



## OPEN ACCESS

## EDITED BY

Cédric Jamet,  
UMR8187 Laboratoire d'océanologie  
et de géosciences (LOG), France

## REVIEWED BY

Qiangqiang Yuan,  
Wuhan University, China  
Maged Marghany,  
Siyah Kuala University, Indonesia

## \*CORRESPONDENCE

Chengzu Bai  
czubai@stu.wzu.edu.cn

## SPECIALTY SECTION

This article was submitted to  
Ocean Observation,  
a section of the journal  
Frontiers in Marine Science

RECEIVED 04 May 2022

ACCEPTED 24 August 2022

PUBLISHED 14 September 2022

## CITATION

Ma M, Bai C, Zhang S, Qian L and  
Yan H (2022) KECANet: A novel  
convolutional kernel network for  
ocean SAR scene classification with  
limited labeled data.  
*Front. Mar. Sci.* 9:935600.  
doi: 10.3389/fmars.2022.935600

## COPYRIGHT

© 2022 Ma, Bai, Zhang, Qian and Yan.  
This is an open-access article  
distributed under the terms of the  
[Creative Commons Attribution License  
\(CC BY\)](https://creativecommons.org/licenses/by/4.0/). The use, distribution or  
reproduction in other forums is  
permitted, provided the original  
author(s) and the copyright owner(s)  
are credited and that the original  
publication in this journal is cited, in  
accordance with accepted academic  
practice. No use, distribution or  
reproduction is permitted which does  
not comply with these terms.

# KECANet: A novel convolutional kernel network for ocean SAR scene classification with limited labeled data

Ming Ma<sup>1,2</sup>, Chengzu Bai<sup>1,2\*</sup>, Shuo Zhang<sup>1,2</sup>, Longxia Qian<sup>3</sup>  
and Hengqian Yan<sup>4</sup>

<sup>1</sup>Beijing Institute of Applied Meteorology, Beijing, China, <sup>2</sup>State Key Laboratory of Geo-Information Engineering, Xi'an, China, <sup>3</sup>School of Science, Nanjing University of Posts and Telecommunications, Nanjing, China, <sup>4</sup>College of Meteorology & Oceanography, National University of Defense Technology, Changsha, China

The ability of synthetic aperture radar (SAR) to capture maritime phenomena is widely acknowledged. However, ocean SAR scene automatic classification remains challenging due to speckle noise interference, the nonlinearities and poor distinguishability of different geophysical phenomena. Kernel entropy component analysis (KECA) was recently proposed as a feature extraction approach. It is capable of handling nonlinear data and revealing different structures of interest. However, KECA suffers from high computational complexity, meaning it cannot penetrate deep for finer feature extraction. To address this issue, this paper proposes an efficient multilayer convolutional kernel network (denoted as KECANet) equipped with KECA for ocean SAR scene classification. The pivoted Cholesky decomposition is employed to accelerate KECA filtering in the network. KECA was trained on hand-labeled but limited samples describing ten oceanic or atmospheric phenomena. Several conventional and state-of-the-art deep learning methods were also included for comparison. According to the classification experiments, KECANet can greatly improve the classification precision of geophysical phenomena, considering that the precision, recall and F-score values increased by 13.3%, 2.3% and 12.2% in average. Overall, the results suggest that KECA is a promising approach for various applications in remote sensing image recognition.

## KEYWORDS

synthetic aperture radar, kernel entropy component analysis, convolutional kernel network, oceanic phenomena detection, cholesky decomposition

# 1 Introduction

The synthetic aperture radar (SAR) has been widely applied in different fields as a result of its unique advantages, including its high resolution, wide swath, and all-time and all-weather application. However, SAR imagery has speckle noise interference and a unique imaging mechanism, which makes target classification in SAR images challenging (Gommenginger et al., 2019; Wang et al., 2020b). Ocean SAR scene classification is particularly difficult due to the following four issues:

1. Environmental conditions, e.g., ocean eddies and wind speed, give rise to the complex features typical of oceanic phenomena in SAR images (Marghany, 2014; Wang et al., 2019a);
2. The distinguishability of backscattering characteristics among various ocean phenomena in SAR images is relatively poor, as compared with their different geometric properties exhibited in the real world (Yan et al., 2020);
3. Limited labeled data are available for modeling due to the long acquisition period and expensive labeling process (Xie et al., 2021);
4. The transmitted signal from the ocean scene to the SAR image can be extended to include nonlinearity and dispersive effects (Krogstad et al., 1994).

To resolve the aforementioned problems, a variety of interesting works have been published with acceptable results (Clausi, 2001; Wu and Liu, 2003; Salberg and Larsen, 2018). These include the especially impressive performances achieved by Convolutional Neural Network (CNN)-based deep learning methods in recent years (Lima et al., 2017; Franz et al., 2018; Krestenitis et al., 2019; Wang et al., 2019b; Li et al., 2020; Li et al., 2022). Lima et al. (2017) attempted to implement automated detection of oceanic fronts in sea surface temperature images using a fine-tuned CNN model, which obtained a significantly higher accuracy than other methods in the experiments. Lguensat et al. (2018) developed a network named EddyNet to achieve effective oceanic eddy detection in sea surface height images. Bao et al. (2020) proposed a detection framework for internal waves using faster regions with convolutional neural networks. This not only obtains nearly 95% recognition accuracy, but achieved a high detection speed of 0.22 s/image. A variant of Inception-v3 CNNs (denoted CMwv) (Wang et al., 2019b) was also employed for automatic SAR image classification, thereby obtaining the optimal accuracy in the detection of geophysical phenomena predefined in the Sentinel-1 satellite SAR dataset. Owing to the multilayer structure, these CNN-based models can robustly extract high-level features of different oceanic phenomena, which is challenging for automated classification in marine environments (Wang et al., 2020a; Yan et al., 2020). One of

the core layers in CNNs is the convolutional layer. This helps the networks extract features from original data using several learnable filters (Qaraei et al., 2021). Training of such filters in conventional CNNs is implemented by the Adam (Kingma and Ba, 2014) or Stochastic Gradient Descent rule (Bottou, 2012). Although these rules can obtain state-of-the-art results, training CNNs remains difficult since conventional CNNs are required to learn millions of parameters, which in turn requires a huge amount of computational resources (Qaraei et al., 2021). In response to this problem, Chan et al. (2015) proposed an efficient CNN (known as PCANet), which uses a two-layer Principal Component Analysis (PCA) structure to learn filters. PCANet is not only layer-wise trained, but can achieve performances on par with the complex deep learning methods for object detection and image classification in the remote sensing field (Gao et al., 2016; Low et al., 2017; Du et al., 2019; Zhang et al., 2020). One limitation of PCANet, however, is that it seeks a linear subspace in which the variance of the input data is maintained at its maximum, meaning that PCANet cannot reveal nonlinear information within remote sensing imagery (Qaraei et al., 2021).

In order to extract more nonlinear features from input data, Mairal et al. (2014) established a new CNN scheme, known as the Convolutional Kernel Network (CKN), bridging a gap between neural networks and kernel methods. CKNs mainly use kernel approximation tricks, such as the Nyström method (Bo and Sminchisescu, 2009) or Random Fourier Features (RFF) (Rahimi and Recht, 2007), to obtain a stack of kernel feature maps, which can be received as multilayer neural networks. Mohammadnia-Qaraei et al. (2018) proposed a new CKN using a fast approximation method based on RFF, which achieved a higher accuracy and faster speed for face recognition than conventional CKNs. Mairal (2016) designed a convolutional kernel network in the context of unsupervised learning, whose applicability has been demonstrated in image classification and super-resolution experiments. Qaraei et al. (2021) developed a CKN that is structurally similar to PCANet, known as RNPCANet, which extends the training techniques of PCANet to a nonlinear case using a randomized kernel approximation trick. The core of RNPCANet relies on the theory of Kernel Principal Component Analysis (KPCA) (Schölkopf et al., 1998). More specifically, RNPCANet implicitly maps the patches from the input data to a higher dimensional feature space, i.e., Reproducing Kernel Hilbert Space (RKHS), using kernel approximation. Then, the principal components extracted in RKHS, i.e., the eigenvectors associated with the top eigenvalues, are employed to assign the filters of each layer (Qaraei et al., 2021). Although RNPCANet equipped with KPCA is superior to traditional CKNs and PCANet in image classification experiments, it performs feature extraction by selecting the top eigenvalues and the corresponding eigenvectors, which cannot reveal the underlying structure of the input data from the perspective of information theory (Zhang and Hancock, 2012).

A novel information-theoretic method, namely kernel entropy component analysis (KECA), was proposed for the nonlinear processing of image patches (Jenssen, 2009). KECA aims to seek optimal eigenvectors (defined as entropic components) in RKHS that can compress the Rényi entropy information of the input space as much as possible instead of selecting top eigenvalues (Bai et al., 2019; Bai et al., 2020a). As compared to the principal components obtained by KPCA that solely correspond to the ranking order of eigenvalues, entropic components are strikingly different and oriented towards the maximum Rényi entropy of the input data (Jenssen, 2009). More specifically, KECA preserves the maximum Rényi entropy in the input space *via* a kernel matrix using Parzen windowing. Then, only the eigenvectors contributing the most entropy to the input data are selected as the nonlinear features extracted in RKHS. These distinguished characteristics make KECA superior to KPCA in image classification and data transformation (Gomez-Chova et al., 2011; Izquierdo-Verdiguier et al., 2016; Bai et al., 2020b). Despite the theoretical appeal, KECA suffers from low efficiency of representation and high computational complexity, as it engages in computing entropic components of the kernel matrix. This leads to the relatively poor performance in computer vision tasks when using KECA as compared to CNN- or CKN-based methods, since KECA cannot penetrate deep for finer feature extraction.

To address the aforementioned problems in ocean SAR scene classification using machine learning methods, we propose an efficient multistage CKN, namely KECANet, to automatically classify several typical geophysical phenomena observed in the sea. In every stage of KECANet, every patch is explicitly mapped to a RKHS or feature space using a kernel approximation technique called the pivoted Cholesky decomposition (Harbrecht et al., 2012). Thus, each patch can be projected on a nonlinear subspace using Nyström-type low-rank KECA as convolution filters. An adjacent layer with mean pooling was also added to average the filter responses. Then, multiple convolution layers can be used to learn the high-level features of oceanic phenomena. At the end of all the stages, similar to PCANet, we use binary hashing in combination with block-wise histograms to capture nonlinear information and encode the features. KECANet was evaluated in image classification tasks using a hand-annotated SAR dataset (Wang et al., 2019a) in terms of the detection of 10 geophysical phenomena that often occur on the ocean surface. The experimental results show that KECANet outperforms traditional and state-of-the-art methods in terms of classification accuracy. The code was made publicly available for result reproduction.

