A Viewpoint Adaptation Ensemble Contrastive Learning Framework for Vessel Type Recognition with Limited Data

Xiaocai Zhang[a], Zhe Xiao[a,*], Xiuju Fu[a], Xiaoyang Wei[a], Tao Liu[b], Ran Yan[c], Zheng Qin[a], Jianjia Zhang[d,*]

[a]*Institute of High Performance Computing, Agency for Science, Technology and Research (A*STAR), 138632, Singapore*
[b]*College of Transport & Communications, Shanghai Maritime University, Shanghai 201306, China*
[c]*School of Civil and Environmental Engineering, Nanyang Technological University, 637551, Singapore*
[d]*School of Biomedical Engineering, Sun Yat-sen University, Shenzhen, 518107, China*

[*]Corresponding author.
Email addresses: zhang_xiaocai@ihpc.a-star.edu.sg (X. Zhang), xiaoz@ihpc.a-star.edu.sg (Z. Xiao), fuxj@ihpc.a-star.edu.sg (X. Fu), wei_xiaoyang@ihpc.a-star.edu.sg (X. Wei), liutao2@shumtu.edu.cn (T. Liu), ran.yan@ntu.edu.sg (R. Yan), qinz@ihpc.a-star.edu.sg (Z. Qin), zhangjj225@mail.sysu.edu.cn (J. Zhang)

October 2, 2023

Abstract

Unmanned Aerial Vehicle (UAV)-based systems are gaining increasing attention in the maritime industry, but one of their major challenges is accurately identifying vessel types from bird-view images captured by UAV. The use of computer vision and deep learning technologies in image recognition requires large amounts of annotated images for model training, but collecting and manually annotating these images is a costly and time-consuming task. To overcome these challenges, we propose a novel Viewpoint Adaptation Ensemble Contrastive Learning (VAECL) framework. With the VAECL framework, first, an improved deep generative model (DGM) is constructed to learn the distribution of the limited vessel image data and to generate more images for data augmentation. Second, transfer learning using a pre-trained Inception V3 network is then presented for vessel viewpoint transfer and adaptation learning. Third, contrastive learning paradigm is adopted by formulating a new loss function to obtain more contrastive feature representation via pre-training. Finally, an ensemble learning algorithm is highlighted to further improve the performance of vessel type recognition. Extensive experiments on a newly-established dataset reveal the following encouraging findings: 1) VAECL can achieve high accuracy of 92.96% with proper parameters; 2) DGM-based image augmentation improves the accuracy by at least 14.68%, and it performs better than the traditional data augmentation techniques; 3) viewpoint transfer learning surges the accuracy by as high as 18.50%; 4) contrastive learning increases performance by at least 2.79% in terms of accuracy; 5) ensemble learning enhances the accuracy by as high as 1.03%.

Keywords: Maritime transportation, vessel type recognition, viewpoint adaptation, ensemble, contrastive learning

1. Introduction

Since 2010, Unmanned Aerial Vehicle (UAV) has been gaining significant popularity in various civilian uses, such as agriculture, environment, transportation, construction and mining, etc., primarily due to their growing accessibility in terms of flexibility, automated features like navigation and flight control, and affordability (Su et al., 2022). In recent years, UAV-based systems have revolutionized traditional maritime transportation by increasing intelligence, safety and security. They have drawn attention due to their high efficiency and low cost, with promising applications in traffic management (Yang, 2019), ship-shore drone delivery (Agata, 2021), pollution surveillance (Messinger & Silman, 2016), vessel emission monitoring (Sun et al., 2022b), water depth estimation(Liu et al., 2022a), and maritime search & rescue (Sun et al., 2022a). Vessel identification is a fundamental and crucial area of research in the UAV-based systems in maritime. Furthermore, the type of vessel, i.e., container vessel, tanker vessel, bulk carrier, etc., is a very reliable indicator for precisely identifying the vessels, as it has been classified and categorized based on internationally recognized classification systems, and typically cannot be altered without compliance with relevant regulations. As a result, the recognition of vessel types plays a crucial role in facilitating both vessel identification and the successful implementation of UAV-based systems in maritime.

For the UAV-based systems in maritime, camera is mounted on the UAV that is usually dozens of meters above the sea level (Frederiksen & Knudsen, 2018). The cameras capture the images from a bird-view with different perspectives, making the accurate recognition of vessel type becomes a challenging problem. In particular, Fig. 1 provides an example of the UAV-view images of a general cargo ship and a bulk carrier ship. In Fig. 1 (a), the onboard crane equipment, which is a crucial factor, is mostly obscured from view by the bridge, making it challenging to be distinguished from the bulk carrier shown in Fig. 1 (b). Moreover, cutting-edge computer vision and deep learning approaches necessitate large amounts of annotated training images with the same or similar distribution as the test data. Image collection, as well as the follow-up manual annotation, is however costly and time-consuming tasks. These challenges spurred a new line of research that focused on finding more effective methods. Therefore, in this work, we attempt to introduce a viewpoint adaptation learning approach based on limited image data to overcome these challenges.

Deep learning models have been successful in image classification, but they require a large amount of data for effective training (Saufi et al., 2020). However, the collection and human labeling of large-scale UAV-view images in the maritime environment are time-consuming and costly (Wang et al., 2021a). In order to compensate for the shortage of data insufficiency, we are motivated to develop a robust method capable of generating new images with similar pixel distribution to the original images. In this work, we leverage the framework of generative adversarial network (GAN) to learn the data distribution from limited vessel image samples via adversarial training (Yang et al., 2021). In particular, an improved DGM (Gulrajani et al., 2017), i.e., conditional Wasserstein deep convolutional GAN with gradient penalty (C-WDCGAN-GP), is designed by considering different types of vessels and ensuring more stable gradients during network training (Gao et al., 2020).

(a) General Cargo Ship (b) Bulk Carrier Ship

Figure 1: An example of a general cargo ship and a bulk carrier ship.

Transfer learning is a method that allows us to utilize well-trained models for new problems, providing a convenient and efficient approach to save time. This is especially useful when the existing models have been trained on large amounts of data and retraining from scratch on new datasets would be difficult (Ahmed et al., 2021). Inspired by this, we highlight an ensemble architecture of two pre-trained deep learning models for viewpoint adaptation learning of the vessel images from the UAV's perspectives. The ensemble architecture employs multiple classifiers on the dataset and integrates the results using an ensemble logic, aiming to enhance performance. The individual deep learning model is trained using a contrastive learning paradigm, which involves formulating a new loss function for training. The contributions of this study are summarized as follows:

1. To the best of our knowledge, this is the first study to explore the vessel type recognition problem under the challenge of insufficient data;

2. We established a new human-annotated vessel image dataset called DVTR (Dataset for Vessel Type Recognition) containing both bird-view and front-view images for experiments, which is also publicly available to facilitate future research;

3. We present the C-WDCGAN-GP method for the data augmentation of vessel images with limited samples, which helps compensate for the lack of learning capability with a small dataset;

4. We propose a novel viewpoint adaptation ensemble contrastive learning (VAECL) framework to improve the accuracy and robustness of vessel type recognition. VAECL consists of two separate networks that work together as an ensemble. These networks are trained using novel pre-training and fine-tuning procedures.

