clear trade-off between programmability and efficiency. To attain higher performance and power efficiency, FPGAs and ASICs require more design complexity to optimize data flow, while ASICs need further hardware optimization. Correspondingly, the time-to-market increases with design complexity. DL algorithms can be deployed

at the edge using these existing electronic-based hardware accelerators.

However, in recent years, these electronic-based accelerators are constantly reaching performance limits in latency, energy consumption, high interconnect cost, excessive heat, and other physical constraints [162]. Figure 11B illustrates the past 50 years of CPU trends in regards to the number of transistors, single-thread performance, frequency, typical power consumption, and number of cores. The trends show that the number of transistors and correspondingly the power consumption continues to grow. Furthermore, the trends indicate that CPU clock frequency has plateaued since around 2005 while single-thread performance and number of cores are slowly tapering. On the other hand, GPU performance has not been limited and is the most popularly used hardware for deep learning training. The single-precision computational throughput of GPUs continues to grow [163]; summarizes the increasing peak performance trends of Nvidia GPUs in GFLOPS from 2006 to 2018. As a result of the high computational speeds, GPUs consume more power [163]. Shows that the power consumption of GPUs increases with the computational throughput in GFLOPS and [164] shows that GPUs have higher computational speed than FGPAs and ASICs at the cost of higher power consumption. Meanwhile, FPGAs and ASICs can achieve good computational performance with lower energy consumption at the cost of design time to develop data flow algorithms and to optimize hardware.

In addition to the hardware performances, the unit prices of each hardware should also be taken into consideration. Recall in Section 4.1.1 we compared the parallel computing performance between server-grade CPUs (Intel Xeon Platinum 8280 and AMD EPYC 9645) and desktop-grade GPUs (NVIDIA RTX 2080 and 4090), where the GPUs outperform CPUs due to the higher number of computing cores. Not only can desktop GPUs perform better than server-grade CPUs, they are also more price efficient. The NVIDIA RTX 2080 and 4090 have starting prices around $1,000 and $1,600, respectively, while the Xeon Platinum 8280 and AMD EPYC 9645 cost over $5,000. The unit prices of FPGAs can vary from as low as a few USD to thousands of USD depending on various factors such as the manufacturer, the number of units, the number of configurable logic blocks, the number of input/output connections, and the amount of available RAM [165]. On the other hand, the unit prices of custom designed ASICs can be lower than that of FPGAs, but only when purchased in large quantities [166]. The starting cost of ASICs is easily over $1,000 as it suffers from very high non-recurring engineering (NRE) costs [167], while FPGAs have no NRE costs. Nonetheless, Google offers prototyping products using the Edge TPU starting at $60.

At any rate, electronic accelerators are traditionally designed to follow von Neumann architecture where the processor and memory units are connected by buses [168], which inherently increases data transfer and power consumption during computation [148]. Demonstrates that more than 75% of the energy utilized by processors comes from DRAM accesses. These limits in electronic based computing gives rise to a shift in focus to analog neuromorphic computing and non-von Neumann architectures such as optical neural networks and bio-inspired spiking neural networks for high-speed, energy-efficient, and parallel computing [169,170].

4.2 Neuromorphic hardware outlook

This section presents two emerging neuromorphic hardware solutions, namely, optical neural networks (ONNs) and spiking neural networks (SNNs), as promising architectures for highly energy efficient and parallel processing.
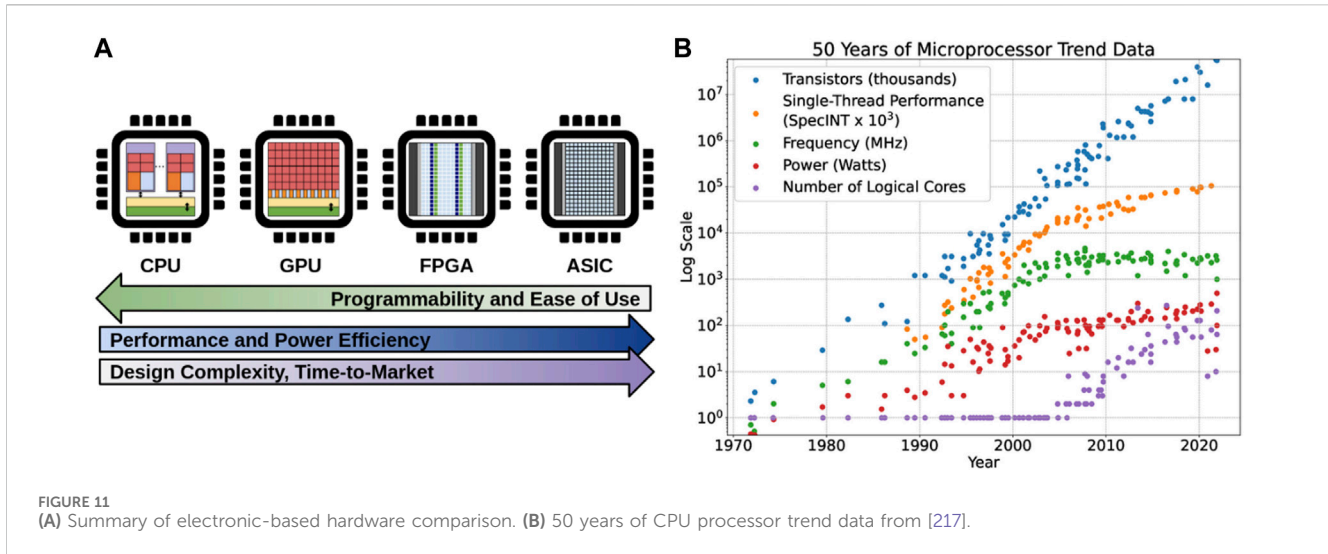
4.2.1 Optical neural networks

ONNs have emerged as a promising avenue for achieving high-performance and energy-efficient computing, given their compute-in-light speed, ultra-high parallelism, and near-zero computation energy [171–174]. Series of photonic tensor cores (PTCs) are designed to enhance the execution of linear matrix operations, the fundamental operations in AI and signal processing, with coherent photonic integrated circuits [171,175], micro-ring resonators [176], photonic phase-change materials [177–179], and diffractive optics [180–182].

Comparing metal wire connections, optical signals modulated at different wavelengths, can be concurrently processed using wavelength-division multiplexing (WDM) within the waveguide and photonic tensor cores [175,183]. Besides, waveguides are free from inductance, which means frequency-dependent signal distortions are negligible for the extended connections in neural interconnects. Hence, given the extensive parallel signal fan-out and fan-in requirements in neural networks, the physical implementation based on PTCs offers distinct advantages. On the basis of the linear optical computing paradigms, ONNs have been constructed for various machine learning tasks such as image classification [182,184,185], vowel recognition [171], and edge detection [186]. Photonic computing methods [187] also feature great potential for supporting advanced Transformer models. Furthermore, ONNs holds significant promise for real-time image processing, where they process image signals directly in light fields, as opposed to after digitalization [188–191]. For instance, recent advancements include the proposal of an image sensor with an ONN encoder [188], which filters relevant information within a scene using an energy-efficient ONN decoder before detection by image sensors.

Despite their advantages, PTCs face significant challenges related to cross-domain signal conversion energy overhead, specifically in analog-to-digital (A/D) and digital-to-analog (D/A) conversion. Moreover, the physical layout constraints of PTCs, manufacturing complexities, and elevated costs have made scalability a primary obstacle in the broad adoption of ONNs. For example, Mach-Zehnder interferometer (MZI)-based PTCs [171] require $O(m^2 + n^2)$ bulky MZIs and approximately $\sim (m + n)$ cascaded MZIs within a single optical path to implement an n-input, m-output layer. Current state-of-art ONNs therefore employ time-division multiplexing with WDM, trading bandwidth and chip complexity.

Efficient analog-to-digital conversion solution [192] and various hardware-software co-design methodologies [186] have been investigated to reduce signal conversion overhead by reducing precision and energy per conversion. In pursuit of enhancing the scalability and efficiency of ONNs, researchers have delved into innovative optimizations at both the architecture and device levels. One noteworthy approach at the architecture level is the introduction of optical subspace neural networks (OSNNs),

FIGURE 11
(A) Summary of electronic-based hardware comparison. **(B)** 50 years of CPU processor trend data from [217].

which make a trade-off between weight representation universality and the reduction of optical component usage, area costs, and energy consumption. For example, a butterfly-style OSNN as shown in Figure 12A, which achieved a remarkable reduction of 7 times in trainable optical components compared to GEMM-based ONNs, was reported and demonstrated a measured accuracy of 94.16% in image recognition tasks [184]. Without sacrificing much model expressiveness, OSNNs can reduce footprints, often ranging from one to several orders of magnitude less than previous MZI-based ONN [171].

At the device level, employing compact custom-designed PTCs, such as multi-operand optical neurons (MOON) [185,193,194], enables the consolidation of matrix operations into arrays of optical components. Figures 12B, C shows a customized multi-operand MZI-based and microring resonator-based PTCs, respectively. Instead of performing a single math operation (e.g., scalar product) per device, MOON fuses a tensor operation in the single device. Crucially, this approach retains the capability to represent general matrices while still maintaining an exceptionally compact layout, in contrast to prior compact tensor designs like star couplers and metasurfaces [195]. One specific achievement in MOON is the development of multi-operand MZI-based (MOMZI) ONN [194], which has realized a two-orders-of-magnitude reduction in propagation loss, delay, and total footprint without losing matrix expressivity. The customized ONN demonstrated an 85.89% measured accuracy in the street view house number (SVHN) recognition dataset with 4-bit control precision. The combined progress in architecture, device design, and optimization techniques is pivotal in advancing the capabilities of ONNs, making themselves efficient, scalable, and practical for AI applications.