The main contributions of this work are as follows.

1. In comparison to CKNs, KECANet is structurally more similar to CNNs but can be trained with the goal of retaining the maximum Rényi entropy of the input data.

In contrast to CNNs, KECANet has fewer hyperparameters and layers, making training much easier.

2. In each stage of the network, KECANet can capture the nonlinear structure of the patches, unlike PCANet, which utilizes linear PCA. Additionally, KECANet can achieve more accurate classification results using less memory than PCANet (please refer to section 3).
3. To the best of our knowledge, this is the first time KECA has been utilized in the context of deep learning methods. Additionally, it can mitigate the high computational complexity often required for kernel matrix evaluation.
4. KECANet is not only robust in handling image noise, but it can obtain the regional and seasonal statistics of different geophysical phenomena (e.g., sea ice) and help evolve certain aspects of numerical ocean models.

The remainder of this paper is divided into the following three sections: data description and the design of the proposed method, the analysis of the experimental results, and the concluding remarks.

## 2 Material and methods

In this section, to facilitate understanding of KECANet, the datasets utilized in this study are first described. Thereafter, we review several basic concepts related to KECA. Finally, we present a detailed description of the ocean SAR scene classification process of the new network.

### 2.1 Data illustration

Since SEASAT in 1978, several satellite missions, including ERS-1/2 (1991–2003), RADARSAT-1/2 (1995–present), TerraSAR-X (2007–present), Sentinel-1 (S-1) (2014–present), and Gaofen (GF)-3 (2016–present), have provided ever-improving ocean SAR images. The S-1 mission receives the most attention as it is the only satellite that provides publicly available routine SAR wave mode (WV) measurements at the global scale (Torres et al., 2012; Potin et al., 2016; Bioresita et al., 2018; Bjerreskov et al., 2021). Additionally, its production is characterized by the outstanding characteristics of a large scene footprint and a fine spatial resolution. Therefore, S-1 SAR data were adopted in this paper to implement global-scale classification of an ocean SAR scene.

#### 2.1.1 Sentinel-1 wave mode

WV is the default imaging mode operated by the microwave SAR instruments on S-1 satellites, which are two polar-orbiting

and sun-synchronous, namely, S-1A and S-1B. WV obtains small SAR image scenes (defined as imagettes) at two alternating center incidence angles of 23° (WV1) and 36.5° (WV2). Both angles often apply in linear vertical (VV) image transmission and polarization receive, although certain horizontal (HH) images have been acquired during special phases. The size of each imagette is 20 × 20 km with a 5 m spatial resolution. This paper mainly focuses on S-1A WV data, as S-1B SAR data essentially have equivalent characteristics (Wang et al., 2019b).

### 2.1.2 TenGeoP-SARwv dataset

Wang et al. (2019a) constructed an ocean SAR dataset that includes more than 37,000 S-1 20 × 20 km imagettes, namely, the TenGeoP-SARwv dataset, which is of great use for scholars and users when modeling in geoscience, remote sensing, and deep learning applications. Specifically, the total number of imagettes in this dataset is 16,068 and 21,485, which were obtained under the WV1 and WV2 modes, respectively. Each imagette in this dataset is labeled corresponding to 10 atmospheric or oceanic phenomena (please refer to Figure 1): atmospheric fronts (AF); biological slicks (BS); icebergs (IB); low-wind areas (LWA); microconvective cells (MCC); oceanic fronts (OF); pure ocean waves (PW); rain cells (RC); sea ice (SI); and windstreaks (WS). Additionally, data quality control was implemented according to the following two steps: first, the raw products were converted into the Normalized Radar Cross Section (NRCS) by applying the nominal calibration method proposed by the European Space Agency; second, the NRCS was recalibrated to reduce the incidence angle effect with the CMOD5n model function. The dataset was made publicly available at <http://www.seanoe.org/data/00456/56796/>

provided by the French Research Institute for Exploitation of the Sea (IFREMER). Figure 2 shows the distribution of the TenGeoP-SARwv dataset in a 5 × 5° global spatial grid. As can be seen, the data coverage is nearly complete over the Pacific, Indian, and South Atlantic oceans. However, the SAR image density in each region is relatively low considering that the largest number of WV images is smaller than 50. Carrying out geophysical phenomena classification is made difficult by this lack of data.

## 2.2 Kernel entropy component analysis

As the name implies, KECA attempts to select the most entropic components instead of variance-based or principal components in KPCA to extract features of interest. As is well known, the concept of Rényi quadratic entropy is as follows:

$$H(p) = -\log \int p^2(x)dx \quad (1)$$

where  $p(x)$  denotes the probability density function. Equation (1) can be regarded as the measure of information derived from the given input images  $X = [x_1, \dots, x_N]$  ( $x_i \in \mathbb{R}^D$ )

in KECA (Jenssen, 2009). Considering the monotonic property of logarithmic function, Equation (1) can be simplified as follows:

$$V(p) = \int p^2(x)dx \quad (2)$$

Using a kernel  $k_\sigma(x, x_i)$  of the Parzen window density estimator combined with the width coefficient  $\sigma$  (Jenssen, 2009), Equation (2) is converted to

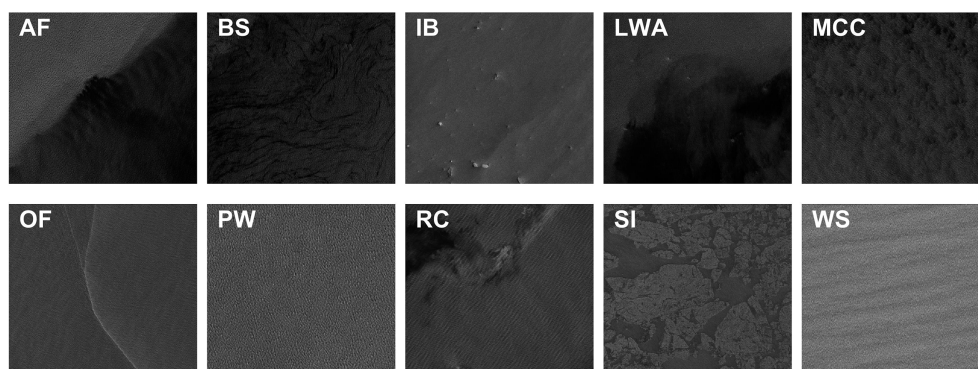


FIGURE 1

Description of the TenGeoP-SARwv dataset. Ten examples of defined geophysical phenomena: atmospheric fronts (AF); biological slicks (BS); icebergs (IB); low-wind areas (LWA); microconvective cells (MCC); oceanic fronts (OF); pure ocean waves (PW); rain cells (RC); sea ice (SI); and windstreaks (WS).

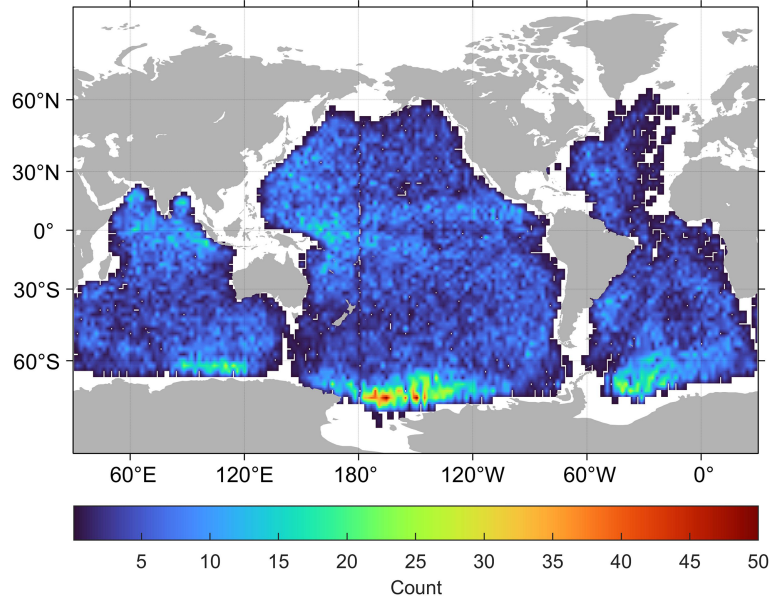


FIGURE 2 SAR data coverage under VV polarization in 2016. The color represents the number of VV images within each 5° by 5° spatial bin.

$$\begin{aligned}
 V(p) &\approx \hat{V}(p) \\
 &= \frac{1}{N} \sum_{x \in X} p(x) \\
 &= \frac{1}{N} \sum_{x_i \in X} \frac{1}{N} \sum_{x_j \in X} k_{\sigma}(x_i, x_j) \\
 &= \frac{1}{N^2} \mathbf{1}^T K \mathbf{1}
 \end{aligned} \tag{3}$$

where  $K = \{K_{ij} | K_{ij} = k_{\sigma}(x_i, x_j)\}$  is the kernel matrix and  $\mathbf{1}$  is an  $N$ -dimensional vector containing all ones. Employing eigendecomposition (Jenssen, 2009)

$$K = AA^T = (WD^{\frac{1}{2}})(D^{\frac{1}{2}}W^T) \tag{4}$$

Equation (9) can be transformed into

$$\hat{V}(p) = \frac{1}{N^2} \sum_{i=1}^N \left( \sqrt{\lambda_i} \mathbf{1}^T \omega_i \right)^2 \tag{5}$$

where the eigenvalues  $\lambda_1, \dots, \lambda_N$  corresponding to  $\omega_1, \dots, \omega_N$  is reserved as the entropy estimate. In another word, KECA performs feature extraction not by the eigenvectors associated with the top  $N_c$  eigenvalues but by the axes contributing most to the Rényi entropy estimate  $\hat{V}(p)$  (Jenssen, 2009; Bai et al., 2020a). This can be concluded as the most distinct difference between KECA and KPCA (Jenssen, 2009).

### 2.3 KECANet for ocean SAR scene classification

In the interest of easy understanding for the reader, we first present the main flowchart of an ocean SAR scene classification based on KECA in Figure 3, whose main steps can be summarized as follows. 1) The SAR imageries are preprocessed using a batch normalization technique and are then used as input into KECANet. 2) Noting that KECANet consists of multiple stages of successive convolutional layers and pooling layers, an approximation of KECA by the pivoted Cholesky decomposition (Harbrecht et al., 2012) is adopted for learning the filters to extract features in its convolutional layers. 3) The features of different imageries are introduced to a final stage of spectral histogram (SH) encoding. Then, the output of classification results can be accomplished by the most naive nearest neighbor classifier with cosine similarity scores. The internal structure of KECANet is also shown in Figure 4 for experimental reproduction. In the following sections, we describe the entirety of the process in detail.