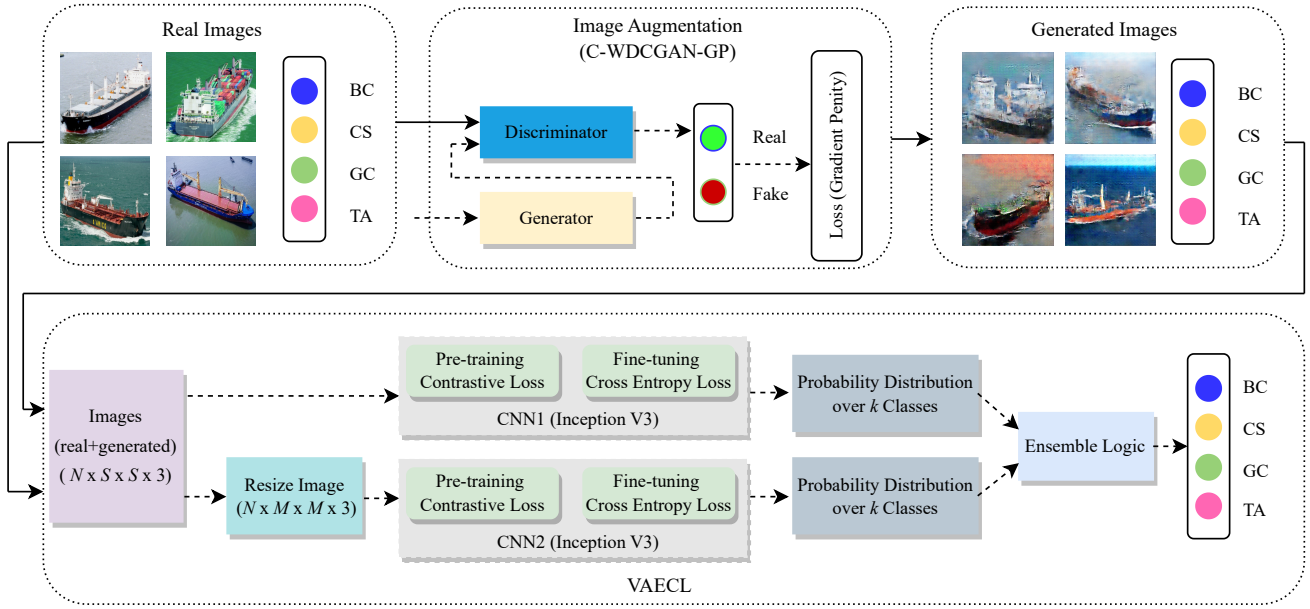


Figure 2: The workflow of our proposed method for vessel type recognition, where "BC", "CS", "GC" and "TA" stand for "bulk carrier ship", "container ship", "general cargo ship" and "tanker ship", respectively.

The remaining of this paper is organized as follows. Section 2 presents a comprehensive summary of previous studies on vessel type recognition. Section 3 presents the details of our proposed methods including fundamental background knowledge, C-WDCGAN-GP-based image data augmentation, and VAECL framework. Section 4 reports experimental findings and analyses on the constructed DVTR dataset. The work's conclusions are summarized in Section 5.

2. Related Work

Vessel type recognition has gained significant attention in recent studies, resulting in various algorithms proposed to solve the problem using different data sources. These sources include 1) vessel trajectory; 2) radar image; 3) optical image; 4) computer-simulated image; 5) CAD model profile of the ship; 6) audio signal data; 7) laser sensor data; and 8) hybrid data sources. In the subsequent paragraphs, we will examine studies that were modelled based on these data sources.

In the work by Huang et al. (2018), an extreme gradient boosting (XGBoost) classifier based on trajectory data was constructed for fishing vessel type recognition. Zhang et al. (2021) visualized the voyage trajectories of the fishing vessel as characteristic trajectory map. Following that, a pre-trained VGG16 network was introduced for fishing vessel classification with the feature trajectory map as input. Marzuki et al. (2017) extracted features from the vessel monitoring system (VMS) position data in an unsupervised way by Gaussian mixture model (GMM). Then, supervised learning algorithms based on random forest and support vector machine were employed for fishing gear identification. In the study by Yang et al. (2022), vessel type recognition based on automatic identification system (AIS) data was investigated. The AIS data is pre-processed and transformed into trajectory images with labels, and a convolutional neural network (CNN) model was developed to learn over these generated trajectory images. In the study by Buscema et al. (2023), they introduced a method utilizing artificial neural networks (ANN) to categorize ship types using information from the ship's radar trajectory. The radar trajectory data used for this classification comprises attributes such as identification, location, instantaneous speed, and heading. Xu et al. (2023) presented an approach for ship classification that relies on trajectory data, utilizing the light gradient boosting machine (LightGBM). They further improved the classification accuracy by incorporating additional features like the offshore distance.

In the study by Hou et al. (2020), a high-resolution satellite synthetic aperture radar (SAR) image dataset was collected. The detection and recognition of ships in SAR images was performed by a CNN model with an average accuracy of 79%. Zhu et al. (2016) proposed a novel method based on the projection shape template (PST) of SAR images to enhance the accuracy and robustness of the recognition. The PST is obtained by projecting the 3-D model to the 2-D slant-plane image. Zhao et al. (2023) developed an improved YOLOv5s, consisting of a convolutional block attention module (CBAM), receptive fields block (RFB) and adaptively spatial feature fusion (ASFF), for vessel detection and classification from satellite SAR images.

Voinov et al. (2018) introduced a faster R-CNN Inception-ResNet (FRCIR), which combines residual connections with the Inception architecture, for automatic vessel object detection and recognition form optical satellite images. The presented method exhibits potential near-real time applications. The research work by Santos & Bhanu (2018) simulated ad-

ditional sensor data by applying image transformation on the original optical vessel images. This can improve the robustness of the CNN model for classification tasks. Furthermore, the contextual information was exploited to calculate and update the classification confidence of the CNN predictions. Chen et al. (2020b) developed a cascaded CNN network (i.e., CFC-CNN) based on ship image data. It is trained by a coarse step followed by a fine step. The coarse step is trained in a similar way to the vanilla CNN, while the fine step leverages regularisation mechanisms to extract more intrinsic features and fine-tune the weights. Liu et al. (2019) demonstrated a deep residual network with cross-layer jump connection policy for ship recognition and tracking from camera video datasets. The work by Chen et al. (2020a) focused on the maritime surveillance video data by suggesting a you only look once (YOLO) deep learning framework to address the research problem of ship recognition. The recent YOLO v7 version has also been used by Haijoub et al. (2023) for ship detection and recognition purposes. Salem et al. (2023) introduced a transfer learning approach using the EfficientNetB0 network for classifying vessel types. The authors applied particle swarm optimization (PSO) for optimizing hyperparameters to further improve the performance of the model. Han et al. (2021) highlighted an efficient information reuse network for fine-grained recognition of vessels. To optimize the use of multi-layer information and eliminate information redundancy, the network includes a dense feature fusion network. To enhance performance, a dual-mask attention module refines the fused features. Ren et al. (2021) constructed a ship recognition algorithm based on CNN and Hu invariant moments. First, CNN is employed to extract features of ship image after denoising and segmentation. Second, the image is divided into sub-images horizontally, and Hu invariant moments is then extracted. Third, CNN features and Hu invariant moments are fused to a classifier. Wang et al. (2022) devised a multi-scale paralleling CNN model for military ship recognition and classification. The method parallelizes convolution branches with different receptive fields to extract the feature of images of different sizes. Experimental results showed that the formulated method can achieve 84.79% accuracy on the dataset. Meng et al. (2022) demonstrated a global to local progressive learning model for fine-grained ship recognition. In order to obtain high-level global features, the feature pyramid attention module is first introduced. Second, the output global features enable the weighting of local key pixel values in the target feature region and progressively guide the learning of local feature extraction networks at various resolution levels. Finally, the global to local progressive module's output vector and the backbone network's output are combined for classification.

The research work by Karabayır et al. (2017) firstly explored the high resolution range profiles of the CAD models of different ships to establish a CNN network for radar target recognition. Shen et al. (2020) probed the ship type classification problem using the hydrophone acoustic data. It introduced a CNN network with multiple auditory-like mechanisms for recognizing the type of ship. In the research conducted by Yildirim (2023), they introduced a deep learning approach for recognizing ship types based on audio signals. Their method involves converting the audio signal into the frequency domain through Fourier transform, resulting in the creation of frequency-magnitude graphs as images. Subsequently, they constructed a ResNet50 network for the purpose of classifying these images. For the ship recognition based on laser sensor data, Zheng et al. (2023) applied the concept of transfer learning to identify small fishing vessels using laser sensor data. Initially, they applied a polynomial fitting technique to extract the contours of fishing vessels. Next, they converted these one-dimensional vessel contours into two-dimensional images. Lastly, they employed the VGG-16 model for the vessel type recognition process.