4.2.2 Spiking neural networks

In addition to photonic neuromorphic computing, extensive research has been done for other neuromorphic computing architectures. Due to the bottleneck seen in von Neumann architectures, these computing paradigms aim to greatly reduce data movement between memory and PEs to attain high energy efficiency and parallel processing. Taking a unique approach to improve energy efficiency, neuromorphic computing architectures are inspired by the human brain's neurons and synapses. The human brain is extremely energy efficient, where in terms of computing terminology, it is estimated to have a computing power of 1 exaFLOPS while only consuming 20 W. In recent years, there is a rise in interest to explore brain-inspired neural network computing architectures, better known as SNNs [196,197].

SNNs are a special type of artificial neural network (ANN) that closely mimics biological neural networks. While ANNs are traditionally modeled after the brain, there are still many fundamental differences between them such as neuron computation and learning rules. In addition, one major difference is the propagation of information between neurons. Biological neurons, shown in Figure 13A, transmit information to downstream neurons using a spike train of signals, or a time-series of delta functions. The individual spikes (delta functions) are known to be sparse in time and have high information content. Therefore, SNNs are designed to convey information by utilizing the spike timings and spike rates [198,199] as shown in Figure 13C. Furthermore, the advantages of the spiking event sparsity can be exploited in special hardware to reduce energy consumption while maintaining the transmission of high information content [200].

The hardware industry as well as academia are striving to develop unique solutions for neuromorphic computing chips. Intel's Loihi [201] features 128 neuromorphic cores with 1024 spiking neural units per core. A recent work [202] surveys different works that utilize Loihi as a computing platform for applications such as event-based sensing and perception, odor recognition, closed-loop control for robotics, and simultaneous localization and mapping. For medical image analysis [203], uses Loihi to implement a SNN for brain cancer MRI image classification. IBM developed TrueNorth [204], a neurmorphic chip featuring 4096 neuromorphic cores, 1 million spiking neurons and 256 million synapses. A work by [205] uses the TrueNorth computing platform to detect and count cars from input images by mapping CNNs, such as AlexNet and VGG-16, onto TrueNorth. A few other well-known SNN hardwares are Neurogrid [206], BrainScaleS [207], and SpiNNaker [208], which all adopt different solutions to emulate spiking neurons. For a comprehensive review, see details in [209,210].

Due to its low power consumption, SNN hardware is a potential platform for edge computing. A work by [211] presents preliminary
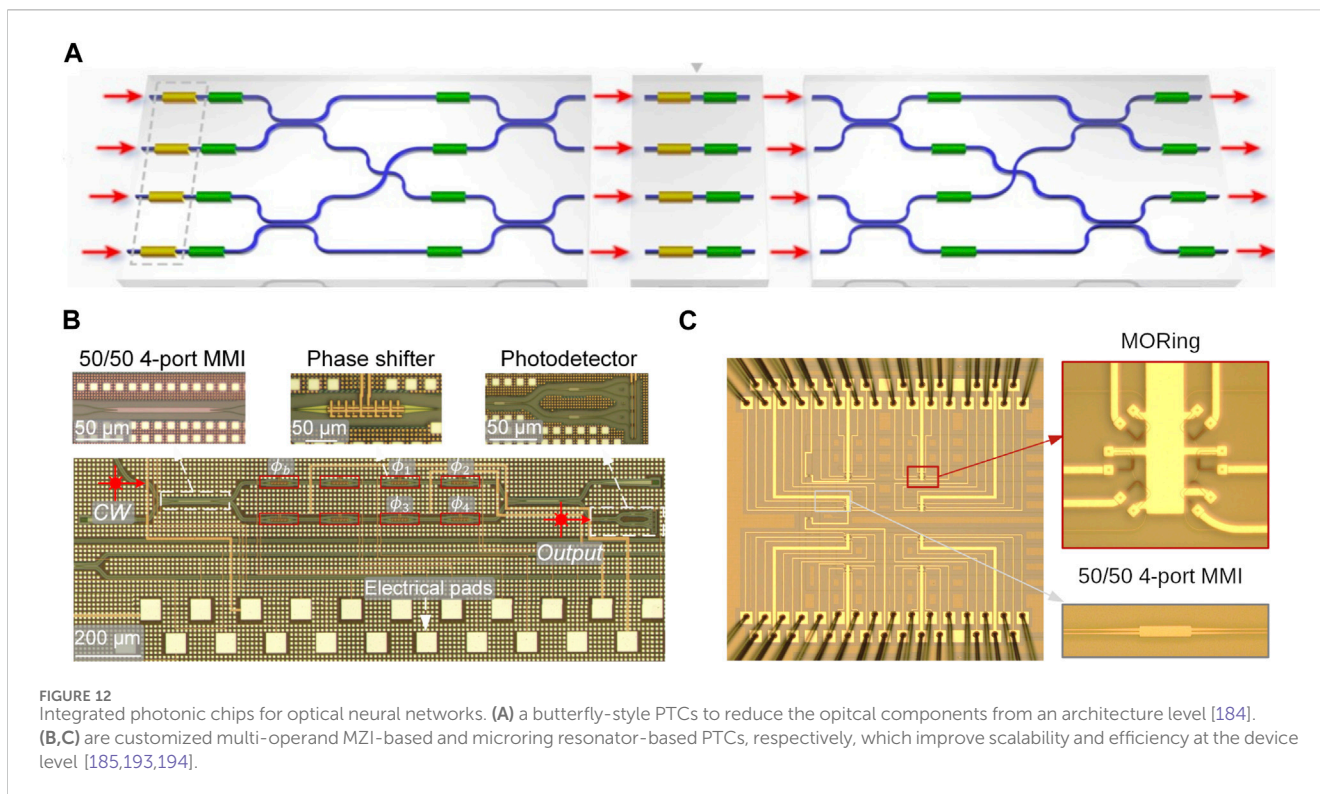
FIGURE 12
Integrated photonic chips for optical neural networks. **(A)** a butterfly-style PTCs to reduce the opitcal components from an architecture level [184]. **(B,C)** are customized multi-operand MZI-based and microring resonator-based PTCs, respectively, which improve scalability and efficiency at the device level [185,193,194].

results for implementing SNN on a mixed analog digital memresistive hardware for classifying neutrino scattering data collected at Fermi National Accelerator Laboratory using the MINERvA detector [212]. Two different SNNs, the neuroscience-inspired dynamic architecture (NIDA) [213] and a memresistive dynamic adaptive neural network array (mrDANNA) [214], were trained and tested on the MINERvA dataset's X view. The training and testing datasets consisted of 10,000 and 90,000 synthetic instances, respectively, generated by a Monte Carlo generator. The NIDA network was trained on the Oak Ridge Leadership Computing Facility's Titan using 10,000 computing nodes, and achieved a classification accuracy of 79.11% on the training set. Meanwhile, the mrDANNA was trained on a desktop and achieved a classification accuracy of 76.14% and 73.59% on the training and combined training and testing dataset, respectively. Both networks can attain an accuracy close to the state-of-art CNN accuracy of 80.42% while using far less neurons and synapses. In addition, the energy consumption was computed for the mrDANNA network and is estimated to be 1.66 μJ per calculation. Although there is an accuracy drop using the smaller SNN networks, the energy consumption per calculation is very small, and thus can be deployed in edge devices.

A recent work (R [215]) implemented an SNN algorithm for filtering data from edge electronics in high energy collider experiments conducted at the High Luminosity Large Hadron Collider (HL-LHC), in order to reduce large data transfer rate or bandwidth (on the order of a few petabytes per second) to downstream electronics. In collider experiments, the collision events of charged particles with energy greater than 2 GeV is of significant interest. However, the high energy charged particles only comprise of approximately 10% of all recorded collision events. Therefore, filtering out low energy particle track clusters will greatly reduce data collection rate at edge devices. A synthetic dataset is used to train and test the SNN.

The full synthetic dataset consists of 4 million charged particle interactions in a silicon pixel sensor. The training dataset is limited to the particle interactions in a 13×21 pixel sub-region of the silicon sensor, with binary classification labels indicating high or low energy. The SNN is realized on Caspian [216], a neuromorphic development platform, and achieved a signal classification accuracy of 91.89%, very close to a prototyped full-precision DNN accuracy of 94.8%. In addition to accuracy, the SNN achieves good performance using nearly half of the number of DNN parameters. The reduced size and improved power efficiency of the SNN model makes it a good candidate for deployment on edge devices which have limited memory and power constraints.