#### 2.3.1 Data preprocessing

Assuming that  $N$  ocean SAR imageries  $\{X_i^{(0)} \in \mathbb{R}^{m_0 \times n_0 \times C_0}\}_{i=1}^N$  derived from the TenGeoP-SARwv dataset penetrate deep into the  $l$ th layer of KECANet, where the channel  $C_0 = 1$  denotes



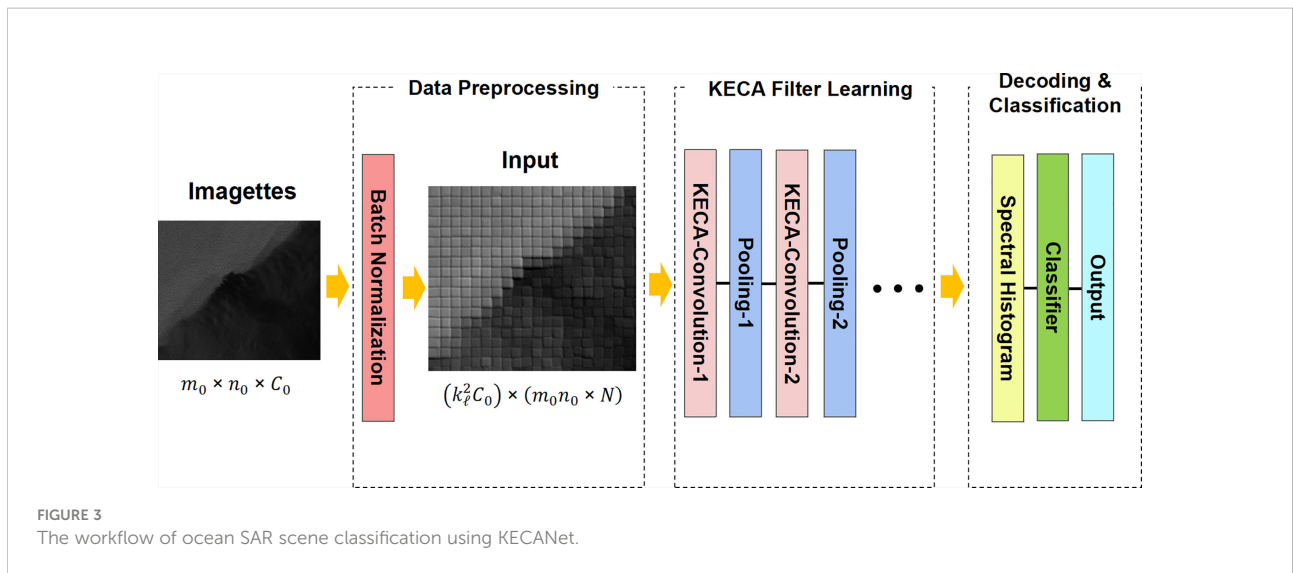


FIGURE 3 The workflow of ocean SAR scene classification using KECANet.

the grayscale input, the overlapping (stride-  $S_\ell$ ,  $S_\ell = 1$  adopted in our work) patches of  $k_\ell \times k_\ell$  are extracted from each  $X_i^{(0)}$ . We subtract batch normalization from each patch using the z-score technique and obtain a cluster of  $\{\bar{X}_i^{(0)} \in \mathbb{R}^{(k_p^2 C_0) \times (m_0 n_0 \times N)}\}_{i=1}^N$ , which is fed into the following chain of convolution and pooling operations.

### 2.3.2 KECA filter learning

$\{\bar{X}_i^{(0)} \in \mathbb{R}^{(k_p^2 C_0) \times (m_0 n_0 \times N)}\}_{i=1}^N$   $D_\ell$ -dimensional RKHS using the Equations (3)–(5). Thereafter, the  $\ell$  th layer can obtain the feature maps from the preceding  $(\ell-1)$  th layer

$$\bar{X}_i^{(\ell)} = \beta(\bar{X}_i^{(\ell-1)} * W^{(\ell)}) \in \mathbb{R}^{m_{\ell-1} \times n_{\ell-1} \times C_\ell} \quad (6)$$

where  $*$  represents a convolution operator and  $\beta(\cdot)$  denotes the nonlinear mean pooling with a pooling window of  $\theta_\ell \times \theta_\ell$  such that  $m_\ell = (m_{\ell-1} - \theta_\ell) / S_\ell + 1$  and  $n_\ell = (n_{\ell-1} - \theta_\ell) / S_\ell + 1$ .  $W^{(\ell)} \in \mathbb{R}^{(k_p^2 C_{\ell-1}) \times C_\ell}$  is defined as the layerwise filter ensemble with  $c_\ell$

KECA filters prelearned from the  $(\ell-1)$  th layer. However, there are several limitations in directly using the KECA algorithm, e.g., too many computations when calculating the kernel matrix (as many as  $O(N^2)$ , where  $N$  is often more than 1K) and low efficiency of the representation as compared to deep learning methods (Bengio and LeCun, 2007). Hence, we provide an efficient algorithm (please refer to Algorithm 1) to obtain a low-rank approximation of the kernel matrix, while effectively maintaining the outstanding performance of KECA. The key to the mathematical analysis of Algorithm 1 is illustrated as follows. Let  $K_i^{(\ell)} \in \mathbb{R}^{(m_\ell n_\ell) \times (m_\ell n_\ell)}$  be a kernel matrix derived from  $\bar{X}_i^{(\ell)}$ . In order to save time and memory when obtaining  $K_i^{(\ell)}$ , on the basis of Equation (4), we carry out the eigendecomposition on a smaller feature space, i.e., of size  $r_\ell \ll m_\ell n_\ell$ . Then, we expand the results back up to  $m_\ell n_\ell$  dimensions. In other words, our intention here is to seek a Nyström-type low-rank approximation  $\tilde{K}_i^{(\ell)}$  of  $K_i^{(\ell)}$  that satisfies the approximation error  $\|E_i^{(\ell)}\| = \|K_i^{(\ell)} - \tilde{K}_i^{(\ell)}\| \leq \varepsilon$ . Finally, after substituting

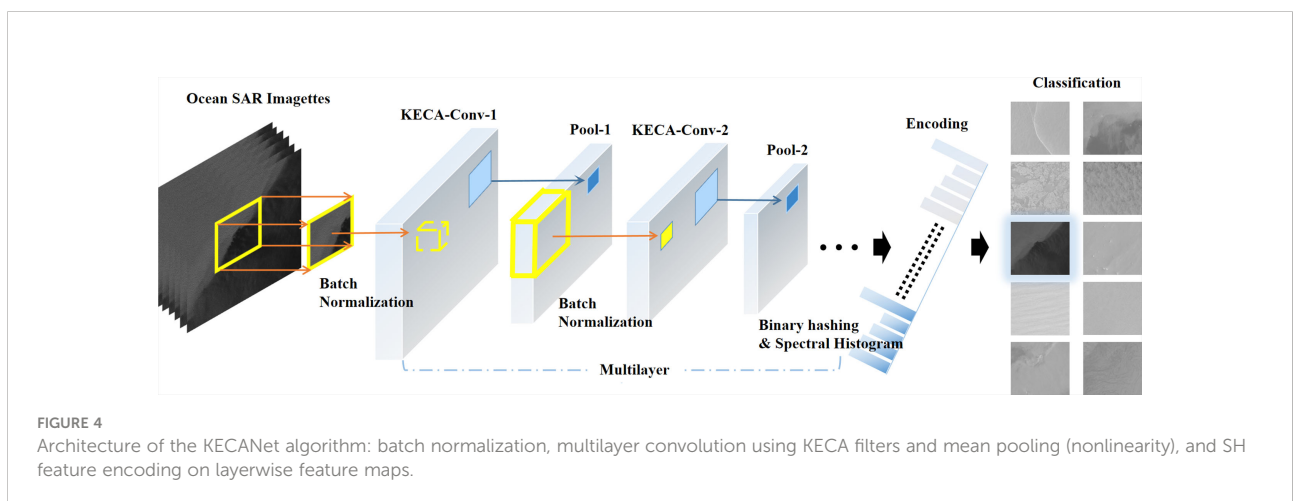


FIGURE 4 Architecture of the KECANet algorithm: batch normalization, multilayer convolution using KECA filters and mean pooling (nonlinearity), and SH feature encoding on layerwise feature maps.

$\tilde{K}_i^{(\ell)}$  into Equations (4) and (5), we can obtain the KECA learning filters  $w^{(\ell)}$  from the  $(\ell - 1)$  th layer for feature extraction.

**Lemma 1.** The low-rank approximation  $\tilde{K}_i^{(\ell)}$  derived from the Algorithm 1 satisfies  $\|E_i^{(\ell)}\| = \|K_i^{(\ell)} - \tilde{K}_i^{(\ell)}\| \leq \epsilon$ , where  $\epsilon$  denotes a minimum value.

**Proof of Lemma 1.** For simplicity, let  $\tilde{K}$  and  $K_i^{(\ell)}$  represent  $\tilde{K}_i^{(\ell)}$  and  $K$  with  $N_\ell = m_\ell n_\ell$ , respectively. Considering that  $\tilde{K}_j$  obtained in the  $j$  th iteration is a positive semidefinite matrix (Harbrecht et al., 2012),  $\tilde{K}_j$  can be decomposed by the Cholesky technique (Harbrecht et al., 2012; Zhou, 2015).

$$\tilde{K}_j = K^{N_j \times j} (K^{j \times j})^{-1} (K^{N_j \times j})^T = P_j (P_j)^T \quad (7)$$

Without loss of generality, we assume

$$\tilde{K}_{j+1} = K^{N_j \times (j+1)} (K^{(j+1) \times (j+1)})^{-1} (K^{N_j \times (j+1)})^T = P_{j+1} (P_{j+1})^T \quad (8)$$

On the basis of Cholesky factorization:

$$K^{j \times j} = L_j L_j^T \quad (9)$$

where  $L_j$  is a lower triangular matrix. Then, substituting Equation (9) into Equation (7), we obtain

$$P_j = K^{N_j \times j} (L_j)^{-T} \quad (10)$$

Motivated by the pivoted Cholesky decomposition (Harbrecht et al., 2012; Zhou, 2015), let

$$K^{(j+1) \times (j+1)} = \begin{bmatrix} K_{11}^{(j+1)} & K_{12}^{(j+1)} & K_{13}^{(j+1)} & \dots & K_{1n}^{(j+1)} \\ K_{21}^{(j+1)} & K_{22}^{(j+1)} & K_{23}^{(j+1)} & \dots & K_{2n}^{(j+1)} \\ K_{31}^{(j+1)} & K_{32}^{(j+1)} & K_{33}^{(j+1)} & \dots & K_{3n}^{(j+1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ K_{n1}^{(j+1)} & K_{n2}^{(j+1)} & K_{n3}^{(j+1)} & \dots & K_{nn}^{(j+1)} \end{bmatrix} = (L_{j+1})^T L_{j+1} \quad (11)$$

where  $K_t^{j \times 1} = (K_{1t}, \dots, K_{jt})^T$ , i.e., the  $t$ th kernel column. Using Equations (9)–(11), we can obtain Step 4 in Algorithm 1:

$$L_{j+1} = \begin{bmatrix} L_j & 0 \\ u^T & \mu \end{bmatrix} \quad (12)$$

where  $u = (L_j)^{-1} K_t^{j \times 1}$  is exactly the  $t$ th column of  $P_j$  and  $\mu = \sqrt{K_{tt} - u^T u}$ . Let

$$P_{j+1} = [P_j P_j] \quad (13)$$

On the basis of Equations (8) and (12), we obtain

$$P_{j+1} = K^{N_j \times (j+1)} L_{j+1}^{-T} = \begin{bmatrix} P_j \mu^{-1} (K_t^{N_j \times 1} L_j u) \end{bmatrix} \quad (14)$$

Therefore, the approximation error in iterations is

$$E_{j+1} = \tilde{K}_j - \tilde{K}_{j+1} = P_j (P_j)^T = E_j - p_j^2 \quad (15)$$

which means that the error is monotonically decreasing until it meets the error tolerance  $\epsilon$ .