In addition to using only one data source, Liu et al. (2021) provided a method using hybrid data sources including marine radar and closed-circuit television (CCTV) images. First, the marine radar image is characterized to obtain a region of interest (ROI) area via a coarse detection of the ship. Following that, a CNN model is set up for the ship recognition based on the CCTV images. Similarly, Sun et al. (2022a) highlighted a network network based on domain adaptation and transformer for horizontal view ship images recognition. The ship images consist of real image and computer-simulated image. The local maximum mean discrepancy was utilized in the domain adaptation block to overcome the domain barrier between real and simulation images. Furthermore, a transformer was constructed to extract features for classification task.

Upon examining the available studies on identifying vessel types, it becomes evident that previous works have not adequately addressed the issue of limited data. Additionally, these methods have not explore the potential of bird-view image recognition for optical images. Moreover, cutting-edge techniques like contrastive learning have not been extensively investigated within this domain. Furthermore, a thorough and comprehensive exploration of various existed pre-trained models for transfer learning has not been extensively conducted on this field.

3. Methodologies

In this section, we provide a comprehensive overview of the techniques utilized in this study. Section 3.1 introduces the key concepts and algorithms, such as GAN, CNN, Inception V3, contrastive learning, and ensemble learning. Subsequently, the C-WDCGAN-GP method for image data augmentation is described in detail, aimed at addressing the insufficiency problem of training data. Finally, the VAECL framework, which integrates multiple techniques to achieve enhanced vessel type recognition performance, is presented. A complete workflow is presented in Fig. 2, encapsulating the data augmentation and ensemble learning steps involved in our approach.

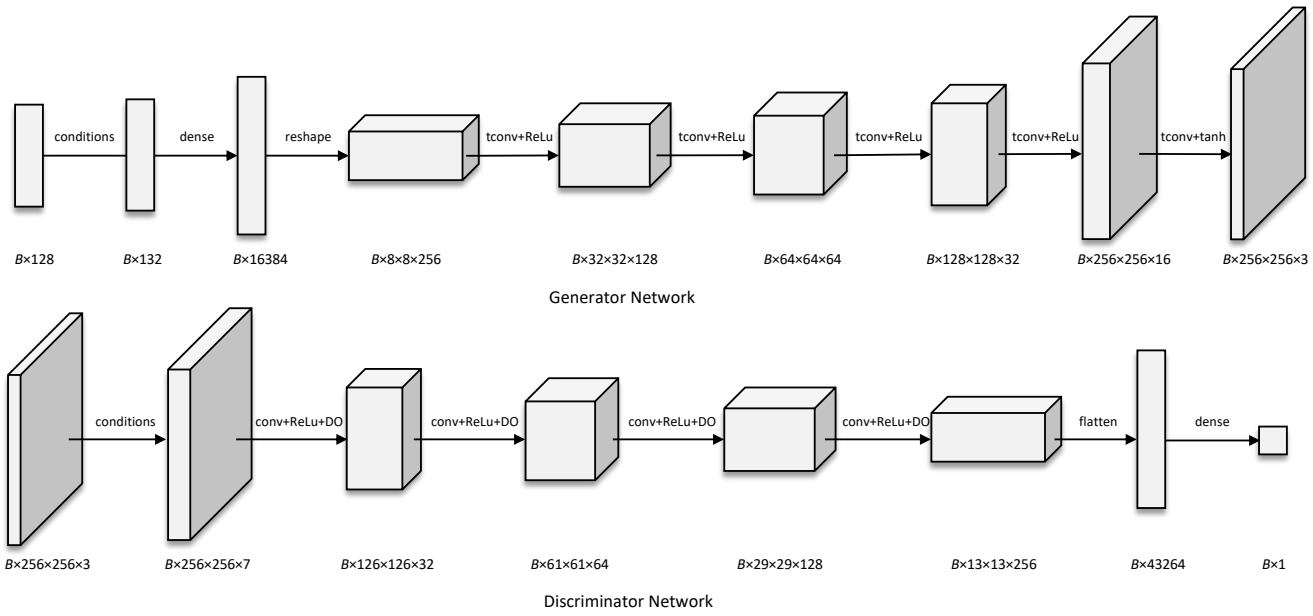


Figure 3: The architecture of C-WDCGAN-GP for vessel image augmentation. "tcov" stands for transposed convolution. "DO" is the abbreviation for dropout. B denotes the batch size.

3.1. Fundamental Backgrounds

3.1.1. Generative Adversarial Network (GAN)

GAN has made a huge impact in synthesizing new data samples by giving a prior data distribution (Wu et al., 2021). A basic GAN framework consists of a generator G and a discriminator D, which are simultaneously trained (Goodfellow et al., 2014). The G is to capture the data distribution, while D is to estimate the probability of a sample being from the training data rather than G. The training procedure for G is to maximize the probability of D making a mistake, while for D is to minimize the discriminative error. The GAN framework is introduced to learn the distribution from limited UAV-view vessel images and tried to generate more reliable images to compensate the shortage of limited image data.

3.1.2. Convolutional Neural Network (CNN)

CNN is a basic deep learning architecture that is capable of preserving and extracting the spatial features of the input data (Gu et al., 2018). A typical layered CNN architecture contains two different variants of operator, i.e., convolution and pooling. The convolution operation $C(x, y)$ is based on the Laplacian filters to generate feature maps of the input data or the last hidden features $h(x, y)$, as formulated by

$$f_{conv}(x, y) = \sum_{x=1}^{M} \sum_{y=1}^{N} h(x, y) * C(x, y). \quad (1)$$

Then, a nonlinear transformation is applied to represent the output feature maps into a higher and more abstract level (LeCun et al., 2015). The pooling operator (e.g., maximum, minimum, average, etc.) is intrinsically performed as the downsampling of its input. Finally, a fully-connected layer is attached to the top of the CNN network to perform classification tasks. Many pre-trained CNN models have been developed for fine-tuning downstream tasks, and this can be particularly advantageous for the task of recognizing vessel types.

3.1.3. Inception V3

Since 2014, the quality of CNN network architectures has been substantially improved by utilizing deeper and wider networks as well as the pre-training technology that has become a well-established paradigm in many computer vision tasks (Wang et al., 2021b). The Inception architecture is one of the successful stories. As the third version of the GoogLeNet family, Inception V3 (Szegedy et al., 2016) is a CNN that was initially presented in the ILSVRC-2015 image classification challenge and obtained a high accuracy (Al Husaini et al., 2022). The pre-trained Inception V3 network was trained on the ImageNet database, and it is able to classify images into 1,000 categories. The preliminaries of Inception V3 are to introduce several strategies for optimizing the network and loosening the constraints for easier model adaptation. Such strategies include factorized convolutions, regularization, dimension reduction, parallelized computation, and so on. Inception V3 incurs lower computational costs (Ahmed et al., 2023), which is a critical factor to take into account in our research.

3.1.4. Contrastive Learning

Contrastive learning (CL) is a popular self-supervised paradigm and has made great success in computer vision tasks in recent years (Wang & Qi, 2022). Its goal is to find a parametric function f_θ that maps the input image $x \in \mathbb{R}^D$ to a feature representation $z \in \mathbb{R}^d$, so that the representation z in the feature space can reflect the semantic similarities in the

input space (Wang & Qi, 2022). To achieve this, a contrastive loss is brought forward to optimize the network f_θ, which encourages z and its positive pair z' to be close in the feature space while pushing away representations of all other negative pairs, where images from the same classes are defined as positive pairs while images from different classes as negative pairs. In self-supervised learning, contrastive loss is to maximize the agreement of representations of different views of the same instance while minimizing the agreement with other negative samples. CL can extract more distinctive and discriminative features, which serves a valuable asset in our vessel type pattern recognition efforts.