5 Summary

Experimental data generation at photon sources are rapidly increasing due to the advancements in light sources, detectors, and more efficient methods or modalities to collect data. As tabulated in Table 1, detectors can achieve frame-rates on the order of thousands and millions of frames per second in continuous and burst mode, respectively. Each frame can consist of thousands to millions of pixels, depending on the size of the pixel array format, with at least 10-bit data resolution. As a result, the detectors can achieve data rates over 1 GB/s in continuous mode, and orders of magnitude higher data rate in burst mode. Specifically, the state-of-the-art detectors with a 10-bit data format have demonstrated a data rate above 12.5 GB/s in continuous mode and 1.25 TB/s in burst mode. The high data rate is very costly in terms of data storage and transmission over long distances. These issues motivates the use of edge computing on detectors for real-time data processing and for reducing data transmission latency and storage volumes.
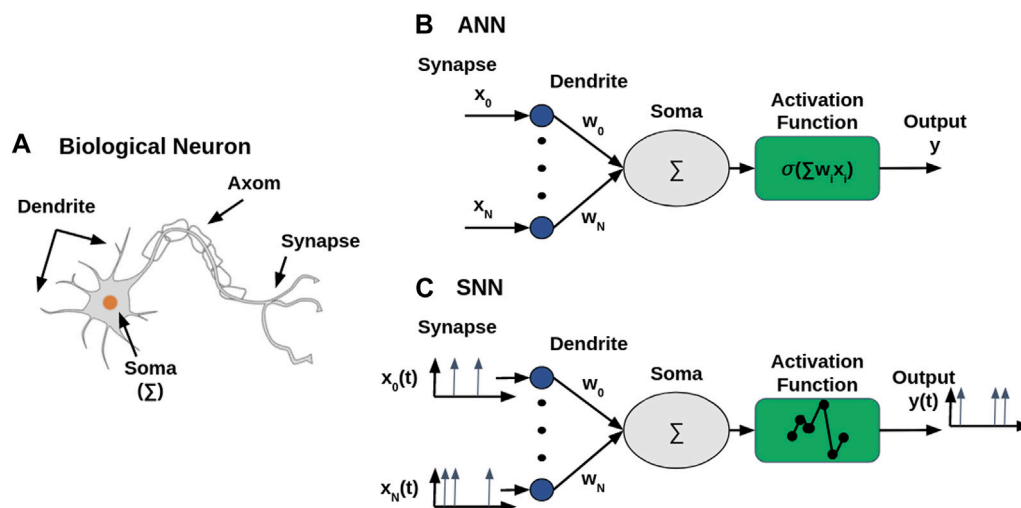
FIGURE 13
A comparison among **(A)** a biological neuron, **(B)** artificial network neuron [46], and **(C)** spiking network neuron [197].

Deep learning approaches have achieved significant progress in image processing tasks including but not limited to restoration, segmentation, compression, and 3D reconstruction. Their superior nonlinear approximation capabilities allow them to learn complex underlying structures and patterns in high dimensional data. The state-of-the-art methods for each image processing task achieve superior performance compared to conventional methods, while also overcoming the issues of conventional methods such as computational burdens associated with explicit programming for each data processing steps. Furthermore, once trained, deep learning methods can achieve very fast inference speeds for real-time computation.

While deep learning approaches are widely used for many applications, they require deep networks to achieve good performance, and thus require heavy computational power and high energy consumption. This is critical hurdle for edge computing devices which have design constraints such as latency and energy. To address this issue, hardware accelerators now exist that leverage the model and data parallelism characteristics of neural network algorithms to implement parallel computing paradigms. Electronic-based hardware accelerators such as CPUs, GPUs, FPGAs, and ASICs are popularly used platforms for deep learning. However, the electronic-based solutions are constantly reaching performance limitations in clock speed, energy consumption, and other physical constraints. This gives rise to research in analog neuromorphic computing paradigms such as ONNs and SNNs to achieve high-speed, energy-efficient, and high-parallel computing, with significant potential for radiation detection and applications in photon science. Nonetheless, note that the power constraint can be alleviated if the experimental space can accommodate the installation of larger processing centers such as workstations or servers, as well as the necessary data transmission networks. Furthermore, a larger processing center allows for the deployment of heavier DL models with improved accuracy for experiments that do not necessarily require real-time processing. To help alleviate data transmission, it is possible to offload simple computing and preprocessing steps to downstream edge devices.

Conflict of interest

Author RC was employed by Omega Optics Inc.

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.

References

1. Wang Z, Leong AF, Dragone A, Gleason AE, Ballabriga R, Campbell C, et al. Ultrafast radiographic imaging and tracking: an overview of instruments, methods, data, and applications. *Nucl Instr Methods Phys Res Section A: Acc Spectrometers, Detectors Associated Equipment* (2023) 1057:168690. doi:10.1016/j.nima.2023.168690

2. Young L, Ueda K, Gühr M, Bucksbaum PH, Simon M, Mukamel S, et al. Roadmap of ultrafast x-ray atomic and molecular physics. *J Phys B: At Mol Opt Phys* (2018) 51:032003. doi:10.1088/1361-6455/aa9735

3. Nature Reviews Physics. *The next decade of XFELs* (2020). doi:10.1038/s42254-020-0206-4

4. Russo P. *Handbook of X-ray imaging: physics and technology*. Boca Raton, Florida: CRC Press (2017).

5. Weisenburger S, Sandoghdar V. Light microscopy: an ongoing contemporary revolution. *Contemp Phys* (2015) 56:123–43. doi:10.1080/00107514.2015.1026557

6. Lu W, Friedrich B, Noll T, Zhou K, Hallmann J, Ansaldi G, et al. Development of a hard x-ray split-and-delay line and performance simulations for two-color pump-probe experiments at the european xfel. *Rev Scientific Instr* (2018) 89:063121. doi:10.1063/1.5027071

7. Inoue I, Inubushi Y, Sato T, Tono K, Katayama T, Kameshima T, et al. Observation of femtosecond x-ray interactions with matter using an x-ray–x-ray pump–probe scheme. *Proc Natl Acad Sci* (2016) 113:1492–7. doi:10.1073/pnas.1516426113

8. Eberhardt W. Synchrotron radiation: a continuing revolution in x-ray science—diffraction limited storage rings and beyond. *J Electron Spectrosc Relat Phenomena* (2015) 200:31–9. doi:10.1016/j.elspec.2015.06.009

9. Dooling J, Borland M, Berg W, Calvey J, Decker G, Emery L, et al. Collimator irradiation studies in the argonne advanced photon source at energy densities expected in next-generation storage ring light sources. *Phys Rev Accel Beams* (2022) 25:043001. doi:10.1103/physrevaccelbeams.25.043001

10. Schroer CG, Agapov I, Brefeld W, Brinkmann R, Chae Y-C, Chao H-C, et al. PETRA IV: the ultralow-emittance source project at DESY. *J Synchrotron Radiat* (2018) 25:1277–90. doi:10.1107/S1600577518008858

11. Huang N, Deng H, Liu B, Wang D, Zhao Z. Features and futures of x-ray free-electron lasers. *The Innovation* (2021) 2(2):100097. doi:10.1016/j.xinn.2021.100097

12. LeCun Y, Bengio Y, Hinton G. Deep learning. *nature* (2015) 521:436–44. doi:10.1038/nature14539

13. Goodfellow I, Bengio Y, Courville A. *Deep learning*. MIT Press (2016). http://www.deeplearningbook.org (Accessed October 10, 2023).

14. Wu J-L, Kashinath K, Albert A, Chirila D, Prabhat Xiao H. Enforcing statistical constraints in generative adversarial networks for modeling chaotic dynamical systems. *J Comput Phys* (2020) 406:109209. doi:10.1016/j.jcp.2019.109209

15. Lin YT, Tian Y, Livescu D, Anghel M. Data-driven learning for the mori–zwanzig formalism: a generalization of the koopman learning framework. *SIAM J Appl Dynamical Syst* (2021) 20:2558–601. doi:10.1137/21m1401759

16. Kochkov D, Smith JA, Alieva A, Wang Q, Brenner MP, Hoyer S. Machine learning–accelerated computational fluid dynamics. *Proc Natl Acad Sci* (2021) 118:e2101784118. doi:10.1073/pnas.2101784118

17. O'Driscoll L, Nichols R, Knott PA. A hybrid machine learning algorithm for designing quantum experiments. *Quan Machine Intelligence* (2019) 1:5–15. doi:10.1007/s42484-019-00003-8

18. Li J, Lopez S. *Machine learning accelerated photodynamics simulations* (2023).

19. Cern . *CERN Data Centre passes the 200-petabyte milestone* (2017). https://home.cern/news/news/computing/cern-data-centre-passes-200-petabyte-milestone (Accessed December 15, 2023).

20. Chen J, Ran X. Deep learning with edge computing: a review. *Proc IEEE* (2019) 107:1655–74. doi:10.1109/jproc.2019.2921977

21. Kumar P. CMOS vs CCD: why CMOS sensors are ruling the world of embedded vision (2023). Available at: https://www.edge-ai-vision.com/2023/04/cmos-vs-ccd-why-cmos-sensors-are-ruling-the-world-of-embedded-vision/ (Accessed December 9, 2023).

22. Tabrizchi S, Nezhadi A, Angizi S, Roohi A. Appcip: energy-efficient approximate convolution-in-pixel scheme for neural network acceleration. *IEEE J Emerging Selected Top Circuits Syst* (2023) 13:225–36. doi:10.1109/jetcas.2023.3242167

23. So HM, Bose L, Dudek P, Wetzstein G (2023). Pixelrnn: in-pixel recurrent neural networks for end-to-end-optimized perception with neural sensors. *arXiv preprint arXiv:2304.05440*

24. Snoeys W. Monolithic cmos sensors for high energy physics–challenges and perspectives. In: *Nuclear instruments and methods in physics research section A: accelerators, spectrometers, detectors and associated equipment* (2023). 168678.