For convenient computing, the trace norm is chosen in Step 9 as the error measurement in Algorithm 1.  $\square$

### Algorithm 1.

**Input:** the normalized patches  $\{\tilde{X}_i^{(0)} \in \mathbb{R}^{(k^2 C_0) \times (m_0 n_0 \times N)}\}_{i=1}^N$ , the error tolerance  $\epsilon$  and the number  $r$  of the pivot elements. 1: Let approximation error

1: Let approximation error  $e = \|d_1\| = \|[K_{11}, \dots, K_{N_1, N_1}]\|^T$ ,  $N_1 = 1 : N_1$

2: **While**  $j = 1 \geq r$  &&  $e_j > \epsilon$ :

3: seek  $t = \arg \max_{i \in N} j d_{ji}$ , where  $d_{ji}$  denotes the  $i$ th element of  $d_j$

4: **if**  $j=1$  **then**

5:  $P_{j+1} = p = K_t^{N_1 \times 1} / \sqrt{K_{tt}}$

6: **else**

7:  $P_{j+1} = [P_j p_j]$ , where  $p_j = \mu^{-1} (K_t^{N_1 \times 1} - L_j u)$ ,  $u = (L_j)^{-1} K_t^{j \times 1}$ ,  $\mu = \sqrt{K_{tt} - u^T u}$

8: **end if**

9: let  $N_{j+1} = N_j \setminus t$ , then  $d_{j+1,i} = d_{ji} - p_j^2$  ( $i \in N_{j+1}$ ),  $e_j = \sum_{i \in N_{j+1}} d_{j+1,i}^2$

10:  $j = j + 1$

11: **end while**

**Output:** the low-rank approximation  $\tilde{K}_i^{(\ell)} = P^T P$

### 2.3.3 Encoding and classification

After the aforementioned stages, involving the convolutional, mean pooling, and KECA layers, a binary hashing similar to PCANet's is employed in KECANet to compute the block-wise histograms in encoding stage. Specifically, we first binarize the outputs of  $(X^{(\ell-1)} * W^{(\ell)}) \in \mathbb{R}^{m_{\ell-1} \times n_{\ell-1} \times C_\ell}$  before pooling. The hashed feature maps are divided into  $C_\ell / C_\lambda$  (pre-fixed  $C_\lambda$  to 8 after fine tuning) different sets and each set is encoded with a  $C_\ell$  bit integer. Secondly, we can obtain

$$S_t^{(\ell)} = \sum_{c=1}^{C_\lambda} \text{unit} \left( \bar{X}^{(\ell-1)} * (W^{(\ell)})_{C_\lambda(t-1)+c} \right) 2_t^{c-1} \quad (16)$$

where  $t = 1, \dots, H_\ell$  and  $\text{unit}(\cdot)$  represents the unit step function (Low et al., 2017). Thirdly,  $S_t^{(\ell)}$  is clustered into  $B_\ell$  nonoverlapping blocks, each of whom is histogrammed into  $2^{C_\lambda}$  bins. Thus, in each layer,  $H_\ell B_\ell$  blockwise histograms are cascaded to form the final feature vector of the input SAR imagette. Finally, the feature vectors derived from KECANet can be trained by the most naive nearest neighbor classifier with Cosine similarity scores and achieve ocean scene recognition.

### 2.4 Computational complexity

In the data preprocessing stage, the main cost of z-score normalization is  $O(Nm_0n_0)$  computations. In the KECA filter learning stage, the complexity of Algorithm 1 is  $O(r^2N)$ . This is very efficient as compared to  $O(N^3)$ , which is required by the conventional kernel method ( $r \leq 9$ ). The eigendecomposition of approximation kernel matrix needs  $O(D_\ell^3)$  computations. The cost of mapping the patches to the RKHS space is  $O(D_\ell^2N)$ . In the

final stage, the binary hashing and block-wise histograms require  $O(2D_\ell N)$  and  $O(H_\ell B_\ell D_\ell N \log_2)$ . Considering  $N \gg \max(m_\ell, n_\ell, D_\ell, H_\ell, B_\ell)$ , the overall cost of KECANet is presented as follows:

$$O\left(\sum_{\ell=1}^L N m_\ell n_\ell + \sum_{\ell=1}^L N (r^2 + D_\ell^2)\right) \tag{17}$$

which can be simplified as (due to  $D_\ell \gg r$ )

$$O\left(\sum_{\ell=1}^L N (D_\ell^2 + m_\ell n_\ell)\right) \tag{18}$$

Considering that the complexity of a two-stage PCANet is  $O(\sum_{\ell=1}^2 N (m_\ell n_\ell)^2)$  (Chan et al., 2015) and  $D_\ell / m_\ell n_\ell \ll 1$  for SAR imagery, the complexity of KECANet is lower than PCANet, which requires far fewer computations than CKNs (Mairal et al., 2014; Mohammadnia-Qaraei et al., 2018; Qaraei et al., 2021). In the next section, we demonstrate the potential and performance of KECANet in detail.

### 3 Experiment and results

This section starts with a detailed ablation study to analyze the contributions of deep learning architecture and the pivoted Cholesky decomposition to a conventional KECA method. Afterward, considering that deep kernelized methods are greatly impacted by the number of layers and the width coefficient, we implemented a sensitivity analysis on these two important coefficients using the TenGeoP-SARwv dataset. Then, another experiment on noise-filled SAR imagettes was also performed to test the noise robustness of KECANet. To further validate the model performance, a geophysical map of sea ice in Antarctica, as detected by KECANet, is analyzed and compared with the global gridded product. Finally, we compared KECA with several different methods and presented state-of-the-art results for undertaking the task of ocean SAR scene auto-classification. For the training strategies in this section, we adopted the set proposed by Wang et al. (2019b). A total of 320 imagettes per class ( $N = 320$ ) were randomly chosen as the input dataset from the annotated TenGeoP-SARwv dataset separated by WV1 and WV2. Notably, the size of the input was relatively small. Moreover, 70% and 30% of the dataset were selected as the training and testing subsets, respectively. All of the experiments without data augmentation were implemented by MATLAB R2021b on a server with GN7.2XLARGE32 processors, 32 GB memory, and a Windows Server 2016 operating system. All results were averaged over 10 runs. The reported values of the recall (R), precision (P), and F-score (F) were averaged over 10 repeated experiments. The values of these three parameters were all expected to approach one (Wang et al., 2019b). KECANet was implemented using the parameter configuration shown in Table 1.

TABLE 1 KECANet parameter configuration for the TenGeoP-SARwv database.

Parameters	Values
$X_i^{(0)}$	$299 \times 299$
$N$	320
$\max(\ell)$	3
$k_\ell$	8
$\theta_\ell$	3
$C_\ell$	$8 \times 16 \times 16$
$D_\ell$	64
$r$	64
$H_\ell$	128
$B_\ell$	8
$\sigma$	$1 \times 10^{-3}$
$k_\sigma(x, x_i)$	$(2\pi\sigma^2)^{-1/2} \exp(-\ x - x_i\  / (2\sigma^2))$

### 3.1 Ablation study

Table 2 shows the classification results of various variants of KECA on the test set of the TenGeoP-SARwv dataset. Noting that KECA is the original method (Jenssen, 2009), *KECANet-1* and *KECANet-2* denote a KECA, which has penetrated deep into the 1st and 2nd layers, respectively, without using the accelerating technique of the pivoted Cholesky decomposition. *Low-rank KECANet-1* and *Low-rank KECANet-2* integrate the deep learning architecture and the accelerating technique with the conventional KECA (Jenssen, 2009). Table 2 empirically validates the effectiveness of using deep learning methods, considering that *KECANet-1* improves the averaged classification result of KECA by 6.48%, 23.25%, and 12.96% in R, P, and F statistics, respectively. Table 2 also shows that using the accelerating technique can significantly reduce computing time and memory since *KECANet-2* was so prone to exhausting its memory that it severely impeded its functionality. Compared to the original KECA, we progressively improved classification accuracy in R, P, and F statistics by 46.21%, 44.99%, and 46.53%, respectively.

### 3.2 Impact of the number of layers

We evaluated KECANet's performance on the selected dataset using different numbers of layers. The KECA filter ensemble  $C_\ell$  of  $8 \times 8 \times 16 \times 16$  was selected in layer 4. Table 3 illustrates the overall performance for multilayer KECANet. It can be observed that KECANet's average performance regarding R, P, and F increased from 58.99%, 56.63%, and 56.87% in layer 1, to 86.40%, 84.73%, and 84.95% in layer 4. Considering a tradeoff between computing time and classification performance (the advantage of KECANet-4 is



TABLE 2 Ablation Study of KECA on WV1 (upper) and WV2 (lower) imagette recognition (%).