3.1.5. Ensemble Learning

Ensemble learning fuses multiple hypotheses of the base learners to form a better hypothesis, yield producing more accurate and robust results (Zhang et al., 2020; Zheng et al., 2018). The fusion functions that determine the ensemble output are diverse, including voting, averaging, stacking, bagging, boosting, rule learning, etc., and the application varies depending on the specific objective of each problem (Ahmed et al., 2021). Ensemble learning can be used for both classification and regression problems. Ensemble learning is adopted to further enhance the recognition accuracy and robustness in this research.

3.2. C-WDCGAN-GP-Based Image Data Augmentation

The framework of C-WDCGAN-GP employs the deep convolutional network as the basic architecture for both the generator and discriminator. The conditions in this study are represented by the category of vessel type. Suppose the size of input RGB image is $256 \times 256 \times 3$, and the number of category is 4 (i.e., BC, CS, GC and TA) in this study, the structures of generator network (G) as well as discriminator network (D) are displayed in Fig. 3, where B stands for the batch size while using mini-batch stochastic gradient descent (SGD) for training. For the generator, a batch of latent variables z with the size of 128 is sampled from a given distribution $p(z)$. Then, conditions represented by one-hot categorical vectors are concatenated to z, deriving variables with size of 132. Following that, a dense layer and a reshape layer are appended to map it into 3D space ($B \times 8 \times 8 \times 256$). Afterward, five transposed convolutional layers with nonlinear transformations (i.e., rectified linear unit (ReLu) and Tanh) are followed to generate images of the same size as the real images. Since the real images are all normalized into $[-1, 1]$, a Tanh activation function is placed at the top of the generator network to produce output in the same range.

For the discriminator network, the input comes from the real image, generated image or a mixture of them. Similarly, the conditional vectors are embedded in each pixel of the input image, thus giving a feature map with a size of $256 \times 256 \times 7$. Next, four layers of the combination of convolution, ReLu and dropout are attached to extract the latent features from the input images and conditions. The latent features are then flattened and fed into a dense layer at the top of the network.

Algorithm 1: C-WDCGAN-GP algorithm for vessel image augmentation

Parameters: the number of critic iteration n_{critic}, the number of epoch n_{epoch}, learning rate α, batch size B, and gradient penalty coefficient λ
Input: data including images and conditions $\mathbf{X}$
Output: generator weights θ^G

1: Initialize critic weights ω_0;
2: Initialize generator weights θ_0^G;
3: **for** $epoch = 1, 2, \cdots, n_{epoch}$ **do**
4: $\quad n_{batch} \leftarrow 0$
5: $\quad$ **for** image and condition batches in $\mathbf{X}$ **do**
6: $\quad\quad n_{batch}++$
7: $\quad\quad$ **if** $epoch == 1$ and $n_{batch} == 1$ **then**
8: $\quad\quad\quad \omega \leftarrow \omega_0, \theta^G \leftarrow \theta_0^G$;
9: $\quad\quad$ **end if**
10: $\quad\quad$ **if** $n_{batch} \% (n_{critic} + 1)! = 0$ **then**
11: $\quad\quad\quad$ **for** $i = 1, 2, \cdots, B$ **do**
12: $\quad\quad\quad\quad$ Sample real vessel images x and conditions c, $x, c \in \mathbf{X}$, latent variables $z \sim p(z)$, a random variable $\epsilon \sim U[0, 1]$;
13: $\quad\quad\quad\quad \tilde{x} \leftarrow G\left(z, c|\theta^G\right)$;
14: $\quad\quad\quad\quad \hat{x} \leftarrow \epsilon x + (1 - \epsilon)\,\tilde{x}$;
15: $\quad\quad\quad\quad L^{(i)} \leftarrow D(\tilde{x}, c|\omega) - D(x, c|\omega) + \lambda\left(\|\nabla_{\hat{x}} D(\hat{x}, c|\omega)\|_2 - 1\right)^2$;
16: $\quad\quad\quad$ **end for**
17: $\quad\quad\quad \omega \leftarrow \mathrm{Adam}\left(\nabla_\omega \frac{1}{B} \sum_{i=1}^{B} L^{(i)}, \alpha\right)$;
18: $\quad\quad$ **else**
19: $\quad\quad\quad$ Sample a batch of latent variables $\left\{z^i\right\}_{i=1}^{B} \sim p(z)$;
20: $\quad\quad\quad \theta^G \leftarrow \mathrm{Adam}\left(\nabla_{\theta^G} \frac{1}{B} \sum_{i=1}^{B} -D\left(G\left(z, c|\theta^G\right), c|\omega\right), \alpha\right)$;
21: $\quad\quad$ **end if**
22: $\quad$ **end for**
23: **end for**
24: **return** θ^G

The training process of the C-WDCGAN-GP is outlined in Algorithm 1. The procedure begins by initializing the weights of the critic (ω_0) and the generator (θ_0^G) in steps 1 and 2. The training continues for n_{epoch} epochs, spanning from steps 3 to 23. During the training, the weights of the critic (ω) are updated in steps 10 to 17, while the generator's weights (θ^G) are updated in steps 19 and 20. The learning of the weights of both the critic and generator is achieved through the use of the back-propagation (BP) algorithm and the Adam optimizer. To ensure smooth gradients during the training of the critic, a loss function with gradient penalty is defined in step 15, where a coefficient λ is introduced to control the penalty. This helps ensure the stability and convergence of the training process.

3.3. Viewpoint Adaptation Ensemble Contrastive Learning (VAECL)

Suppose we designate the initial training data's sample size as N_o, and we introduce an oversampling ratio δ_{or} to augment the size of the training data. Thus, the enlarged training data is represented by

$$\mathbf{x} \in \mathbb{R}^{N \times S \times S \times 3}, \tag{2}$$

and

$$N = N_o \times (\delta_{or} + 1), \tag{3}$$

where $S \times S \times 3$ stands for the dimension of the input image. The VAECL model was devised by applying transfer learning to UVA-view vessel image data, enabling adaptation to the bird's eye viewpoints of vessels. This transfer learning process involved fine-tuning two pre-trained Inception V3

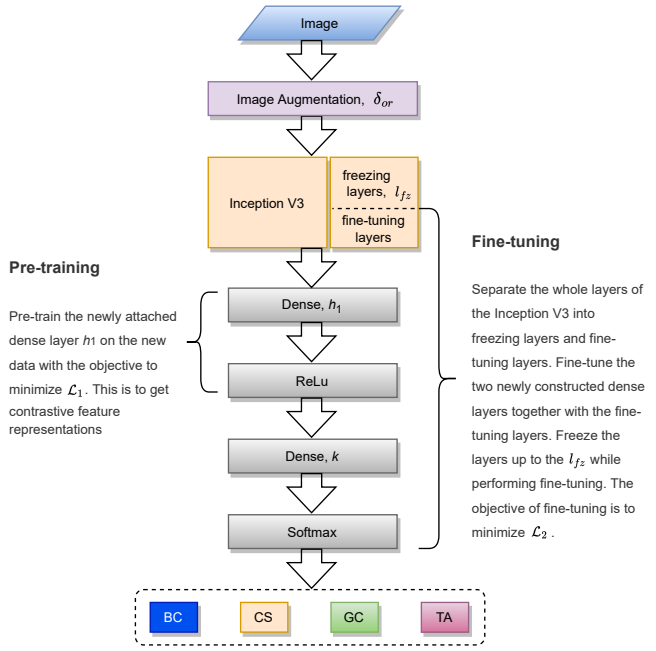


Figure 4: The architecture and training procedure of the Inception V3 model.