25. van Schayck JP, Zhang Y, Knoops K, Peters PJ, Ravelli RB. Integration of an event-driven timepix3 hybrid pixel detector into a cryo-em workflow. *Microsc Microanalysis* (2023) 29:352–63. doi:10.1093/micmic/ozac009

26. Tsigaridas S, Ponchut C. High-z pixel sensors for synchrotron applications. In: *Advanced X-ray detector technologies: design and applications* (2022). p. 87–107.

27. Porter J, Looker Q, Claus L. Hybrid cmos detectors for high-speed x-ray imaging. *Rev Scientific Instr* (2023) 94:061101. doi:10.1063/5.0138264

28. Carulla M, Doblas A, Flores D, Galloway Z, Hidalgo S, Kramberger G, et al. 50μm thin low gain avalanche detectors (lgad) for timing applications. *Nucl Instr Methods Phys Res Section A: Acc Spectrometers, Detectors Associated Equipment* (2019) 924:373–9. doi:10.1016/j.nima.2018.08.041

29. Zhang J, Barten R, Baruffaldi F, Bergamaschi A, Borghi G, Boscardin M, et al. Development of lgad sensors with a thin entrance window for soft x-ray detection. *J Instrumentation* (2022) 17:C11011. doi:10.1088/1748-0221/17/11/c11011

30. Giacomini G. Lgad-based silicon sensors for 4d detectors. *Sensors* (2023) 23:2132. doi:10.3390/s23042132

31. Pietropaolo A, Angelone M, Bedogni R, Colonna N, Hurd A, Khaplanov A, et al. Neutron detection techniques from μev to gev. *Phys Rep* (2020) 875:1–65. doi:10.1016/j.physrep.2020.06.003

32. Graafsma H. Hybrid pixel array detectors for photon science. In: *Semiconductor radiation detection systems*. Boca Raton, Florida: CRC Press (2018). p. 229–48.

33. Niemann B, Rudolph D, Schmahl G. X-ray microscopy with synchrotron radiation. *Appl Opt* (1976) 15:1883–4. doi:10.1364/ao.15.001883

34. Spanne P, Raven C, Snigireva I, Snigirev A. In-line holography and phase-contrast microtomography with high energy x-rays. *Phys Med Biol* (1999) 44:741–9. doi:10.1088/0031-9155/44/3/016

35. Miao J, Ishikawa T, Robinson IK, Murnane MM. Beyond crystallography: diffractive imaging using coherent x-ray light sources. *Science* (2015) 348:530–5. doi:10.1126/science.aaa1394

36. Pfeiffer F. X-ray ptychography. *Nat Photon* (2018) 12:9–17. doi:10.1038/s41566-017-0072-5

37. Villanueva-Perez P, Bajt S, Chapman H. Dose efficient compton x-ray microscopy. *Optica* (2018) 5:450–7. doi:10.1364/optica.5.000450

38. Gomez J, Patel S, Sarwar SS, Li Z, Capoccia R, Wang Z, et al. *Distributed on-sensor compute system for ar/vr devices: a semi-analytical simulation framework for power estimation* (2022). *arXiv preprint arXiv:2203.07474*.

39. Allahgholi A, Becker J, Delfs A, Dinapoli R, Goettlicher P, Graafsma H, et al. Megapixels@ megahertz–the agipd high-speed cameras for the european xfel. *Nucl Instr Methods Phys Res Section A: Acc Spectrometers, Detectors Associated Equipment* (2019) 942:162324. doi:10.1016/j.nima.2019.06.065

40. Allahgholi A, Becker J, Delfs A, Dinapoli R, Goettlicher P, Greiffenberg D, et al. The adaptive gain integrating pixel detector at the european xfel. *J synchrotron Radiat* (2019) 26:74–82. doi:10.1107/s1600577518016077

41. Lin S, Baldwin J, Blatnik M, Clayton S, Cude-Woods C, Currie S, et al. Demonstration of sub-micron ucn position resolution using room-temperature cmos sensor. *Nucl Instr Methods Phys Res Section A: Acc Spectrometers, Detectors Associated Equipment* (2023) 1057:168769. doi:10.1016/j.nima.2023.168769

42. Orponen P. Computational complexity of neural networks: a survey. *Nordic J Comp* (1994) 94–110.

43. Maji P, Mullins R. On the reduction of computational complexity of deep convolutional neural networks. *Entropy* (2018) 20:305. doi:10.3390/e20040305

44. Freire PJ, Srivallapanondh S, Napoli A, Prilepsky JE, Turitsyn SK. *Computational complexity evaluation of neural network applications in signal processing* (2022). arXiv: 2206.12191.

45. Golub GH, Van Loan CF. *Matrix computations*. 4 edn. Baltimore, MD: Johns Hopkins University Press (2013).

46. Meedeniya D. *Deep learning: a beginners' guide*. Boca Raton, Florida: CRC Press (2023).

47. Yip W. *Lifecycle of machine learning models* (2024). https://www.oracle.com/a/ocom/docs/data-science-lifecycle-ebook.pdf (Accessed December 9, 2023).

48. Patterson J, Gibson A. *Deep learning: a practitioner's approach*. Sebastopol, CA: O'Reilly Media, Inc. (2017).

49. Zhang A, Lipton ZC, Li M, Smola AJ. *Dive into deep learning*. Cambridge University Press (2023).

50. Masanet E, Shehabi A, Lei N, Smith S, Koomey J. Recalibrating global data center energy-use estimates. *Science* (2020) 367:984–6. doi:10.1126/science.aba3758

51. IEA. Data centres and data transmission networks (2023). https://www.iea.org/energy-system/buildings/data-centres-and-data-transmission-networks (Accessed October 15, 2023).

52. RMI. Cryptocurrency's energy consumption problem (2023). https://rmi.org/cryptocurrencys-energy-consumption-problem/ (Accessed December 5, 2023).

53. Kohli V, Chakravarty S, Chamola V, Sangwan KS, Zeadally S. An analysis of energy consumption and carbon footprints of cryptocurrencies and possible solutions. *Digital Commun Networks* (2023) 9:79–89. doi:10.1016/j.dcan.2022.06.017

54. Kirillov A, Mintun E, Ravi N, Mao H, Rolland C, Gustafson L, et al. Segment anything (2023). *arXiv* 2304 https://segment-anything.com/ (Accessed December 5, 2023).

55. Morse J. *NVIDIA's Project Clara is creating game-changing technology for medical imaging* (2018). https://community.radrounds.com/profiles/blogs/nvidia-s-project-clara-is-creating-game-changing-technology-for (Accessed December 5, 2023).

56. Marko K. Ai-enhanced instrumentation - the fusion of deep learning and medical sensors creates dramatic improvements (2018). https://diginomica.com/ai-enhanced-instrumentation-fusion-deep-learning-medical-sensors-\\creates-dramatic-improvements (Accessed December 5, 2023).

57. Milletari F, Navab N, Ahmadi S-A V-net: fully convolutional neural networks for volumetric medical image segmentation. In: 2016 Fourth International Conference on 3D Vision (3DV); Stanford, CA (2016), 3DV. IEEE. 565–71.

58. Cern . *Environmental awareness: the challenges of CERN's IT infrastructure* (2022). https://home.cern/news/news/cern/environmental-awareness-challenges-cerns-it-infrastructure (Accessed December 18, 2023).

59. Adam-Bourdarios C, Cowan G, Germain C, Guyon I, Kégl B, Rousseau D. The Higgs boson machine learning challenge. In: NIPS 2014 Workshop on High-energy Physics and Machine Learning; Montreal, Canada. PMLR (2015). p. 19–55.

60. Azhari M, Abarda A, Ettaki B, Zerouaoui J, Dakkon M. Higgs boson discovery using machine learning methods with pyspark. *Proced Comp Sci* (2020) 170:1141–6. doi:10.1016/j.procs.2020.03.053

61. Kalinin SV, Ziatdinov M, Hinkle J, Jesse S, Ghosh A, Kelley KP, et al. Automated and autonomous experiments in electron and scanning probe microscopy. *ACS nano* (2021) 15:12604–27. doi:10.1021/acsnano.1c02104

62. Khan S, Naseer M, Hayat M, Zamir SW, Khan FS, Shah M. Transformers in vision: a survey. *ACM Comput Surv (Csur)* (2022) 54:1–41. doi:10.1145/3505244

63. Zhang N, Ding S, Zhang J, Xue Y. An overview on restricted Boltzmann machines. *Neurocomputing* (2018) 275:1186–99. doi:10.1016/j.neucom.2017.09.065

64. Cao J, Lin Z. Extreme learning machines on high dimensional and large data applications: a survey. *Math Probl Eng* (2015) 2015(3):1–13. doi:10.1155/2015/103796

65. Zeiler MD. *Hierarchical convolutional deep learning in computer vision*. Ph.D. thesis. Ann Arbor, Michigan: New York University Proquest (2013).

66. Boureau YL, Ponce J, LeCun Y. A theoretical analysis of feature pooling in visual recognition. In: Proceedings of the 27th international conference on machine learning (ICML-10) (2010). p. 111–8.

67. Scherer D, Müller A, Behnke S. Evaluation of pooling operations in convolutional architectures for object recognition. In: International conference on artificial neural networks. Springer (2010). 92–101.

68. Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. *Adv Neural Inf Process Syst* (2012) 25. doi:10.1145/3065386

69. Simonyan K, Zisserman A. *Very deep convolutional networks for large-scale image recognition* (2014). arXiv preprint arXiv:1409.1556.

70. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, et al. Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2015). p. 1–9.

71. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2016). p. 770–8.

72. Yu Y, Si X, Hu C, Zhang J. A review of recurrent neural networks: lstm cells and network architectures. *Neural Comput* (2019) 31:1235–70. doi:10.1162/neco_a_01199

73. Bahdanau D, Cho K, Bengio Y. *Neural machine translation by jointly learning to align and translate* (2014). arXiv preprint arXiv:1409.0473.

74. Herdade S, Kappeler A, Boakye K, Soares J. Image captioning: transforming objects into words. *Adv Neural Inf Process Syst* (2019) 32.

75. Chiu C-C, Sainath TN, Wu Y, Prabhavalkar R, Nguyen P, Chen Z, et al. State-of-the-art speech recognition with sequence-to-sequence models. In: 2018 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE (2018). p. 4774–8.

76. Ronneberger O, Fischer P, Brox T. U-net: convolutional networks for biomedical image segmentation. In: Proceedings, Part III 18 Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference; October 5-9, 2015; Munich, Germany. Springer (2015). p. 234–41.

77. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, et al. Generative adversarial nets. *Adv Neural Inf Process Syst* (2014) 27. doi:10.1145/3422622

78. Creswell A, White T, Dumoulin V, Arulkumaran K, Sengupta B, Bharath AA. Generative adversarial networks: an overview. *IEEE Signal Processing Magazine* (2018) 35:53–65. doi:10.1109/msp.2017.2765202

79. Falato M, Wo lfe B, Nguyen N, Zhang X, Wang Z. *Contour extraction of inertial confinement fusion images by data augmentation* (2022). arXiv preprint arXiv: 2211.04597.

80. Wali A, Naseer A, Tamoor M, Gilani S. Recent progress in digital image restoration techniques: a review. *Digital Signal Process.* (2023) 141:104187. doi:10.1016/j.dsp.2023.104187

81. Fan L, Zhang F, Fan H, Zhang C. Brief review of image denoising techniques. *Vis Comput Industry, Biomed Art* (2019) 2:7–12. doi:10.1186/s42492-019-0016-7

82. Zhang K, Ren W, Luo W, Lai W-S, Stenger B, Yang M-H, et al. Deep image deblurring: a survey. *Int J Comp Vis* (2022) 130:2103–30. doi:10.1007/s11263-022-01633-5

83. Yang W, Zhang X, Tian Y, Wang W, Xue J-H, Liao Q. Deep learning for single image super-resolution: a brief review. *IEEE Trans Multimedia* (2019) 21:3106–21. doi:10.1109/tmm.2019.2919431

84. Tian C, Fei L, Zheng W, Xu Y, Zuo W, Lin C-W. Deep learning on image denoising: an overview. *Neural Networks* (2020) 131:251–75. doi:10.1016/j.neunet.2020.07.025

85. Lucas A, Iliadis M, Molina R, Katsaggelos AK. Using deep neural networks for inverse problems in imaging: beyond analytical methods. *IEEE Signal Process. Mag* (2018) 35:20–36. doi:10.1109/msp.2017.2760358

86. Liu Z, Bicer T, Kettimuthu R, Gursoy D, De Carlo F, Foster I. Tomogan: low-dose synchrotron x-ray tomography with generative adversarial networks: discussion. *JOSA A* (2020) 37:422–34. doi:10.1364/josaa.375595

87. Duan X, Ding XF, Li N, Wu F-X, Chen X, Zhu N. Sparse2noise: low-dose synchrotron x-ray tomography without high-quality reference data. *Comput Biol Med* (2023) 165:107473. doi:10.1016/j.compbiomed.2023.107473

88. Hendriksen AA, Bührer M, Leone L, Merlini M, Vigano N, Pelt DM, et al. Deep denoising for multi-dimensional synchrotron x-ray tomography without high-quality reference data. *Scientific Rep* (2021) 11:11895. doi:10.1038/s41598-021-91084-8

89. Lehtinen J, Munkberg J, Hasselgren J, Laine S, Karras T, Aittala M, et al. *Noise2noise: learning image restoration without clean data* (2018). arXiv preprint arXiv:1803.04189.

90. Morris CL, King N, Kwiatkowski K, Mariam F, Merrill F, Saunders A. Charged particle radiography. *Rep Prog Phys* (2013) 76:046301. doi:10.1088/0034-4885/76/4/046301

91. Biyouki SA, Hwangbo H. *A comprehensive survey on deep neural image deblurring* (2023). arXiv preprint arXiv:2310.04719.

92. Yang J, Zhao C, Qiao S, Zhang T, Yao X. Deep learning methods for neutron image restoration. *Ann Nucl Energ* (2023) 188:109820. doi:10.1016/j.anucene.2023.109820

93. Zhang K, Zuo W, Chen Y, Meng D, Zhang L. Beyond a Gaussian denoiser: residual learning of deep cnn for image denoising. *IEEE Trans image Process* (2017) 26:3142–55. doi:10.1109/tip.2017.2662206

94. Gurrola-Ramos J, Dalmau O, Alarcón TE. A residual dense u-net neural network for image denoising. *IEEE Access* (2021) 9:31742–54. doi:10.1109/access.2021.3061062

95. Miao C, Xie L, Wan F, Su C, Liu H, Jiao J, et al. Sixray: a large-scale security inspection x-ray benchmark for prohibited item discovery in overlapping images. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (2019). p. 2119–28.

96. Li Y, Sixou B, Peyrin F. A review of the deep learning methods for medical images super resolution problems. *Irbm* (2021) 42:120–33. doi:10.1016/j.irbm.2020.08.004

97. Jiang J, Wang C, Liu X, Ma J. Deep learning-based face super-resolution: a survey. *ACM Comput Surv (Csur)* (2021) 55:1–36. doi:10.1145/3485132

98. Dong C, Loy CC, He K, Tang X. Image super-resolution using deep convolutional networks. *IEEE Trans pattern Anal machine intelligence* (2015) 38:295–307. doi:10.1109/tpami.2015.2439281

99. Ledig C, Theis L, Huszár F, Caballero J, Cunningham A, Acosta A, et al. Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2017).

100. Wang Z, Chen J, Hoi SC. Deep learning for image super-resolution: a survey. *IEEE Trans pattern Anal machine intelligence* (2020) 43:3365–87. doi:10.1109/tpami.2020.2982166

101. Park J, Hwang D, Kim KY, Kang SK, Kim YK, Lee JS. Computed tomography super-resolution using deep convolutional neural network. *Phys Med Biol* (2018) 63:145011. doi:10.1088/1361-6560/aacdd4

102. Eckert B, Aschauer S, Holl P, Majewski P, Zabel T, Strüder L. Electron imaging reconstruction for pixelated semiconductor tracking detectors in transmission electron microscopes using the approach of convolutional neural networks. *IEEE Trans Nucl Sci* (2022) 69:1014–21. doi:10.1109/tns.2022.3169281

103. Ryll H, Simson M, Hartmann R, Holl P, Huth M, Ihle S, et al. A pnccd-based, fast direct single electron imaging camera for tem and stem. *J Instrumentation* (2016) 11:P04006. doi:10.1088/1748-0221/11/04/p04006

104. Szeliski R. *Computer vision: algorithms and applications.* Springer Nature (2022).

105. Wang R, Lei T, Cui R, Zhang B, Meng H, Nandi AK. Medical image segmentation using deep learning: a survey. *IET Image Process* (2022) 16:1243–67. doi:10.1049/ipr2.12419

106. Badrinarayanan V, Kendall A, Cipolla R. Segnet: a deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans pattern Anal machine intelligence* (2017) 39:2481–95. doi:10.1109/tpami.2016.2644615

107. Minaee S, Boykov Y, Porikli F, Plaza A, Kehtarnavaz N, Terzopoulos D. Image segmentation using deep learning: a survey. *IEEE Trans pattern Anal machine intelligence* (2021) 44:3523–42. doi:10.1109/TPAMI.2021.3059968

108. Lewis WE, Knapp PF, Harding EC, Beckwith K. Statistical characterization of experimental magnetized liner inertial fusion stagnation images using deep-learning-based fuel–background segmentation. *J Plasma Phys* (2022) 88:895880501. doi:10.1017/s0022377822000800

109. Rawat W, Wang Z. Deep convolutional neural networks for image classification: a comprehensive review. *Neural Comput* (2017) 29:2352–449. doi:10.1162/neco_a_00990

110. Pak M, Kim S A review of deep learning in image recognition. In: 2017 4th international conference on computer applications and information processing technology (CAIPT) (2017). IEEE. 1–3.

111. Zhao Z-Q, Zheng P, Xu S-t., Wu X. Object detection with deep learning: a review. *IEEE Trans Neural networks Learn Syst* (2019) 30:3212–32. doi:10.1109/tnnls.2018.2876865

112. Sharma VK, Mir RN. A comprehensive and systematic look up into deep learning based object detection techniques: a review. *Comp Sci Rev* (2020) 38:100301. doi:10.1016/j.cosrev.2020.100301

113. Jiang X, Hadid A, Pang Y, Granger E, Feng X. *Deep learning in object detection and recognition* (2019).

114. Girshick R. Fast r-cnn. In: Proceedings of the IEEE international conference on computer vision (2015). p. 1440–8.

115. Ren S, He K, Girshick R, Sun J. Faster r-cnn: towards real-time object detection with region proposal networks. *Adv Neural Inf Process Syst* (2015) 28.