	KECA			KECANet-1			Low-rank KECANet-1			KECANet-2			Low-rank KECANet-2		
	R	P	F	R	P	F	R	P	F	R	P	F	R	P	F
AF	74.3	87.5	80.4	66.1	65.7	65.9	64.7	65.3	65.0	N/A	N/A	N/A	84.4	89.8	87.0
	66.3	67.7	67.0	66.6	85.9	75.0	66.3	85.5	74.7	N/A	N/A	N/A	91.8	86.3	88.9
BS	65.1	85.4	73.9	57.9	65.7	61.5	56.4	65.3	60.5	N/A	N/A	N/A	76.2	89.8	82.4
	43.2	65.6	52.1	47.0	89.3	61.6	46.8	88.8	61.3	N/A	N/A	N/A	72.2	89.6	80.0
IB	35.1	27.1	30.6	56.3	55.7	56.0	54.8	55.3	55.1	N/A	N/A	N/A	74.5	79.8	77.1
	25.8	16.7	20.3	41.7	75.9	53.9	41.5	75.5	53.5	N/A	N/A	N/A	66.9	76.3	71.3
LWA	51.2	44.8	47.8	65.8	44.0	52.7	64.3	43.6	52.0	N/A	N/A	N/A	84.1	68.1	75.3
	51.8	45.8	48.6	60.9	79.3	68.9	60.7	78.8	68.6	N/A	N/A	N/A	86.1	79.6	82.7
MCC	41.7	52.1	46.3	68.3	50.7	58.2	66.9	50.3	57.4	N/A	N/A	N/A	86.6	74.8	80.3
	81.9	89.6	85.6	66.0	77.6	71.4	65.8	77.2	71.0	N/A	N/A	N/A	91.2	77.9	84.1
OF	34.9	30.2	32.4	61.8	65.7	63.7	60.4	65.3	62.7	N/A	N/A	N/A	80.1	89.8	84.7
	63.7	60.4	62.0	66.8	89.3	76.4	66.5	88.8	76.1	N/A	N/A	N/A	92.0	89.6	90.8
PW	37.8	17.7	24.1	38.7	65.7	48.7	37.2	65.3	47.4	N/A	N/A	N/A	56.9	89.8	69.7
	25.8	17.7	21.0	40.5	85.9	55.0	40.2	85.5	54.7	N/A	N/A	N/A	65.7	86.3	74.6
RC	87.7	96.9	92.1	72.2	50.7	59.6	70.8	50.3	58.8	N/A	N/A	N/A	90.5	74.8	81.9
	95.8	95.8	95.8	72.0	80.9	76.2	71.8	80.5	75.9	N/A	N/A	N/A	97.2	81.3	88.5
SI	65.9	56.3	60.7	59.7	50.7	54.8	58.3	50.3	54.0	N/A	N/A	N/A	78.0	74.8	76.4
	53.0	45.8	49.2	53.5	72.6	61.6	53.2	72.2	61.3	N/A	N/A	N/A	78.7	72.9	75.7
WS	44.9	59.4	51.1	57.6	55.7	56.6	56.2	55.3	55.7	N/A	N/A	N/A	75.9	79.8	77.8
	48.4	64.6	55.4	46.7	82.6	59.6	46.4	82.2	59.3	N/A	N/A	N/A	71.9	82.9	77.0
AVE	53.9	55.7	53.9	60.4	57.0	57.8	59.0	56.6	56.9	N/A	N/A	N/A	78.7	81.1	79.2
	55.6	57.0	55.7	56.2	81.9	66.0	55.9	81.5	65.6	N/A	N/A	N/A	81.4	82.3	81.4
Max(std)	0.81	0.93	0.86	1.92	1.64	1.72	1.93	1.68	1.73	N/A	N/A	N/A	1.56	1.46	1.48
	0.85	0.98	0.91	2.10	1.77	1.91	2.13	1.78	1.92	N/A	N/A	N/A	1.88	1.57	1.64
Time		0.8h			2.7h			0.4h			N/A			18.1h	

AVE, Max(std), and Time denote the average function, the maximum standard deviation, and computing time, respectively.

almost negligible as compared to KECANet-3), the maximum number of layers, i.e.,  $\max(\ell) = 3$ , was adopted in this study. KECANet-3 is denoted as KECANet for simplicity in the rest of the article unless otherwise stated.

### 3.3 Impact of the width coefficient

The width coefficient, also known as the kernel width  $\sigma$ , plays a role in kernel-based methods (Bai et al., 2019; Bai et al., 2020a). We evaluated the impact of the width of KECANet on the selected dataset. The average results of classifying 10 geophysical phenomena are shown in Figure 5. The width coefficients of three layers were the same and ranged from  $1 \times 10^{-4}$  to 1. The results show that the classification performances increase significantly as a result of properly choosing the width coefficient. Therefore,  $1 \times 10^{-3}$  was adopted in this study. More details are shown in Table 4.

### 3.4 KECANet for denoising

In this subsection, we evaluate the noise robustness of KECANet. We randomly selected 200 imagettes of each oceanic phenomenon and added block noise to the input datasets. All noise insertions accounted for 0.01 to 0.15 of the image area and were located within a rectangle with randomly distributed positions (please refer to Figure 6). Thus, we established a new gallery split into two subsets for the classification experiment. One subset was used for training, including 140 noisy imagettes and 84 noise-free imagettes per class. The other subset contained the remainder for testing. We ran this process 10 times. Table 5 shows the classification results versus those derived from KECANet on the original dataset. By comparative analysis, we can conclude that KECANet can provide relatively acceptable robustness in SAR data denoising, considering it obtained more than 82% in R, P, and F statistics, even though the classification performances deteriorated by about 1–2% in calibration metrics.

TABLE 3 Classification performance (%) in WV1 (upper) and WV2 (lower) imagette recognition for KECANet using different layers.

	KECANet-1			KECANet-2			KECANet-3			KECANet-4		
	R (%)	P (%)	F (%)	R (%)	P (%)	F (%)	R (%)	P (%)	F (%)	R (%)	P (%)	F (%)
AF	64.65	65.30	64.97	84.37	89.80	87.00	91.80	93.33	92.56	92.07	93.40	92.73
	66.30	85.49	74.68	91.78	86.25	88.93	94.55	86.67	90.43	94.58	86.97	90.62
BS	56.43	65.30	60.54	76.15	89.80	82.42	83.58	93.33	88.19	83.84	93.40	88.36
	46.76	88.82	61.26	72.23	89.58	79.98	75.00	90.00	81.82	75.04	90.30	81.97
IB	54.82	55.30	55.06	74.54	79.80	77.08	81.97	83.33	82.64	82.23	83.40	82.81
	41.46	75.49	53.52	66.93	76.25	71.29	69.70	76.67	73.02	69.74	76.97	73.18
LWA	64.34	43.63	52.00	84.06	68.14	75.26	91.49	71.67	80.37	91.75	71.73	80.51
	60.65	78.82	68.55	86.12	79.58	82.72	88.89	80.00	84.21	88.93	80.30	84.40
MCC	66.85	50.30	57.40	86.57	74.80	80.26	94.00	78.33	85.45	94.26	78.40	85.60
	65.76	77.16	71.00	91.23	77.92	84.05	94.00	78.33	85.45	94.04	78.64	85.65
OF	60.35	65.30	62.73	80.07	89.80	84.66	87.50	93.33	90.32	87.76	93.40	90.49
	66.49	88.82	76.05	91.97	89.58	90.76	94.74	90.00	92.31	94.78	90.30	92.49
PW	37.22	65.30	47.41	56.94	89.80	69.69	64.37	93.33	76.19	64.63	93.40	76.40
	40.18	85.49	54.67	65.65	86.25	74.55	68.42	86.67	76.47	68.46	86.97	76.61
RC	70.77	50.30	58.80	90.49	74.80	81.90	97.92	78.33	87.04	98.18	78.40	87.18
	71.76	80.49	75.87	97.23	81.25	88.53	100.00	81.67	89.91	100.00	81.97	90.09
SI	58.31	50.30	54.01	78.02	74.80	76.38	85.45	78.33	81.74	85.72	78.40	81.89
	53.24	72.16	61.27	78.71	72.92	75.70	81.48	73.33	77.19	81.52	73.64	77.38
WS	56.18	55.30	55.74	75.90	79.80	77.80	83.33	83.33	83.33	83.60	83.40	83.50
	46.39	82.16	59.29	71.86	82.92	76.99	74.63	83.33	78.74	74.67	83.64	78.90
AVE	58.99	56.63	56.87	78.71	81.14	79.24	86.14	84.67	84.78	86.40	84.73	84.95
	55.90	81.49	65.62	81.37	82.25	81.35	84.14	82.67	82.96	84.18	82.97	83.13
Max(std)	1.93	1.68	1.73	1.56	1.46	1.48	1.13	1.27	1.19	1.09	1.25	1.19
	2.13	1.78	1.92	1.88	1.57	1.64	1.21	1.38	1.23	1.19	1.37	1.22
Time		0.4 h			18.1 h			35.4 h			73.1 h	

AVE, Max(std), and Time denote the average function, the maximum standard deviation, and computing time, respectively.

### 3.5 A geophysical application

To further verify the credibility capabilities of KECANet, the deep learning method was applied to all SAR imagettes

from January to December 2016 in the TenGeoP-SARwv dataset. We examined the geophysical map of detected sea ice comparable to the NOAA/NSIDC Climate Data Record of Passive Microwave Sea Ice Concentration (Version 4,

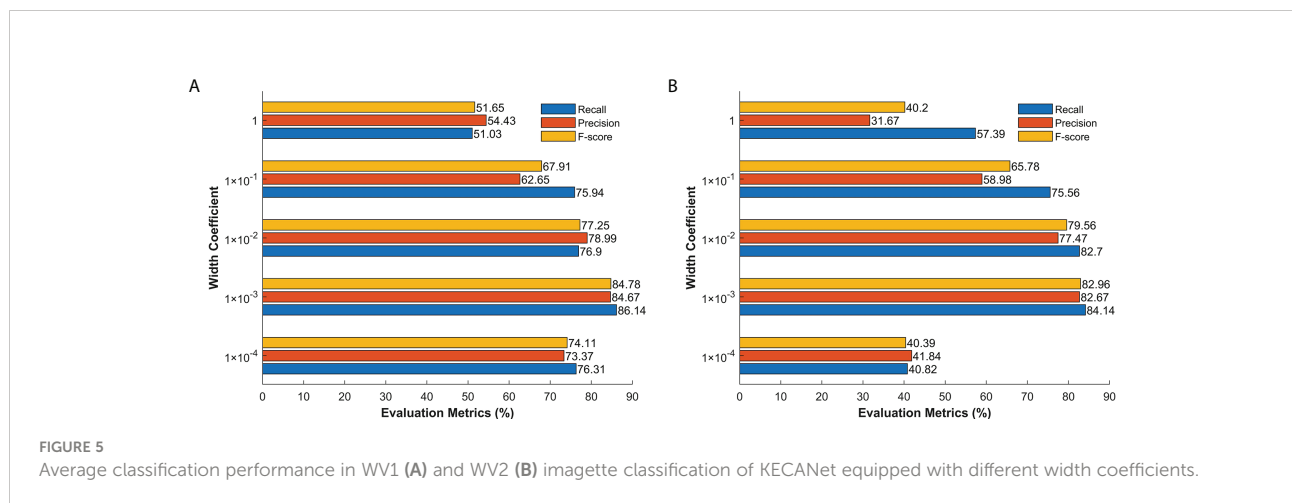


FIGURE 5 Average classification performance in WV1 (A) and WV2 (B) imagette classification of KECANet equipped with different width coefficients.