networks, which we will elaborate on later in this description. Assume the pre-trained Inception V3 model excluding the top layer denotes $f_{InceptionV3}$, we have

$$\mathbf{y}_{InceptionV3} = f_{InceptionV3}\left(\mathbf{x}|\mathbf{W}_{InceptionV3}\right), \tag{4}$$

where $\mathbf{y}_{InceptionV3}$ means the output of the pre-trained Inception V3 model, and $\mathbf{W}_{InceptionV3}$ is the weights. Following that, one average pooling layer and two dense layers with activation layers, i.e., ReLu and Softmax, are attached, as formulated by

$$\mathbf{y}_{pooling} = f_{pooling}\left(\mathbf{y}_{InceptionV3}\right), \tag{5}$$

and

$$\mathbf{y}_{h_1} = f_{dense}\left(\mathbf{y}_{pooling}, h_1|\mathbf{W}_{h_1}\right), \tag{6}$$

and

$$\mathbf{y}_{h_2} = f_{ReLu}\left(\mathbf{y}_{h_1}\right), \tag{7}$$

and

$$\mathbf{y}_{h_3} = f_{dense}\left(\mathbf{y}_{h_2}, k|\mathbf{W}_{h_3}\right), \tag{8}$$

and

$$\mathbf{y} = f_{Softmax}\left(\mathbf{y}_{h_3}\right), \tag{9}$$

where h_1 represents the unit number of the dense layer, k stands for the number of output class, and $k = 4$ in this study. Fig. 4 illustrates the architecture of the Inception V3 model as well as its training process. First, a pre-training is performed to get the initial weights (i.e., $\mathbf{W}_{pt}$, and $\mathbf{W}_{pt} = \mathbf{W}_{h_1}$) of the new attached dense layer using new data. Then, a fine-tuning procedure is applied to further tune the weights (i.e., $\mathbf{W}_{ft}$) of the top fine-tuning layers of Inception V3 as well as the aforementioned two dense layers by freezing layers up to the l_{fz}th layer.

For the first procedure of pre-training, a contrastive loss is defined as

$$\begin{aligned}\mathcal{L}_1(v_1, v_2, Y) &= (1-Y)\,D(v_1, v_2)^2 \\ &+Y\left\{\max\left(0, m - D(v_1, v_2)\right)^2\right\},\end{aligned} \tag{10}$$

and

$$v_1, v_2 = \mathbf{y}_{h_2}\left(x_1|\mathbf{W}_{pt}\right), \mathbf{y}_{h_2}\left(x_2|\mathbf{W}_{pt}\right), \tag{11}$$

and

$$D(v_1, v_2) = \|v_1 - v_2\|_2, \tag{12}$$

and

$$Y = \begin{cases} 0 & l_{x_1} = l_{x_2} \\ 1 & l_{x_1} \neq l_{x_2}, \end{cases} \tag{13}$$

where $m > 0$ is a margin parameter; l_{x_1} and l_{x_2} denote the labels of input images x_1 and x_2, respectively; v_1 and v_2 are the output via Eqs. (6) and (7), and $D(v_1, v_2)$ is the Euclidean distance between v_1 and v_2; $\mathbf{W}_{pt}$ represents the neural network's pre-training weights. During training, we divided a training batch into two halves: part 1 and part 2. x_1 and x_2 are then iterated from part 1 and part 2 to construct the positive or negative pair based on their labels l_{x_1} and l_{x_2}, respectively.

Let $\hat{\mathbf{y}}$ denote the groundtruth class label, and $\mathbf{y}$ signify the corresponding outputted class probability distribution from Inception V3; thus, the fine-tuning loss can be derived according to

$$\mathcal{L}_2\left(\hat{\mathbf{y}}, \mathbf{y}|\mathbf{W}_{ft}\right) = -\sum_{j=1}^{N} \hat{\mathbf{y}}_j \log\left(\mathbf{y}_j\right), \tag{14}$$

where $\mathbf{W}_{ft}$ stands for the network's fine-tuning weights.

The objectives of network pre-training and fine-tuning are to learn the trainable weights over the training data by minimizing the corresponding loss function, as formulated by

$$\mathbf{W}^*_{pt} = \arg\min_{\mathbf{W}_{pt}} \mathcal{L}_1\left(x_1, x_2, Y|\mathbf{W}_{pt}\right), \tag{15}$$

and

$$\mathbf{W}^*_{ft} = \arg\min_{\mathbf{W}_{ft}} \mathcal{L}_2\left(\hat{\mathbf{y}}, \mathbf{y}|\mathbf{W}_{ft}\right), \tag{16}$$

where pre-training is conducted before fine-tuning, and the pre-training weights $\mathbf{W}^*_{pt}$ is performed as the initial weights for fine-tuning. The optimal weights $\mathbf{W}^*_{pt}$ and $\mathbf{W}^*_{ft}$ are determined through BP using the RMSProp optimizer, similar to the way a basic deep learning network is trained.

The framework of the VAECL module is presented in the lower subfigure of Fig. 2. In VAECL, two Inception V3 networks are trained separately. One is trained on the original image data, and the other is based on the images after resizing. Following that, the crisp outputs of the two Inception V3 models are fused by an ensemble logic. A simple averaging operator of the output probabilities is chosen as the ensemble logic in this work. Details of the VAECL algorithm for vessel type recognition are explained in Algorithm 2. Steps 1 to 4 train an Inception V3 model (CNN1), and steps 5 to 9 train another Inception V3 model (CNN2) after the original images

$\mathbf{X}$ have been resized (step 5). For both CNN models training, the objective of pre-training is to optimize the $\mathcal{L}_1$ loss function, while fine-tuning is to minimize the $\mathcal{L}_2$ loss, as indicated by steps 3, 4, 8 and 9. Following that, steps 10 and 11 predict the probability distribution for CNN1 and CNN2 models given a test instance $\mathbf{x}$. An average logic is then applied to get the final ensemble probability distribution $\mathbf{y}_{ensemble}$ (step 12).

Algorithm 2: VAECL algorithm for vessel type recognition

Input: training data $\mathbf{X}$ and test instance $\mathbf{x}$
Output: probability distribution $\mathbf{y}_{ensemble}$ of $\mathbf{x}$
1: Start training Inception V3 (CNN1) with original images $\mathbf{X}$;
2: $model_{CNN1} \leftarrow$ load Inception V3 model;
3: $model_{CNN1} \leftarrow$ pre-train Inception V3 to minimize $\mathcal{L}_1$;
4: $model_{CNN1} \leftarrow$ fine-tune Inception V3 to minimize $\mathcal{L}_2$;
5: $\hat{\mathbf{X}} \leftarrow$ resize all images in $\mathbf{X}$;
6: Start training Inception V3 (CNN2) with resized images $\hat{\mathbf{X}}$;
7: $model_{CNN2} \leftarrow$ load Inception V3 model;
8: $model_{CNN2} \leftarrow$ pre-train Inception V3 to minimize $\mathcal{L}_1$;
9: $model_{CNN2} \leftarrow$ fine-tune Inception V3 to minimize $\mathcal{L}_2$;
10: $\mathbf{y}_{CNN1} \leftarrow model_{CNN1}(\mathbf{x})$;
11: $\mathbf{y}_{CNN2} \leftarrow model_{CNN2}(\mathbf{x})$;
12: $\mathbf{y}_{ensemble} \leftarrow$ ensemble of $\mathbf{y}_{CNN1}$ and $\mathbf{y}_{CNN2}$;
13: **return** $\mathbf{y}_{ensemble}$

4. Experiments and Analyses

In this section, we have performed comprehensive experiments and analyses to answer the following research questions:

RQ1: Is the C-WDCGAN-GP method effective for vessel image generation? How does its performance compare to that of the conditional deep convolutional GAN (C-DCGAN)? How significantly does the C-WDCGAN-GP-based image augmentation improve performance? And how well does it in contrast to other image augmentation techniques?

RQ2: How accurately is the VAECL? And what is the effect of viewpoint adaptation learning, ensemble learning and contrastive learning in VAECL?

RQ3: How well does the Inception V3 model perform in contrast to other popular pre-trained models in vessel type recognition?

RQ4: How sensitive is the developed VAECL framework to the setting of parameters?