116. Redmon J, Divvala S, Girshick R, Farhadi A. You only look once: unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2016). p. 779–88.

117. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C-Y, et al. Ssd: single shot multibox detector. In: Proceedings, Part I Computer Vision–ECCV 2016: 14th European Conference; October 11–14, 2016; Amsterdam, The Netherlands, 14. Springer (2016). p. 21–37.

118. Mishra D, Singh SK, Singh RK. Deep architectures for image compression: a critical review. *Signal Process.* (2022) 191:108346. doi:10.1016/j.sigpro.2021.108346

119. Wallace GK. The jpeg still picture compression standard. *IEEE Trans consumer Electron* (1992) 38. xviii–xxxiv. doi:10.1109/30.125072

120. Donoho DL. Compressed sensing. *IEEE Trans Inf Theor* (2006) 52:1289–306. doi:10.1109/tit.2006.871582

121. Shi W, Jiang F, Liu S, Zhao D. Image compressed sensing using convolutional neural network. *IEEE Trans Image Process* (2019) 29:375–88. doi:10.1109/tip.2019.2928136

122. Machidon AL, Pejović V. Deep learning for compressive sensing: a ubiquitous systems perspective. *Artif Intelligence Rev* (2023) 56:3619–58. doi:10.1007/s10462-022-10259-5

123. Qiao M, Meng Z, Ma J, Yuan X. Deep learning for video compressive sensing. *Apl Photon* (2020) 5. doi:10.1063/1.5140721

124. Cherukara MJ, Zhou T, Nashed Y, Enfedaque P, Hexemer A, Harder RJ, et al. Ai-enabled high-resolution scanning coherent diffraction imaging. *Appl Phys Lett* (2020) 117. doi:10.1063/5.0013065

125. Babu AV, Zhou T, Kandel S, Bicer T, Liu Z, Judge W, et al. *Deep learning at the edge enables real-time streaming ptychographic imaging* (2022). arXiv preprint arXiv:2209.09408. doi:10.1038/s41467-023-41496-z

126. Vasudevan RK, Kelley KP, Hinkle J, Funakubo H, Jesse S, Kalinin SV, et al. Autonomous experiments in scanning probe microscopy and spectroscopy: choosing where to explore polarization dynamics in ferroelectrics. *ACS nano* (2021) 15:11253–62. doi:10.1021/acsnano.0c10239

127. Noack MM, Yager KG, Fukuto M, Doerk GS, Li R, Sethian JA. A kriging-based approach to autonomous experimentation with applications to x-ray scattering. *Scientific Rep* (2019) 9:11809. doi:10.1038/s41598-019-48114-3

128. Venkatakrishnan S, Fancher CM, Ziatdinov M, Vasudevan R, Saleeby K, Haley J, et al. Adaptive sampling for accelerating neutron diffraction-based strain mapping. *Machine Learn Sci Tech* (2023) 4:025001. doi:10.1088/2632-2153/acc512

129. Liu H, Ong Y-S, Shen X, Cai J. When Gaussian process meets big data: a review of scalable gps. *IEEE Trans Neural networks Learn Syst* (2020) 31:4405–23. doi:10.1109/tnnls.2019.2957109

130. Schloz M, Müller J, Pekin TC, Van den Broek W, Madsen J, Susi T, et al. Deep reinforcement learning for data-driven adaptive scanning in ptychography. *Scientific Rep* (2023) 13:8732. doi:10.1038/s41598-023-35740-1

131. Kandel S, Zhou T, Babu AV, Di Z, Li X, Ma X, et al. Demonstration of an ai-driven workflow for autonomous high-resolution scanning microscopy. *Nat Commun* (2023) 14:5501. doi:10.1038/s41467-023-40339-1

132. Han X-F, Laga H, Bennamoun M. Image-based 3d object reconstruction: state-of-the-art and trends in the deep learning era. *IEEE Trans pattern Anal machine intelligence* (2019) 43:1578–604. doi:10.1109/tpami.2019.2954885

133. Fu K, Peng J, He Q, Zhang H. Single image 3d object reconstruction based on deep learning: a review. *Multimedia Tools Appl* (2021) 80:463–98. doi:10.1007/s11042-020-09722-8

134. Guan Z, Tsai EH, Huang X, Yager KG, Qin H. *Ptychonet: fast and high quality phase retrieval for ptychography*. Upton, NY (United States): Brookhaven National Lab (2019). Tech. rep.

135. Cherukara MJ, Nashed YS, Harder RJ. Real-time coherent diffraction inversion using deep generative networks. *Scientific Rep* (2018) 8:16520. doi:10.1038/s41598-018-34525-1

136. Yao Y, Chan H, Sankaranarayanan S, Balaprakash P, Harder RJ, Cherukara MJ. Autophasenn: unsupervised physics-aware deep learning of 3d nanoscale bragg coherent diffraction imaging. *npj Comput Mater* (2022) 8:124. doi:10.1038/s41524-022-00803-w

137. Scheinker A, Pokharel R. Adaptive 3d convolutional neural network-based reconstruction method for 3d coherent diffraction imaging. *J Appl Phys* (2020) 128. doi:10.1063/5.0014725

138. Gholami A, Azad A, Jin P, Keutzer K, Buluc A. Integrated model, batch, and domain parallelism in training neural networks. In: Proceedings of the 30th on Symposium on Parallelism in Algorithms and Architectures (2018). p. 77–86.

139. Sze V, Chen Y-H, Yang T-J, Emer JS. Efficient processing of deep neural networks: a tutorial and survey. *Proc IEEE* (2017) 105:2295–329. doi:10.1109/jproc.2017.2761740

140. Intel. *Compare benefits of CPUs, GPUs, and FPGAs for different oneAPI compute workloads* (2022). https://www.intel.com/content/www/us/en/developer/articles/technical/comparing-cpus-gpus-and-fpgas-for-oneapi.html (Accessed September 25, 2023).

141. Mittal S, Vetter JS. A survey of cpu-gpu heterogeneous computing techniques. *ACM Comput Surv (Csur)* (2015) 47:1–35. doi:10.1145/2788396

142. Mittal S. A survey on optimized implementation of deep learning models on the nvidia jetson platform. *J Syst Architecture* (2019) 97:428–42. doi:10.1016/j.sysarc.2019.01.011

143. Abeykoon V, Liu Z, Kettimuthu R, Fox G, Foster I. Scientific image restoration anywhere. In: 2019 IEEE/ACM 1st Annual Workshop on Large-scale Experiment-in-the-Loop Computing (XLOOP). IEEE (2019). p. 8–13.

144. An L, Peng K, Yang X, Huang P, Luo Y, Feng P, et al. E-tbnet: light deep neural network for automatic detection of tuberculosis with x-ray dr imaging. *Sensors* (2022) 22:821. doi:10.3390/s22030821

145. Sandler M, Howard A, Zhu M, Zhmoginov A, Chen L-C. Mobilenetv2: inverted residuals and linear bottlenecks. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2018). p. 4510–20.

146. Zhang X, Zhou X, Lin M, Sun J. Shufflenet: an extremely efficient convolutional neural network for mobile devices. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2018). p. 6848–56.

147. Iandola FN, Han S, Moskewicz MW, Ashraf K, Dally WJ, Keutzer K. *Squeezenet: alexnet-level accuracy with 50x fewer parameters and< 0.5 mb model size* (2016). *arXiv preprint arXiv:1602.07360.*

148. Li J, Yan G, Lu W, Jiang S, Gong S, Wu J, et al. Smartshuttle: optimizing off-chip memory accesses for deep learning accelerators. In: 2018 Design, Automation & Test in Europe Conference & Exhibition (DATE). IEEE (2018). 343–8.

149. Chen Y-H, Emer J, Sze V. Eyeriss: a spatial architecture for energy-efficient dataflow for convolutional neural networks. *ACM SIGARCH Comput architecture News* (2016) 44:367–79. doi:10.1145/3007787.3001177

150. Capra M, Bussolino B, Marchisio A, Masera G, Martina M, Shafique M. Hardware and software optimizations for accelerating deep neural networks: survey of current trends, challenges, and the road ahead. *IEEE Access* (2020) 8:225134–80. doi:10.1109/access.2020.3039858

151. Dhilleswararao P, Boppu S, Manikandan MS, Cenkeramaddi LR. *Efficient hardware architectures for accelerating deep neural networks: survey.* IEEE Access (2022).

152. Suda N, Chandra V, Dasika G, Mohanty A, Ma Y, Vrudhula S, et al. Throughput-optimized opencl-based fpga accelerator for large-scale convolutional neural networks. In: Proceedings of the 2016 ACM/SIGDA international symposium on field-programmable gate arrays (2016). p. 16–25.

153. Qiu J, Wang J, Yao S, Guo K, Li B, Zhou E, et al. Going deeper with embedded fpga platform for convolutional neural network. In Proceedings of the 2016 ACM/SIGDA international symposium on field-programmable gate arrays (2016).