TABLE 4 Classification performance (%) in WV1 (upper) and WV2 (lower) imagette recognition of KECANet equipped with different width coefficients.

	$1 \times 10^{-4}$			$1 \times 10^{-2}$			$1 \times 10^{-2}$			$1 \times 10^{-1}$			$I$		
	<i>R</i>	<i>P</i>	<i>F</i>	<i>R</i>	<i>P</i>	<i>F</i>	<i>R</i>	<i>P</i>	<i>F</i>	<i>R</i>	<i>P</i>	<i>F</i>	<i>R</i>	<i>P</i>	<i>F</i>
<i>AF</i>	82.0	82.0	82.0	91.8	93.3	92.6	82.6	87.7	85.0	81.6	71.3	76.1	56.7	63.1	59.7
	51.2	45.8	48.4	94.5	86.7	90.4	93.1	81.5	86.9	86.0	63.0	72.7	67.8	35.7	46.7
<i>BS</i>	73.8	82.0	77.7	83.6	93.3	88.2	74.3	87.7	80.5	73.4	71.3	72.3	48.5	63.1	54.8
	31.7	49.2	38.5	75.0	90.0	81.8	73.6	84.8	78.8	66.4	66.3	66.4	48.3	39.0	43.1
<i>IB</i>	72.1	72.0	72.1	82.0	83.3	82.6	72.7	77.7	75.1	71.8	61.3	66.1	46.9	53.1	49.8
	26.4	35.8	30.4	69.7	76.7	73.0	68.3	71.5	69.8	61.1	53.0	56.8	42.9	25.7	32.1
<i>LWA</i>	81.7	60.4	69.4	91.5	71.7	80.4	82.2	66.0	73.2	81.3	49.6	61.6	56.4	41.4	47.8
	45.6	39.2	42.1	88.9	80.0	84.2	87.5	74.8	80.6	80.3	56.3	66.2	62.1	29.0	39.5
<i>MCC</i>	84.2	67.0	74.6	94.0	78.3	85.5	84.8	72.7	78.2	83.8	56.3	67.4	58.9	48.1	52.9
	50.7	37.5	43.1	94.0	78.3	85.5	92.6	73.1	81.7	85.4	54.7	66.7	67.3	27.3	38.9
<i>OF</i>	77.7	82.0	79.8	87.5	93.3	90.3	78.3	87.7	82.7	77.3	71.3	74.2	52.4	63.1	57.2
	51.4	49.2	50.3	94.7	90.0	92.3	93.3	84.8	88.8	86.2	66.3	74.9	68.0	39.0	49.6
<i>PW</i>	54.5	82.0	65.5	64.4	93.3	76.2	55.1	87.7	67.7	54.2	71.3	61.6	29.3	63.1	40.0
	25.1	45.8	32.4	68.4	86.7	76.5	67.0	81.5	73.5	59.8	63.0	61.4	41.7	35.7	38.4
<i>RC</i>	88.1	67.0	76.1	97.9	78.3	87.0	88.7	72.7	79.9	87.7	56.3	68.6	62.8	48.1	54.5
	56.7	40.8	47.5	100.	81.7	89.9	98.6	76.5	86.1	91.4	58.0	71.0	73.3	30.7	43.2
<i>SI</i>	75.6	67.0	71.1	85.5	78.3	81.7	76.2	72.7	74.4	75.3	56.3	64.4	50.3	48.1	49.2
	38.2	32.5	35.1	81.5	73.3	77.2	80.0	68.1	73.6	72.9	49.7	59.1	54.7	22.3	31.7
<i>WS</i>	73.5	72.0	72.8	83.3	83.3	83.3	74.1	77.7	75.8	73.1	61.3	66.7	48.2	53.1	50.5
	31.3	42.5	36.1	74.6	83.3	78.7	73.2	78.1	75.6	66.1	59.7	62.7	47.9	32.3	38.6
<i>AVE</i>	76.3	73.4	74.1	86.1	84.7	84.8	76.9	79.0	77.3	75.9	62.6	67.9	51.0	54.4	51.6
	40.8	41.8	40.4	84.1	82.7	83.0	82.7	77.5	79.6	75.6	59.0	65.8	57.4	31.7	40.2
<i>Max(std)</i>	6.93	6.68	6.73	1.13	1.27	1.19	1.43	1.47	1.45	2.09	2.33	2.15	5.81	5.98	5.92
	7.37	7.46	7.42	1.21	1.38	1.23	1.48	1.59	1.55	2.13	2.35	2.22	6.82	6.87	6.85

*AVE* and *Max(std)* denotes the function of average and the maximum of standard deviation, respectively.

CDRPM-SIC (Meier et al., 2021). From January to December 2016, nearly 5% of SAR imagettes are classified as SI by KECANet. More than 90% of them are distributed near Antarctica due to the satellite coverage and perennial low

temperatures around the polar Southern Ocean. The mapping of classified SI occurrence in Figures 7A–D shows a clear seasonal variability, increasing from early April to a maximum in the Antarctic winter period (from June to

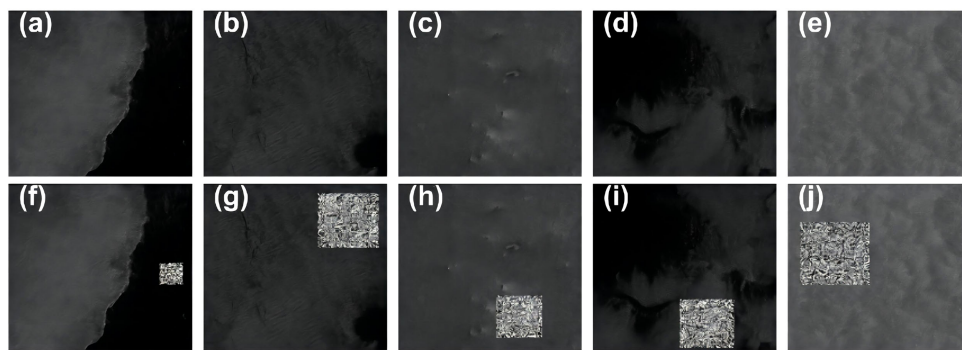


FIGURE 6 The first row (A–E) presents randomly selected 5 imagettes in the TenGeoP-SARwv dataset. The second row (F–J) show the associated noisy imagettes.

TABLE 5 Classification performance (%) in WV1 (upper) and WV2 (lower) imagette recognition for KECANet using noisy and noise-free datasets.

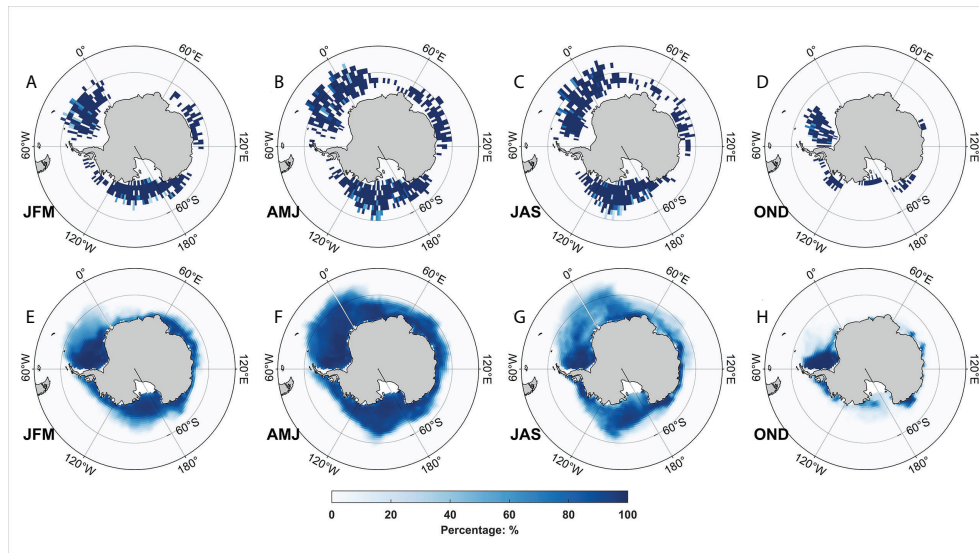
	Noisy			Noise-Free		
	Recall(%)	Precision(%)	F-Score(%)	Recall(%)	Precision(%)	F-Score(%)
AF	90.95	93.07	92.00	91.80	93.33	92.56
	94.36	85.93	89.95	94.55	86.67	90.43
BS	82.81	92.86	87.55	83.58	93.33	88.19
	74.74	89.85	81.60	75.00	90.00	81.82
IB	81.46	82.78	82.11	81.97	83.33	82.64
	69.62	75.95	72.65	69.70	76.67	73.02
LWA	91.35	71.12	79.98	91.49	71.67	80.37
	87.77	79.63	83.50	88.89	80.00	84.21
MCC	93.68	77.93	85.08	94.00	78.33	85.45
	93.25	77.90	84.88	94.00	78.33	85.45
OF	87.00	92.97	89.89	87.50	93.33	90.32
	94.40	89.95	92.12	94.74	90.00	92.31
PW	63.44	92.55	75.28	64.37	93.33	76.19
	67.50	85.95	75.61	68.42	86.67	76.47
RC	97.08	77.82	86.39	97.92	78.33	87.04
	99.71	80.58	89.13	100.00	81.67	89.91
SI	85.04	78.26	81.51	85.45	78.33	81.74
	81.32	72.82	76.84	81.48	73.33	77.19
WS	83.23	82.57	82.90	83.33	83.33	83.33
	73.59	82.27	77.69	74.63	83.33	78.74
AVE	85.60	84.19	84.27	86.14	84.67	84.78
	83.63	82.08	82.40	84.14	82.67	82.96
Max(std)	1.24	1.33	1.24	1.13	1.27	1.19
	1.25	1.44	1.36	1.21	1.38	1.23

AVE and Max(std) denote the average function, the maximum standard deviation, respectively.

October) before decreasing after that. Specifically, in the austral winter (please refer to Figures 7B, C), there is abundant sea ice from 20°E to 60°W and from 100°W to 150°E. Most of it is distributed around Antarctica and poleward of 60°S, or even 54°S. As can be seen in Figure 7D, the summer period sea ice significantly decreases compared to the winter, especially from 120°W to 180°. Sea ice concentration data from the National Snow and Ice Data Center (NSDIC) are also compared in Figures 7E–H. Although the CDRPM-SIC aims at the gridded polar SIC product leading to greatly different resolutions between the first and the second rows of Figure 7, most areas of higher SI occurrence, as detected by KECANet, mirror the high SIC areas. For example, this is especially true for the sea ice extent that expands to the north of 60°S (20°E to 60°W) in the April–June season (see Figures 7B, F). This agreement is another measure of the capability of KECANet as an ocean SAR scene classification tool.