Table 1: Description of the DVTR Dataset

Type	UAV-View Subset Sample	Front-View Subset Sample	Training Sample	Test Sample
BC	221	200	50	171
CS	224	200	50	174
GC	223	200	50	173
TA	213	200	50	163
Total	881	800	200	681

4.1. Dataset

Considering the limitation that there are very few UAV-view vessel image databases currently available for public access, we established a new publicly accessible dataset called DVTR. The whole dataset can be downloaded directly via the Google Drive[1]. The dataset contains a UAV-view as well as a front-view vessel image subset. Table 1 indicates the sample sizes of the above subsets in DVTR. The front-view dataset in this study is only for the baseline evaluation. Four common types of vessel images, i.e., bulk carrier (BC), container ship (CS), general cargo ship (GC) and tanker (TA) (Lampe & Hamann, 2018), were collected from online resources and then annotated manually. The size of all images is 256×256×3. The training set is formed by randomly selecting only 50 images from each type of the UAV-view subset, while the remaining images are used for testing. Thus, the training data is quite limited, including only 200 image samples in total, and the test set has 681 samples. For network training, the augmented training data is further divided into a training set and a validation set with a ratio of 9 : 1. The training set is used to train the deep learning model, and the validation set is for returning the best model.

4.2. Evaluation Metrics

To evaluate the performance of the proposed model, we employ metrics such as accuracy, precision, recall, F1-score, Matthews correlation coefficient (MCC), and area under the receiver operating characteristic (ROC) curve (AUC). The ROC curve is generated based on the true positive rate (TPR) and false positive rate (FPR) definitions, which are formulated as

$$TPR = \frac{TP}{TP + FN}, \tag{17}$$

and

$$FPR = \frac{FP}{FP + TN}. \tag{18}$$

The remaining five assessment indices, i.e., accuracy, precision, recall, F1-score and MCC, are formulated as follows:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \tag{19}$$

and

$$precision = \frac{TP}{TP + FP}, \tag{20}$$

and

$$recall = \frac{TP}{TP + FN}, \tag{21}$$

and

$$F1\text{-}score = 2 \times \frac{precision \times recall}{precision + recall}, \tag{22}$$

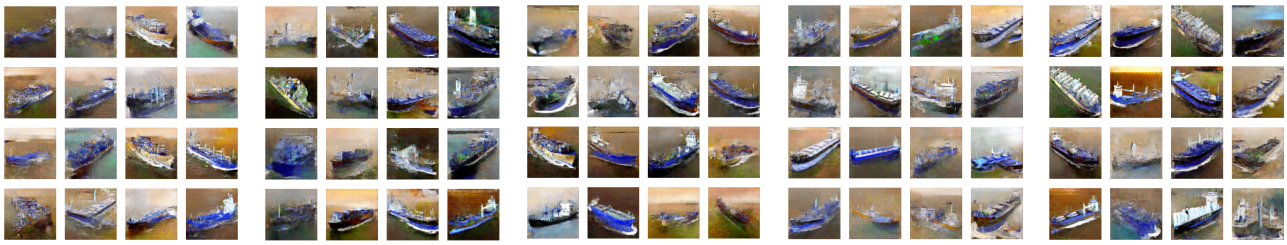
and

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}, \tag{23}$$

where TP, FP, FN, and TN denote true positive, false positive, false negative, and true negative, respectively.

[1] https://drive.google.com/uc?id=132b90eYS_lWbTjYuKXvmqIhPobCAREnq

(a) Epoch 1 (b) Epoch 1000 (c) Epoch 5000 (d) Epoch 10000 (e) Epoch 15000



(f) Epoch 20000 (g) Epoch 25000 (h) Epoch 30000 (i) Epoch 35000 (j) Epoch 40000

Figure 5: Visualization of the generated vessel images by C-WDCGAN-GP at different training epochs.

Table 2: Parameter Settings for the C-WDCGAN-GP

Parameter	n_{critic}	n_{epoch}	α	B	λ
Value	5	4×10^4	5×10^{-4}	32	20

Notes: other parameters regarding the architecture of C-WDCGAN-GP can be found in Fig. 3.

Table 3: Parameter Settings for the VAECL

Parameter	Value	Description
S	256	pixel size of the input image for CNN1
M	512	pixel size of the input image for CNN2
h_1	128	unit for the first attached dense layer
l_{fz}	249	Inception V3 freezing layers for fine-tuning
δ_{or}	20	oversampling ratio
BS	64	batch size
PT-EP	20	training epoch for pre-training
FT-EP	100	training epoch for fine-tuning
LR	0.0001	learning rate
m	2	margin

4.3. Implementation Details

For the C-WDCGAN-GP method, Table 2 lists the setting for each parameter. The rest parameters about the architecture of the generator and discriminator can be referred to Fig. 3. Similarly, Table 3 gives details of the parameter setting of VAECL as well as its description. All models were developed using Python and TensorFlow, and all experiments were conducted under the environment with Intel(R) Xeon(R) Gold 6240 CPU @ 2.60GHz and NVIDIA GeForce RTX 2080 Ti GPU 11GB card. The code is also available in the GitHub repositories[2].

[2] `https://github.com/Xiaocai-Zhang/VAECL`

4.4. Results of Image Data Augmentation (Answering RQ1)

Fig. 5 demonstrates 16 randomly generated vessel images by C-WDCGAN-GP at different training epochs. At the first epoch, the GAN is still unable to synthesize any vessel image. As the training procedure goes on, it can generate images that are taken as real ones. However, in contrast to C-WDCGAN-GP, C-DCGAN fails to synthesize any vessel image, as it encountered mode collapse issue while training. Furthermore, Table 4 presents the effect of C-WDCGAN-GP-based image augmentation in vessel type recognition. For each CNN model in the VAECL module, the model using the original images and generated images by C-WDCGAN-GP is compared to that without any image augmentation. C-WDCGAN-GP-based image augmentation enhances the classification performance of CNN in terms of all five metrics. In particular, the accuracy improves by at least 14.68% (i.e., 90.60% vs 75.92%). Moreover, C-WDCGAN-GP is compared with a traditional data augmentation technique, which augments images by a mixture of flipping, rotation and Gaussian blurring. Overall, C-WDCGAN-GP outperforms the traditional data augmentation technique in all metrics except for the AUC for CNN2, yielding 2.35% and 2.20% improvements in accuracy for CNN1 and CNN2 models.

4.5. Results of VAECL (Answering RQ2)

The comparison between the pre-trained CNN model with and without viewpoint adaptation is described in Table 5. For both CNN models, CNN with viewpoint adaptation outperforms that without viewpoint adaptation greatly in terms of all evaluation indices. The employment of viewpoint adaptation can increase the accuracy by as high as 18.50% (i.e., 90.60% vs 72.10%). Our constructed VAECL is an ensemble

Table 4: Effect of the C-WDCGAN-GP-Based Image Augmentation in Vessel Type Recognition

Model	Accuracy	Precision	Recall	F1-score	MCC	AUC
CNN1 without image augmentation	0.5844	0.6750	0.5787	0.5653	0.4817	0.8863
CNN1 + flipping + rotation + blurring	0.8781	0.8786	0.8775	0.8775	0.8378	0.9767
CNN1 + C-WDCGAN-GP	**0.9016**	**0.9031**	**0.9015**	**0.9022**	**0.8689**	**0.9790**
CNN2 without image augmentation	0.7592	0.8289	0.7590	0.7643	0.7026	0.9276
CNN2 + flipping + rotation + blurring	0.8840	0.9048	0.8852	0.8849	0.8522	**0.9874**
CNN2 + C-WDCGAN-GP	**0.9060**	**0.9076**	**0.9061**	**0.9063**	**0.8751**	0.9845

Table 5: Effect of the Viewpoint Adaptation Learning in Vessel Type Recognition

Model	Accuracy	Precision	Recall	F1-score	MCC	AUC
CNN1 without viewpoint adaptation*	0.7195	0.7319	0.7154	0.7087	0.6316	0.9180
CNN1 + viewpoint adaptation	**0.9016**	**0.9031**	**0.9015**	**0.9022**	**0.8689**	**0.9790**
CNN2 without viewpoint adaptation*	0.7210	0.7480	0.7176	0.7123	0.6419	0.9161
CNN2 + viewpoint adaptation	**0.9060**	**0.9076**	**0.9061**	**0.9063**	**0.8751**	**0.9845**

∗ For a fair comparison, the CNN model without viewpoint adaptation was also trained using the augmented front-view vessel images by the C-WDCGAN-GP approach. The training data sizes for both models are the same.