154. Shawahna A, Sait SM, El-Maleh A. Fpga-based accelerators of deep learning networks for learning and classification: a review. *ieee Access* (2018) 7:7823–59. doi:10.1109/access.2018.2890150

155. Wu R, Guo X, Du J, Li J. Accelerating neural network inference on fpga-based platforms—a survey. *Electronics* (2021) 10:1025. doi:10.3390/electronics10091025

156. Xia M, Huang Z, Tian L, Wang H, Chang V, Zhu Y, et al. Sparknoc: an energy-efficiency fpga-based accelerator using optimized lightweight cnn for edge computing. *J Syst Architecture* (2021) 115:101991. doi:10.1016/j.sysarc.2021.101991

157. Liu X, Yang J, Zou C, Chen Q, Yan X, Chen Y, et al. Collaborative edge computing with fpga-based cnn accelerators for energy-efficient and time-aware face tracking system. *IEEE Trans Comput Soc Syst* (2021) 9:252–66. doi:10.1109/tcss.2021.3059318

158. Cass S. Taking ai to the edge: Google's tpu now comes in a maker-friendly package. *IEEE Spectr* (2019) 56:16–7. doi:10.1109/mspec.2019.8701189

159. Sun Y, Kist AM. *Deep learning on edge tpus* (2021). arXiv preprint arXiv:2108.13732.

160. Liu H, Benoit M, Chen H, Chen K, Di Bello F, Iacobucci G, et al. Development of a modular test system for the silicon sensor r&d of the atlas upgrade. *J Instrumentation* (2017) 12:P01008. doi:10.1088/1748-0221/12/01/p01008

161. Strempfer S, Zhou T, Yoshii K, Hammer M, Babu A, Bycul D, et al. A lightweight, user-configurable detector asic digital architecture with on-chip data compression for mhz x-ray coherent diffraction imaging. *J Instrumentation* (2022) 17:P10042. doi:10.1088/1748-0221/17/10/p10042

162. Waldrop MM. More than moore. *Nature* (2016) 530:144–8. doi:10.1038/530144a

163. Feng X, Jiang Y, Yang X, Du M, Li X. Computer vision algorithms and hardware implementations: a survey. *Integration* (2019) 69:309–20. doi:10.1016/j.vlsi.2019.07.005

164. Le Kernec J, Fioranelli F, Ding C, Zhao H, Sun L, Hong H, et al. Radar signal processing for sensing in assisted living: the challenges associated with real-time implementation of emerging algorithms. *IEEE Signal Process. Mag* (2019) 36:29–41. doi:10.1109/msp.2019.2903715

165. Curtis H. *FPGA programming and its cost comparison* (2024). https://hillmancurtis.com/fpga-programming-and-its-cost-comparison/ (Accessed December 6, 2023).

166. Singh R. *FPGA vs ASIC: differences between them and which one to use* (2018).

167. Sigenics. *Asics* (2024). https://sigenics.com/page/asics-c (Accessed December 6, 2023).

168. Heuring VP, Murdocca MJ. *Principles of computer architecture* (2021).

169. Ganguly A, Muralidhar R, Singh V. Towards energy efficient non-von neumann architectures for deep learning. In: 20th international symposium on quality electronic design (ISQED). IEEE (2019). p. 335–42.

170. Sui X, Wu Q, Liu J, Chen Q, Gu G. A review of optical neural networks. *IEEE Access* (2020) 8:70773–83. doi:10.1109/access.2020.2987333

171. Shen Y, Harris NC, Skirlo S, Prabhu M, Baehr-Jones T, Hochberg M, et al. Deep learning with coherent nanophotonic circuits. *Nat Photon* (2017) 11:441–6. doi:10.1038/nphoton.2017.93

172. Shastri BJ, Tait AN, Ferreira de Lima T, Pernice WH, Bhaskaran H, Wright CD, et al. Photonics for artificial intelligence and neuromorphic computing. *Nat Photon* (2021) 15:102–14. doi:10.1038/s41566-020-00754-y

173. Feng C, Ning S, Gu J, Zhu H, Pan DZ, Chen RT. Integrated photonics for computing and artificial intelligence. In: 2023 IEEE Photonics Society Summer Topicals Meeting Series (SUM). IEEE (2023). p. 1–2.

174. Gu J, Feng C, Zhu H, Chen RT, Pan DZ. Light in ai: toward efficient neurocomputing with optical neural networks—a tutorial. *IEEE Trans Circuits Syst Express Briefs* (2022) 69:2581–5. doi:10.1109/tcsii.2022.3171170

175. Wu J, Lin X, Guo Y, Liu J, Fang L, Jiao S, et al. Analog optical computing for artificial intelligence. *Engineering* (2022) 10:133–45. doi:10.1016/j.eng.2021.06.021

176. Tait AN, De Lima TF, Zhou E, Wu AX, Nahmias MA, Shastri BJ, et al. Neuromorphic photonic networks using silicon photonic weight banks. *Scientific Rep* (2017) 7:7430. doi:10.1038/s41598-017-07754-z

177. Feldmann J, Youngblood N, Karpov M, Gehring H, Li X, Stappers M, et al. Parallel convolutional processing using an integrated photonic tensor core. *Nature* (2021) 589:52–8. doi:10.1038/s41586-020-03070-1

178. Ríos C, Youngblood N, Cheng Z, Le Gallo M, Pernice WH, Wright CD, et al. In-memory computing on a photonic platform. *Sci Adv* (2019) 5. 5759. eaau5759. doi:10.1126/sciadv.aau5759

179. Zhu H, Gu J, Feng C, Liu M, Jiang Z, Chen RT, et al. Elight: toward efficient and aging-resilient photonic in-memory neurocomputing. *IEEE Trans Computer-Aided Des Integrated Circuits Syst* (2022) 42:820–33. doi:10.1109/tcad.2022.3180969

180. Lin X, Rivenson Y, Yardimci NT, Veli M, Luo Y, Jarrahi M, et al. All-optical machine learning using diffractive deep neural networks. *Science* (2018) 361:1004–8. doi:10.1126/science.aat8084

181. Yan T, Wu J, Zhou T, Xie H, Xu F, Fan J, et al. Fourier-space diffractive deep neural network. *Phys Rev Lett* (2019) 123:023901. doi:10.1103/physrevlett.123.023901

182. Ashtiani F, Geers AJ, Aflatouni F. An on-chip photonic deep neural network for image classification. *Nature* (2022) 606:501–6. doi:10.1038/s41586-022-04714-0

183. Huang C, Sorger VJ, Miscuglio M, Al-Qadasi M, Mukherjee A, Lampe L, et al. Prospects and applications of photonic neural networks. *Adv Phys X* (2022) 7:1981155. doi:10.1080/23746149.2021.1981155

184. Feng C, Gu J, Zhu H, Ying Z, Zhao Z, Pan DZ, et al. A compact butterfly-style silicon photonic–electronic neural chip for hardware-efficient deep learning. *Acs Photon* (2022) 9:3906–16. doi:10.1021/acsphotonics.2c01188

185. Feng C, Tang R, Gu J, Zhu H, Pan DZ, Chen RT. Optically-interconnected, hardware-efficient, electronic-photonic neural network using compact multi-operand photonic devices. In: Proceedings Volume PC12427, Optical Interconnects XXIII, San Francisco, California. SPIE (2023). p. 1242702.

186. Gu J, Zhao Z, Feng C, Zhu H, Chen RT, Pan DZ. Roq: a noise-aware quantization scheme towards robust optical neural networks with low-bit controls. In: 2020 Design, Automation & Test in Europe Conference & Exhibition (DATE). IEEE (2020). p. 1586–9.

187. Zhu H, Gu J, Wang H, Tang R, Zhang Z, Feng C, et al. Lightening-transformer: A dynamically-operated optically-interconnected photonic transformer accelerator. In *IEEE International Symposium on High-Performance Computer Architecture (HPCA)* (2024). Available at: https://arxiv.org/abs/2305.19533.

188. Wang T, Sohoni MM, Wright LG, Stein MM, Ma S-Y, Onodera T, et al. Image sensing with multilayer nonlinear optical neural networks. *Nat Photon* (2023) 17: 408–15. doi:10.1038/s41566-023-01170-8

189. Zhou T, Wu W, Zhang J, Yu S, Fang L. Ultrafast dynamic machine vision with spatiotemporal photonic computing. *Sci Adv* (2023) 9:eadg4391. doi:10.1126/sciadv.adg4391

190. Yamaguchi T, Arai K, Niiyama T, Uchida A, Sunada S. Time-domain photonic image processor based on speckle projection and reservoir computing. *Commun Phys* (2023) 6:250. doi:10.1038/s42005-023-01368-w

191. Huang L, Mukherjee S, Tanguy Q, Fröch J, Majumdar A. Photonic advantage of optical encoders. In: 2023 Conference on Lasers and Electro-Optics (CLEO). IEEE (2023). p. 1–2.

192. Zhu H, Zhu K, Gu J, Jin H, Chen RT, Incorvia JA, et al. Fuse and mix: macam-enabled analog activation for energy-efficient neural acceleration. In: Proceedings of the 41st IEEE/ACM International Conference on Computer-Aided Design (2022). p. 1–9.