### 3.6 Comparison with state-of-the-art methods

In this subsection, we present the averaged results for the task of ocean SAR scene classification using several well-known models over 10 runs for comparison. The same training strategies with KECANet were adopted here. CMW (Wang et al., 2019b), a CNN-type model, was selected as the baseline as a result of its notable ability in ocean SAR scene classification. Additionally, CKN, as proposed by Mairal et al. (2014) (denoted Gauss-CKN), and RNPCANet (Qaraei et al., 2021) were also employed as state-of-the-art representative models for CKN. We also selected PCANet (Chan et al., 2015) for comparison considering its impressive results in face recognition tasks. As proposed in (Qaraei et al., 2021), for PCANet and RNPCANet, the number of layers was set to 2. The size of both filters and blocks was  $7 \times 7$  and the overlap ratio of the blocks was 0.5. The parameter set for Gauss-CKN was the default set in the code provided by Dr. Mairal, which is available at <https://lear>.



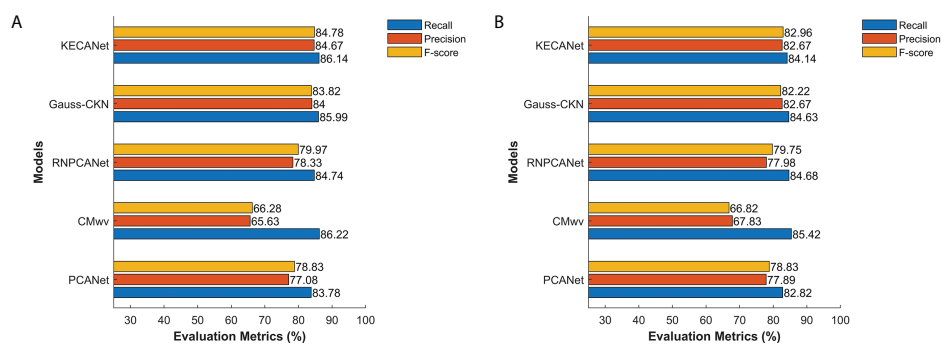
**FIGURE 7**  
Sea ice around the Antarctica from January 2016 to December 2016. Sea ice coverage detected by KECANet in seasons: January-February-March (JFM), April-May-June (AMJ), July-August-September (JAS) and October-November-December (OND) are shown in (A-D) with color denoting the occurrence percentage in 2° × 2° boxes. (E-H) show the mean sea ice concentration from the NSIDC monthly product.

[Inrialpes.fr/people/mairal/solaris/software.html](https://inrialpes.fr/people/mairal/solaris/software.html). CMwv was fine-tuned as suggested in (Wang et al., 2019b). The performances of the aforementioned methods are summarized in Figure 8 including average recall, precision, and F-score statistics. More details can be found in Table 6. Our observations are as follows:

1. KECANet yielded the best classification accuracy considering that F-score statistics are a combined index of both precision and recall statistics, when no data augmentation is applied. This is because the feature extracted by KECA optimally reveals the structure corresponding to the maximum Rényi entropy of the

original SAR data and encodes the discriminant information. Overall, the average recall, precision, and F-score values obtained with KECANet were 86.1% (84.1%), 84.7% (82.7%), and 84.8% (83.0%) higher than the other methods when applied in WV1 (WV2) imagette classification, excepting the recall achieved with Gauss-CKN on WV2;

2. Although KECANet performs marginally better than Gauss-CKN, Gauss-CKN requires more computational resources and time than KECANet;
3. The classification performances of CKN models (i.e., KECANet, Gauss-CKN, and RNPCANet) are superior



**FIGURE 8**  
Average classification performance in WV1 (A) and WV2 (B) imagette classification of different methods.



TABLE 6 Classification performance (%) in WV1 (upper) and WV2 (lower) imagette recognition of different methods.

	PCANet			CMwv			RNPCANet			Gauss-CKN			KECANet		
	R	P	F	R	P	F	R	P	F	R	P	F	R	P	F
AF	69.9	90.4	78.9	93.2	38.3	54.3	70.9	91.7	80.0	64.8	95.0	77.0	91.8	93.3	92.6
	69.0	91.2	78.6	93.2	36.3	52.3	70.8	91.3	79.8	64.8	98.3	78.1	94.5	86.7	90.4
BS	89.2	88.7	89.0	93.2	86.3	89.6	90.2	90.0	90.1	74.7	93.3	83.0	83.6	93.3	88.2
	88.3	89.6	88.9	87.2	89.3	88.3	90.1	89.7	89.9	70.5	91.7	79.7	75.0	90.0	81.8
IB	77.5	58.7	66.8	95.2	14.3	24.9	78.5	60.0	68.0	89.7	86.7	88.1	82.0	83.3	82.6
	76.5	59.6	67.0	90.2	16.3	27.7	78.4	59.7	67.8	90.2	76.7	82.9	69.7	76.7	73.0
LWA	81.3	58.7	68.2	98.2	85.3	91.3	82.2	60.0	69.4	89.8	73.3	80.7	91.5	71.7	80.4
	80.3	59.6	68.4	98.2	77.3	86.5	82.2	59.7	69.1	96.0	80.0	87.3	88.9	80.0	84.2
MCC	93.2	70.4	80.2	78.2	74.3	76.2	94.1	71.7	81.4	90.6	96.7	93.5	94.0	78.3	85.5
	92.2	71.2	80.4	83.2	92.3	87.5	94.1	71.3	81.1	93.4	95.0	94.2	94.0	78.3	85.5
OF	97.3	85.4	91.0	98.2	3.3	6.4	98.2	86.7	92.1	94.8	91.7	93.2	87.5	93.3	90.3
	96.3	86.2	91.0	98.2	3.3	6.4	98.2	86.3	91.9	85.5	98.3	91.5	94.7	90.0	92.3
PW	62.0	97.1	75.7	45.2	98.3	61.9	63.0	98.3	76.8	77.0	78.3	77.7	64.4	93.3	76.2
	61.1	97.9	75.2	37.2	96.3	53.7	62.9	98.0	76.6	80.0	73.3	76.5	68.4	86.7	76.5
RC	89.7	85.4	87.5	91.2	86.3	88.7	90.6	86.7	88.6	86.4	95.0	90.5	97.9	78.3	87.0
	88.7	86.2	87.4	91.2	78.3	84.3	90.6	86.3	88.4	96.5	91.7	94.0	100.	81.7	89.9
SI	87.0	85.4	86.2	88.2	94.3	91.2	88.0	86.7	87.3	94.3	55.0	69.5	85.5	78.3	81.7
	86.1	86.2	86.2	94.2	94.3	94.3	87.9	86.3	87.1	90.0	45.0	60.0	81.5	73.3	77.2
WS	90.7	50.4	64.8	81.2	75.3	78.2	91.7	51.7	66.1	97.8	75.0	84.9	83.3	83.3	83.3
	89.7	51.2	65.2	81.2	94.3	87.3	91.6	51.3	65.8	79.3	76.7	78.0	74.6	83.3	78.7
AVE	83.8	77.1	78.8	86.2	65.6	66.3	84.7	78.3	80.0	86.0	84.0	83.8	86.1	84.7	84.8
	82.8	77.9	78.8	85.4	67.8	66.8	84.7	78.0	79.7	84.6	82.7	82.2	84.1	82.7	83.0
Max(std)	1.14	1.07	1.09	1.09	1.19	1.13	1.09	1.17	1.13	1.02	1.10	1.08	1.13	1.27	1.19
	1.31	1.43	1.38	1.22	1.34	1.26	1.33	1.45	1.38	1.03	1.13	1.10	1.21	1.38	1.23
Time <sup>l</sup>	18.2h			4.4h			18.7h			38.3h			35.4h		

<sup>l</sup> It should be noted that the computing time of PCANet and RNPCANet is just for training two layers, compared to that of KECANet for training three layers. The server are prone to be out of memory when training PCANet and RNPCANet with three layers.

AVE, Max(std), and Time denotes the function of average, the maximum of standard deviation and computing time, respectively.

to those of CNN-type models (i.e., PCANet and CMwv) on ocean SAR imagettes.

the use of GPU acceleration technology to improve classification task efficiency. Another involves leveraging the theoretical interpretation of our convolutional kernel networks to better understand network function.

## 4 Discussion and conclusions

This paper proposes a new deep learning method, KECANet, for combining kernel entropy component analysis with the pivoted Cholesky decomposition and convolutional kernel networks. Through an ablation study, we demonstrate that mixing these three concepts is fruitful. We also achieved an acceptable performance compared with several conventional and state-of-the-art models (e.g., PCANet, RNPCANet, CMwv, etc.) on a high-quality SAR dataset called TenGeoP-SARwv, which has a simple architecture and limited labeled samples. Although KECANet has shown potential for automated ocean SAR scene classification, various challenges related to our model remain open for future research. One such challenge concerns

## Data availability statement

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

## Author contributions

MM designed and conceived the experiments; CB and HY implemented the experiments; LQ analyzed the data; CB and SZ wrote the paper. All authors contributed to the article and approved the submitted version.

## Funding

This work was supported by the National Natural Science Foundation of China (Grant No. 42106200).