Table 6: Effect of the Ensemble Learning

Model	Accuracy	Precision	Recall	F1-score	MCC	AUC
CNN1	0.9016	0.9031	0.9015	0.9022	0.8589	0.9790
CNN2	0.9060	0.9076	0.9061	0.9063	0.8751	0.9845
VAECL	**0.9119**	**0.9134**	**0.9118**	**0.9124**	**0.8826**	**0.9896**

Table 7: Effect of the Contrastive Learning

Model	Accuracy	Precision	Recall	F1-score	MCC	AUC
CNN1*	0.8708	0.8783	0.8711	0.8715	0.8299	0.9747
CNN2*	0.8781	0.8901	0.8773	0.8794	0.8407	0.9823
Ensemble	0.9046	0.9072	0.9044	0.9047	0.8734	0.9860
CNN1 + CL	0.9016	0.9031	0.9015	0.9022	0.8589	0.9790
CNN2 + CL	0.9060	0.9076	0.9061	0.9063	0.8751	0.9845
VAECL	**0.9119**	**0.9134**	**0.9118**	**0.9124**	**0.8826**	**0.9896**

∗ CNN1 and CNN2 denote the Inception V3 models that have not utilized the contrastive learning paradigm, and both the procedure of pre-training and fine-tuning adopted the cross entropy loss function ($\mathcal{L}_2$) for learning.

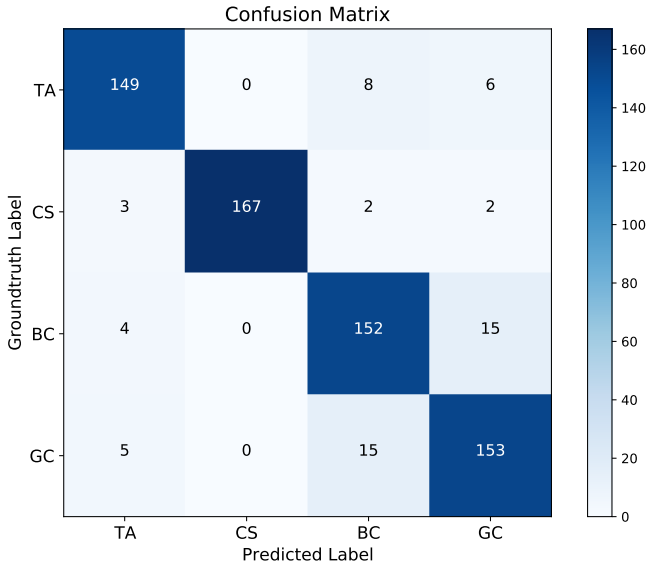


Figure 6: Confusion matrix derived by the VAECL model.

deep learning algorithm based on two pre-trained CNN models. Table 6 reveals the recognition performance of VAECL as well as individual CNN models. The recognition accuracy, precision, recall, F1-score, MCC and AUC reach as high as 91.19%, 91.34%, 91.18%, 91.24%, 88.26% and 98.96%, respectively. The improvement of ensemble mechanism has also been confirmed in Table 7, where the ensemble mechanism increases the accuracy of CNN1 and CNN2 by as high as 1.03% (91.19% vs 90.16%). Table 7 indicates the effect of the proposed contrastive learning paradigm on Inception V3. The accuracy improves by 3.08% (90.16% vs 87.08%) for CNN1 and 2.79% (90.60% vs 87.81%) for CNN2 with contrastive learning. As a result, the performance for ensembling two CNNs with contrastive learning enhances by 0.73% (91.19% vs 90.46%). In Fig. 6, we plot the confusion matrix of the recognition results derived by the VAECL model.

4.6. Comparison with Other Models (Answering RQ3)

For the VAECL framework, the pre-trained Inception V3 model is compared with other eight widely adopted pre-trained models, including VGG19 (Simonyan & Zisserman, 2015), ResNet101 (He et al., 2016), DenseNet121 (Huang et al., 2017), Xception (Chollet, 2017), MobileNetV2 (Sandler et al., 2018), NASNet (Zoph et al., 2018), ConvNeXt (Liu et al., 2022b), and YOLO v5 (Jocher et al., 2022). The results are shown in Fig. 7 (a), which displays the accuracy, precision, recall, F1-score, MCC, and AUC for each model. The best-performing models are Xception and Inception V3, both achieving an accuracy of over 91%. For the rest pre-trained models, ResNet101, DenseNet121 and YOLO v5 also get very competitive performance with high accuracy values of 89.87%, 89.57% and 88.69%, respectively. VGG19, MobileNetV2 and ConvNeXt rank at the third place, with classification accuracy values of 87.96%, 87.81% and 87.22%, respectively. NASNet performs the worst, with an accuracy of 86.49%. Fig. 7 (b) provides the training time for each model. The most computationally efficient model is MobileNetV2, with a training time of 2.85 h. Inception V3 takes 3.77 h for training, making it the second fastest model. On the other hand, VGG19, ResNet101 and ConvNeXt require large amount of time for training, i.e., 20.14 h, 17.65 h and 43.60 h, making them less computationally efficient. While Xception performs similarly to Inception V3 in terms of accuracy, it takes 9.32 h for training, making it less computationally efficient than Inception V3. YOLO v5 requires approximately 9.89 h to complete its training process, which is slightly more than the training time of Xception.

The proposed VAECL approach has been compared to three existing methods for recognizing and classifying vessel types using optical images of vessels, namely FRCIR (Voinov et al., 2018), CFCCNN Chen et al. (2020b), and hyperparameter optimized CNN (HPOCNN) (Salem et al., 2023). The comparison results can be found in Table 8. In all the evaluation metrics, VAECL demonstrates superior performance compared to the other baseline methods. Among the three baseline approaches, FRCIR achieves the highest performance with an accuracy of 89.43% and an AUC of 98.12%. However, VAECL surpasses FRCIR by at least 1.76% in accuracy and 0.84% in AUC.

4.7. Sensitivity Analyses (Answering RQ4)

The parameters of VAECL, such as l_{fz}, δ_{or}, BS, h_1, and so on, are estimated empirically. To test the robustness of VAECL, we also provide the performance with different parameters, including freezing layer (l_{fz}), oversampling ratio (δ_{or}), batch size (BS), pre-training epoch (PT-EP), hidden unit (h_1) and margin (m). As shown in Fig. 8 (a), the performance of VAECL increases with the freezing layer from 169 to 229, and obviously deteriorates when it exceeds 229. The best performance is achieved when the freezing layer is 229 with an accuracy of 92.66% and an AUC of 99.02%. The VAECL is not quite sensitive to the oversampling ratio, as the magnitude for all evaluation metrics does not fluctuate greatly. The best performance is achieved when δ_{or} is set as 30 with an accuracy of 92.36%. For the parameter of batch size, when the BS exceeds 128, the performance of VAECL tends to drop slightly, as depicted in Fig. 8 (c). Overall, as shown in Fig. 8 (d), the performance enhances with more pre-training epochs applied. However, when it is over an optimal pre-training epoch (i.e., 40), the performance deteriorates. The best performance is achieved, with an accuracy of 92.95%, MCC of 90.62% and AUC of 99.02%, when the PT-EP is set as 40. In a similar way, the performance increases as the hidden unit increases. However, when it exceeds 512, the performance starts to decrease. Fig. 8 (f) presents the performance comparison with different margin (m) settings in the contrastive loss. The performance tends to improve slightly when m increases from 2 to 8. The performance reaches a peak when $m = 8$, and it starts to decline when $m > 8$. The accuracy with the best margin can reach 92.51%. In summary, the performance of VAECL is the most sensitive to the setting of freezing layer, as it fluctuates the most significantly (i.e., accuracy fluctuates between