193. Gu J, Feng C, Zhu H, Zhao Z, Ying Z, Liu M, et al. Squeezelight: a multi-operand ring-based optical neural network with cross-layer scalability. *IEEE Trans Computer-Aided Des Integrated Circuits Syst* (2022) 42:807–19. doi:10.1109/tcad.2022.3189567

194. Feng C, Gu J, Zhu H, Tang R, Ning S, Hlaing M, et al. *Integrated multi-operand optical neurons for scalable and hardware-efficient deep learning* (2023). *arXiv preprint arXiv:2305.19592.*

195. Zhu H, Zou J, Zhang H, Shi Y, Luo S, Wang N, et al. Space-efficient optical computing with an integrated chip diffractive neural network. *Nat Commun* (2022) 13: 1044. doi:10.1038/s41467-022-28702-0

196. Nunes JD, Carvalho M, Carneiro D, Cardoso JS. Spiking neural networks: a survey. *IEEE Access* (2022) 10:60738–64. doi:10.1109/access.2022.3179968

197. Roy K, Jaiswal A, Panda P. Towards spike-based machine intelligence with neuromorphic computing. *Nature* (2019) 575:607–17. doi:10.1038/s41586-019-1677-2

198. Tavanaei A, Ghodrati M, Kheradpisheh SR, Masquelier T, Maida A. Deep learning in spiking neural networks. *Neural networks* (2019) 111:47–63. doi:10.1016/j.neunet.2018.12.002

199. Schuman CD, Kulkarni SR, Parsa M, Mitchell JP, Date P, Kay B. Opportunities for neuromorphic computing algorithms and applications. *Nat Comput Sci* (2022) 2:10–9. doi:10.1038/s43588-021-00184-y

200. Pfeiffer M, Pfeil T. Deep learning with spiking neurons: opportunities and challenges. *Front Neurosci* (2018) 12:774. doi:10.3389/fnins.2018.00774

201. Davies M, Srinivasa N, Lin T-H, Chinya G, Cao Y, Choday SH, et al. Loihi: a neuromorphic manycore processor with on-chip learning. *Ieee Micro* (2018) 38:82–99. doi:10.1109/mm.2018.112130359

202. Davies M, Wild A, Orchard G, Sandamirskaya Y, Guerra GAF, Joshi P, et al. Advancing neuromorphic computing with loihi: a survey of results and outlook. *Proc IEEE* (2021) 109:911–34. doi:10.1109/jproc.2021.3067593

203. Getty N, Brettin T, Jin D, Stevens R, Xia F. Deep medical image analysis with representation learning and neuromorphic computing. *Interf Focus* (2021) 11:20190122. doi:10.1098/rsfs.2019.0122

204. Merolla PA, Arthur JV, Alvarez-Icaza R, Cassidy AS, Sawada J, Akopyan F, et al. A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science* (2014) 345:668–73. doi:10.1126/science.1254642

205. Shukla R, Lipasti M, Van Essen B, Moody A, Maruyama N. Remodel: rethinking deep cnn models to detect and count on a neurosynaptic system. *Front Neurosci* (2019) 13:4. doi:10.3389/fnins.2019.00004

206. Benjamin BV, Gao P, McQuinn E, Choudhary S, Chandrasekaran AR, Bussat J-M, et al. Neurogrid: a mixed-analog-digital multichip system for large-scale neural simulations. *Proc IEEE* (2014) 102:699–716. doi:10.1109/jproc.2014.2313565

207. Schemmel J, Brüderle D, Grübl A, Hock M, Meier K, Millner S. A wafer-scale neuromorphic hardware system for large-scale neural modeling. In: 2010 IEEE International Symposium on Circuits and Systems (ISCAS). IEEE (2010). 1947–50.

208. Furber SB, Galluppi F, Temple S, Plana LA. The spinnaker project. *Proc IEEE* (2014) 102:652–65. doi:10.1109/jproc.2014.2304638

209. Bouvier M, Valentian A, Mesquida T, Rummens F, Reyboz M, Vianello E, et al. Spiking neural networks hardware implementations and challenges: a survey. *ACM J Emerging Tech Comput Syst (Jetc)* (2019) 15:1–35. doi:10.1145/3304103

210. Basu A, Deng L, Frenkel C, Zhang X. Spiking neural network integrated circuits: a review of trends and future directions. In: 2022 IEEE Custom Integrated Circuits Conference (CICC). IEEE (2022). 1–8.

211. Schuman CD, Potok TE, Young S, Patton R, Perdue G, Chakma G, et al. Neuromorphic computing for temporal scientific data classification. In: Proceedings of the Neuromorphic Computing Symposium (2017). p. 1–6.

212. Aliaga L, Bagby L, Baldin B, Baumbaugh A, Bodek A, Bradford R, et al. Design, calibration, and performance of the minerva detector. *Nucl Instr Methods Phys Res Section A: Acc Spectrometers, Detectors Associated Equipment* (2014) 743:130–59. doi:10.1016/j.nima.2013.12.053

213. Schuman CD, Birdwell JD, Dean ME. Spatiotemporal classification using neuroscience-inspired dynamic architectures. *Proced Comp Sci* (2014) 41:89–97. doi:10.1016/j.procs.2014.11.089

214. Cady NC. *Development of a memristive dynamic adaptive neural network array (mrdanna)* (2019). of Nanoscale Science, S. P. C., and Engineering.

215. Kulkarni R, Young A, Date P, Rao Miniskar N, Vetter J, Fahim F, et al. On-sensor data filtering using neuromorphic computing for high energy physics experiments. In: Proceedings of the 2023 International Conference on Neuromorphic Systems (2023). p. 1–8.

216. Mitchell JP, Schuman CD, Patton RM, Potok TE. Caspian: a neuromorphic development platform. In: Proceedings of the 2020 Annual Neuro-Inspired Computational Elements Workshop (2020). p. 1–6.

217. Rupp K. *Microprocessor trend data* (2022). https://github.com/karlrupp/microprocessor-trend-data/tree/master (Accessed September 25, 2023).

218. Turchetta R (2017). Towards gfps cmos image sensors. In In Workshop on Computational Image Sensors and Smart Cameras; May, 2017; Barcelona, Spain, WASC. vol. 2017.

219. Philipp HT, Hromalik M, Tate M, Koerner L, Gruner SM. Pixel array detector for x-ray free electron laser experiments. *Nucl Instr Methods Phys Res Section A: Acc Spectrometers, Detectors Associated Equipment* (2011) 649:67–9. doi:10.1016/j.nima.2010.11.189

220. Porro M, Andricek L, Bombelli L, De Vita G, Fiorini C, Fischer P, et al. Expected performance of the depfet sensor with signal compression: a large format x-ray imager with mega-frame readout capability for the european xfel. *Nucl Instr Methods Phys Res Section A: Acc Spectrometers, Detectors Associated Equipment* (2010) 624:509–19. doi:10.1016/j.nima.2010.02.254

221. Carini G, Alonso-Mori R, Blaj G, Caragiulo P, Chollet M, Damiani D, et al. epix100 camera: use and applications at lcls. In: Proceedings of the 12th International Conference on Synchrotron Radiation Instrumentation – SRI2015; New York, NY, 1741. AIP Publishing (2016).

222. Veale M, Seller P, Wilson M, Liotti E. Hexitec: a high-energy x-ray spectroscopic imaging detector for synchrotron applications. *Synchrotron Radiat News* (2018) 31:28–32. doi:10.1080/08940886.2018.1528431

223. Claus L, England T, Fang L, Robertson G, Sanchez M, Trotter D, et al. Design and characterization of an improved, 2 ns, multi-frame imager for the ultra-fast x-ray imager (uxi) program at sandia national laboratories. In: Proceedings Volume 10390, Target Diagnostics Physics and Engineering for Inertial Confinement Fusion VI; San Diego, California, 10390. SPIE (2017). 16–26.

224. Leonarski F, Nan J, Matej Z, Bertrand Q, Furrer A, Gorgisyan I, et al. Kilohertz serial crystallography with the jungfrau detector at a fourth-generation synchrotron source. *IUCrJ* (2023) 10:729–37. doi:10.1107/s2052252523008618

225. Gadkari D, Shanks K, Hu H, Philipp H, Tate M, Thom-Levy J, et al. Characterization of 128 × 128 MM-PAD-2.1 ASIC: a fast framing hard x-ray detector with high dynamic range. *J Instrumentation* (2022) 17:P03003. doi:10.1088/1748-0221/17/03/p03003

226. Tochigi Y, Hanzawa K, Kato Y, Kuroda R, Mutoh H, Hirose R, et al. A global-shutter cmos image sensor with readout speed of 1-tpixel/s burst and 780-mpixel/s continuous. *IEEE J Solid-State Circuits* (2012) 48:329–38. doi:10.1109/jssc.2012.2219685

227. Lewis A, Baker S, Corredor A, Fegenbush L, Fitzpatrick Z, Jones M, et al. New design yields robust large-area framing camera. *Rev Scientific Instr* (2021) 92:083103. doi:10.1063/5.0049110

228. Mozzanica A, Andrä M, Barten R, Bergamaschi A, Chiriotti S, Brückner M, et al. The jungfrau detector for applications at synchrotron light sources and xfels. *Synchr Rad News* (2018) 31:16–20. doi:10.1080/08940886.2018.1528429

229. Hinger V, al Haddad A, Barten R, Bergamaschi A, Brückner M, Carulla M, et al. Advancing the jungfrau detector toward low-energy x-ray applications. *J Instrumentation* (2022) 17:C09027. doi:10.1088/1748-0221/17/09/c09027