## Conflict of interest

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

## References

- Bai, L., Han, Z., Ren, J., and Qin, X. (2020b). Research on feature selection for rotating machinery based on supervision kernel entropy component analysis with whale optimization algorithm. *Appl. Soft Comput.* 92, 106245. doi: 10.1016/j.asoc.2020.106245
- Bai, C., Zhang, R., Xu, Z., Cheng, R., Jin, B., and Chen, J. (2019). L1-norm-based kernel entropy components. *Pattern Recognit.* 96, 106990. doi: 10.1016/j.patcog.2019.106990
- Bai, C., Zhang, R., Xu, Z., Jin, B., Chen, J., Zhang, S., et al. (2020a). Kernel low-rank entropic component analysis for hyperspectral image classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* 13, 5682–5693. doi: 10.1109/jstars.2020.3024241
- Bao, S., Meng, J., Sun, L., and Liu, Y. (2020). Detection of ocean internal waves based on faster r-CNN in SAR images. *J. Oceanol. Limnol.* 38, 55–63. doi: 10.1007/s00343-019-9028-6
- Bengio, Y., and LeCun, Y. (2007). Scaling learning algorithms towards AI. in *Large-Scale Kernel Mach.* (Boston, America: MIT Press): 34, 1–41.
- Bioresita, F., Puissant, A., Stumpf, A., and Malet, J. P. (2018). A method for automatic and rapid mapping of water surfaces from sentinel-1 imagery. *Remote Sens.* 10, 217. doi: 10.3390/rs10020217
- Bjerreskov, K. S., Nord-Larsen, T., and Fensholt, R. (2021). Classification of nemoral forests with fusion of multi-temporal sentinel-1 and 2 data. *Remote Sens.* 13, 950. doi: 10.3390/rs13050950
- Bo, L., and Sminchisescu, C. (2009). Efficient match kernel between sets of features for visual recognition. *Adv. Neural Inf. Process. Syst.* 22, 1–9.
- Bottou, L. (2012). “Stochastic gradient descent tricks,” in *Neural networks: Tricks of the trade*. Eds. G. Montavon, G. B. Orr and K. R. Müller (Berlin/Heidelberg, Germany: Springer), 421–436.
- Chan, T. H., Jia, K., Gao, S., Lu, J., Zeng, Z., and Ma, Y. (2015). PCANet: a simple deep learning baseline for image classification? *IEEE Trans. Image Process.* 24, 5017–5032. doi: 10.1109/tip.2015.2475625
- Clausi, D. A. (2001). Comparison and fusion of co-occurrence, gabor and MRF texture features for classification of SAR sea-ice imagery. *Atmos. Ocean* 39, 183–194. doi: 10.1080/07055900.2001.9649675
- Du, Y., Song, W., He, Q., Huang, D., Liotta, A., and Su, C. (2019). Deep learning with multi-scale feature fusion in remote sensing for automatic oceanic eddy detection. *Inf. Fusion* 49, 89–99. doi: 10.1016/j.inffus.2018.09.006
- Franz, K., Roscher, R., Milioto, A., Wenzel, S., and Kusche, J. (2018). Ocean eddy identification and tracking using neural networks. *Proceedings of the IGARSS 2018 IEEE International Geoscience and Remote Sensing Symposium*, Valencia, Spain. 6887–6890 (IEEE). doi: 10.1109/IGARSS.2018.8519261
- Gao, F., Dong, J., Li, B., and Xu, Q. (2016). Automatic change detection in synthetic aperture radar images based on PCANet. *IEEE Geosci. Remote Sens. Lett.* 13, 1792–1796. doi: 10.1109/lgrs.2016.2611001
- Gomez-Chova, L., Jenssen, R., and Camps-Valls, G. (2011). Kernel entropy component analysis for remote sensing image clustering. *IEEE Geosci. Remote Sens. Lett.* 9, 312–316. doi: 10.1109/lgrs.2011.2167212
- Gommenginger, C., Chapron, B., Hogg, A., Buckingham, C., Fox-Kemper, B., Eriksson, L., et al. (2019). SEASTAR: A mission to study ocean submesoscale dynamics and small-scale atmosphere-ocean processes in coastal, shelf and polar seas. *Front. Mar. Sci.* 6. doi: 10.3389/fmars.2019.00457
- Harbrecht, H., Peters, M., and Schneider, R. (2012). On the low-rank approximation by the pivoted cholesky decomposition. *Appl. Numer. Math.* 62, 428–440. doi: 10.1016/j.apnum.2011.10.001
- Izquierdo-Verdiguier, E., Laparra, V., Jenssen, R., Gomez-Chova, L., and Camps-Valls, G. (2016). Optimized kernel entropy components. *IEEE Trans. Neural Netw. Learn. Syst.* 28, 1466–1472. doi: 10.1109/tnnls.2016.2530403
- Jenssen, R. (2009). Kernel entropy component analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* 32, 847–860. doi: 10.1109/tpami.2009.100
- Kingma, D. P., and Ba, J. (2014). Adam: a method for stochastic optimization. *arXiv* 1412, 6980. doi: 10.48550/arXiv.1412.6980
- Krestenitis, M., Orfanidis, G., Ioannidis, K., Avgerinakis, K., Vrochidis, S., and Kompatsiaris, I. (2019). Oil spill identification from satellite images using deep neural networks. *Remote Sens.* 11, 1762. doi: 10.3390/rs11151762
- Krogstad, H. E., Samset, O., and Vachon, P. W. (1994). Generalizations of the non-linear ocean-SAR transform and a simplified SAR inversion algorithm. *Atmos. Ocean* 32, 61–82. doi: 10.1080/07055900.1994.9649490
- Lguensat, R., Sun, M., Fablet, R., Tandeo, P., Mason, E., and Chen, G. (2018). “EddyNet: a deep neural network for pixel-wise classification of oceanic eddies,” in *Proceedings of the IGARSS 2018 IEEE International Geoscience and Remote Sensing Symposium*, Valencia, Spain. 1764–1767 (IEEE). doi: 10.1109/IGARSS.2018.8518411
- Li, H., Huang, H., Chen, L., Peng, J., Huang, H., Cui, Z., et al. (2020). Adversarial examples for CNN-based SAR image classification: an experience study. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* 14, 1333–1347. doi: 10.1109/jstars.2020.3038683
- Li, W., Liu, L., and Zhang, J. (2022). AdaRW training optimization algorithm for deep learning model of marine target detection based on SAR. *Int. J. Remote Sens.* 43, 120–131. doi: 10.1080/01431161.2021.2005841
- Lima, E., Sun, X., Dong, J., Wang, H., Yang, Y., and Liu, L. (2017). Learning and transferring convolutional neural network knowledge to ocean front recognition. *IEEE Geosci. Remote Sens. Lett.* 14, 354–358. doi: 10.1109/lgrs.2016.2643000
- Low, C. Y., Teoh, A. B. J., and Toh, K. A. (2017). Stacking PCANet+: an overly simplified ConvNets baseline for face recognition. *IEEE Signal Process. Lett.* 24, 1581–1585. doi: 10.1109/lsp.2017.2749763
- Mairal, J. (2016). End-to-end kernel learning with supervised convolutional kernel networks. *Adv. Neural Inf. Process. Syst.* 29, 1–9.
- Mairal, J., Koniusz, P., Harchaoui, Z., and Schmid, C. (2014). Convolutional kernel networks. *Adv. Neural Inf. Process. Syst.* 27, 1–10.
- Marghany, M. (2014). Utilization of a genetic algorithm for the automatic detection of oil spill from RADARSAT-2 SAR satellite data. *Mar. pollut. Bull.* 89 (1–2), 20–29. doi: 10.1016/j.marpolbul.2014.10.041
- Meier, W. N., Fetterer, F., Windnagel, A. K., and Stewart, J. S. (2021). *NOAA/NSIDC climate data record of passive microwave Sea ice concentration, version 4* (Boulder, Colorado USA: NSIDC: National Snow and Ice Data Center). doi: 10.7265/efmz-2t65
- Mohammadnia-Qaraei, M. R., Monsefi, R., and Ghiasi-Shirazi, K. (2018). Convolutional kernel networks based on a convex combination of cosine kernels. *Pattern Recognit. Lett.* 116, 127–134. doi: 10.1016/j.patrec.2018.09.016
- Potin, P., Rosich, B., Miranda, N., Grimont, P., Bargellini, P., Monjoux, E., et al. (2016). “Sentinel-1 mission status,” in *Proceedings of the EUSAR 2016: 11<sup>th</sup> European Conference on Synthetic Aperture Radar*, (Hamburg, Germany: IEEE). 1–6.

## 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.

- Qaraei, M., Abbaasi, S., and Ghiasi-Shirazi, K. (2021). Randomized non-linear PCA networks. *Inf. Sci.* 545, 241–253. doi: 10.1016/j.ins.2020.08.005
- Rahimi, A., and Recht, B. (2007). Random features for large-scale kernel machines. *Adv. Neural Inf. Process. Syst.* 20, 1–8.
- Salberg, A.-B., and Larsen, S. O. (2018). Classification of ocean surface slicks in simulated hybrid-polarimetric SAR data. *IEEE Trans. Geosci. Remote Sens.* 56, 7062–7073. doi: 10.1109/tgrs.2018.2847724
- Schölkopf, B., Smola, A., and Müller, K.-R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Comput.* 10, 1299–1319. doi: 10.1162/089976698300017467
- Torres, R., Snoeij, P., Geudtner, D., Bibby, D., Davidson, M., Attema, E., et al. (2012). GMES sentinel-1 mission. *Remote Sens. Environ.* 120, 9–24. doi: 10.1016/j.rse.2011.05.028
- Wang, C., Mouche, A., Tandeo, P., Stopa, J. E., Longépé, N., Erhard, G., et al. (2019a). A labelled ocean SAR imagery dataset of ten geophysical phenomena from sentinel-1 wave mode. *Geosci. Data J.* 6, 105–115. doi: 10.1002/gdj3.73
- Wang, C., Tandeo, P., Mouche, A., Stopa, J. E., Gressani, V., Longepe, N., et al. (2019b). Classification of the global sentinel-1 SAR vignettes for ocean surface process studies. *Remote Sens. Environ.* 234, 111457. doi: 10.1016/j.rse.2019.111457
- Wang, C., Vandemark, D., Mouche, A., Chapron, B., Li, H., and Foster, R. C. (2020a). An assessment of marine atmospheric boundary layer roll detection using sentinel-1 SAR data. *Remote Sens. Environ.* 250, 112031. doi: 10.1016/j.rse.2020.112031
- Wang, W., Wu, J., Pei, J., Mao, X., and Yang, J. (2020b). An antideceptive jamming method for multistatic synthetic aperture radar based on collaborative localization and spatial suppression. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* 13, 2757–2768. doi: 10.1109/jstars.2020.2997345
- Wu, S. Y., and Liu, A. K. (2003). Towards an automated ocean feature detection, extraction and classification scheme for SAR imagery. *Int. J. Remote Sens.* 24, 935–951. doi: 10.1080/01431160210144606
- Xie, D., Ma, J., Li, Y., and Liu, X. (2021). “Data augmentation of sar sensor image via information maximizing generative adversarial net,” in *Proceedings of the 2021 IEEE 4th International Conference on Electronic Information and Communication Technology (ICEICT)*, Xi'an, China. 454–458 (IEEE). doi: 10.1109/ICEICT53123.2021.9531250
- Yan, Z., Chong, J., Zhao, Y., Sun, K., Wang, Y., and Li, Y. (2020). Multifeature fusion neural network for oceanic phenomena detection in SAR images. *Sensors* 20, 210. doi: 10.3390/s20010210
- Zhang, Z., and Hancock, E. R. (2012). Kernel entropy-based unsupervised spectral feature selection. *Int. J. Pattern Recognit. Artif. Intell.* 26, 1260002. doi: 10.1142/s0218001412600026
- Zhang, X., Liu, G., Zhang, C., Atkinson, P. M., Tan, X., Jian, X., et al. (2020). Two-phase object-based deep learning for multi-temporal SAR image change detection. *Remote Sens.* 12, 548. doi: 10.3390/rs12030548
- Zhou, S. (2015). Sparse LSSVM in primal using cholesky factorization for large-scale problems. *IEEE Trans. Neural Netw. Learn. Syst.* 27, 783–795. doi: 10.1109/tnnls.2015.2424684