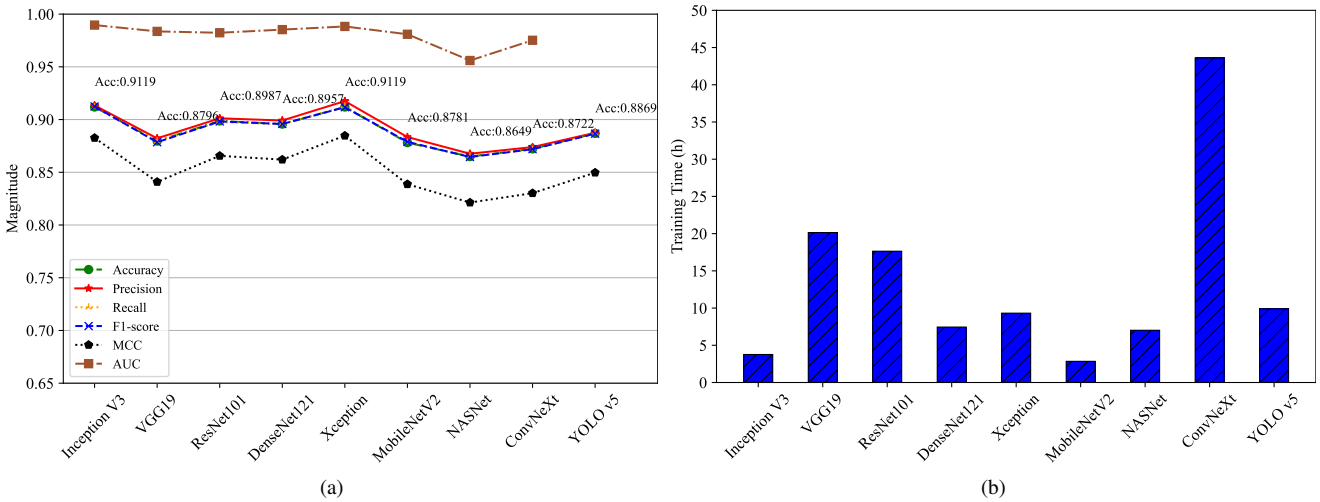


Figure 7: Comparison between VAECL using pre-trained Inception V3 model and VAECL using other pre-trained CNN models. (a) comparison in terms of accuracy, precision, recall, F1-score, MCC and AUC; (b) comparison in terms of model training time. AUC is not available for YOLO v5. All models compared here are trained using the CPU.

Table 8: Comparison between VAECL and the existing baselines in vessel type recognition

Model	Accuracy	Precision	Recall	F1-score	MCC	AUC
FRCIR (Voinov et al., 2018)	0.8943	0.8945	0.8947	0.8939	0.8595	0.9812
CFCCNN (Chen et al., 2020b)	0.8370	0.8498	0.8498	0.8384	0.7864	0.9616
HPOCNN (Salem et al., 2023)	0.8532	0.8683	0.8538	0.8523	0.8095	0.9799
VAECL (ours)	**0.9119**	**0.9134**	**0.9118**	**0.9124**	**0.8826**	**0.9896**

92.66% (i.e., l_{fz} = 229) and 74.16% (i.e., l_{fz} = 309)). For the remaining parameters, the performance is not particularly sensitive, with the largest accuracy swing occurring between 92.95% (i.e., PT-EP= 40) and 89.13% (i.e., PT-EP= 50).

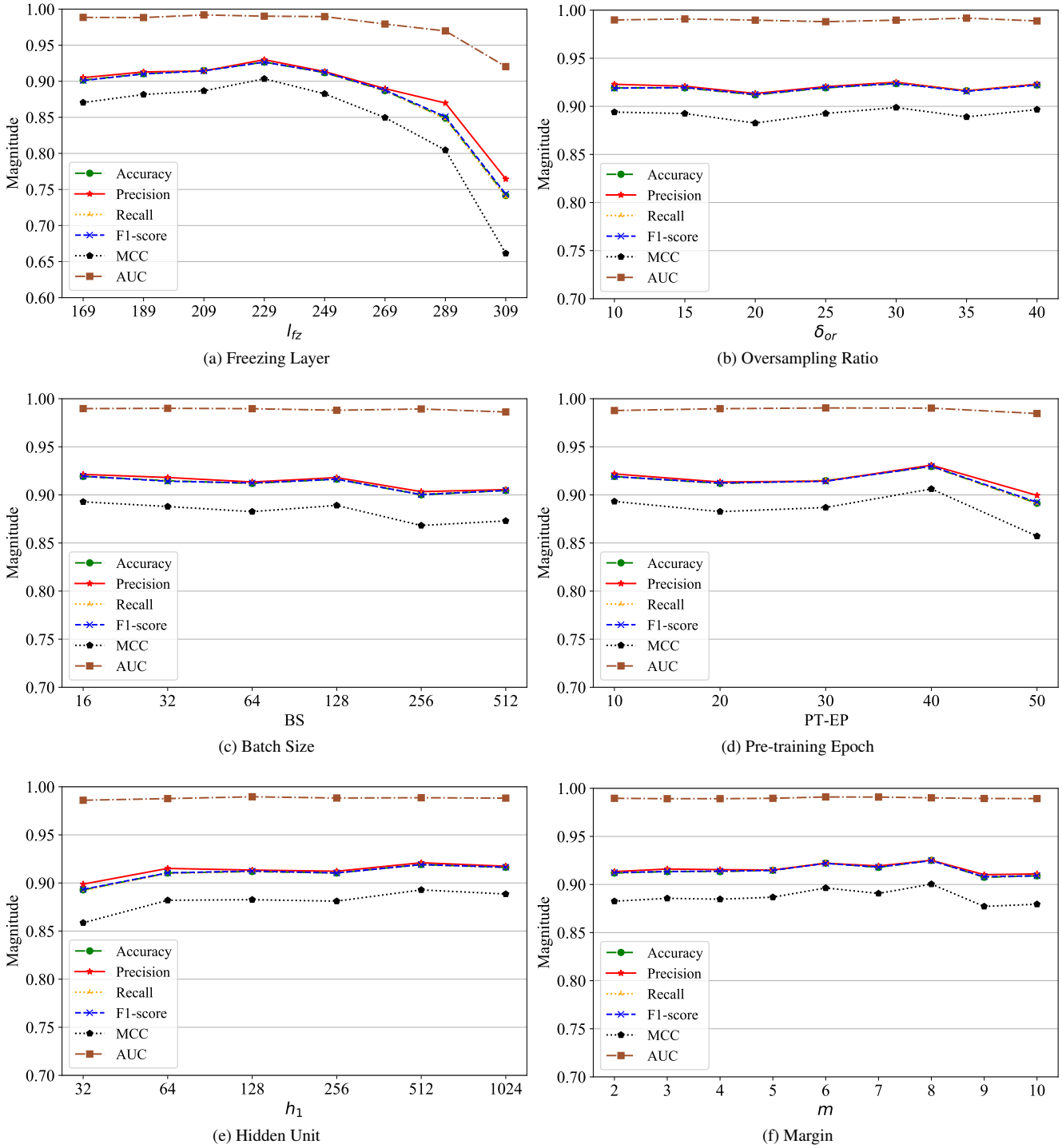
5. Conclusions and Future Work

This work proposes a novel viewpoint adaptation ensemble contrastive learning (VAECL) framework for vessel type recognition with limited training data. The fundamental idea behind VAECL is to utilize an improved deep generative model (i.e., C-WDCGAN-GP) to learn the distribution of vessel image data, and more importantly, to synthesize more images for addressing the shortcoming of limited training data. The transfer learning approach using a pre-trained Inception V3 network is then adopted for vessel viewpoint transfer and adaptation learning. A contrastive loss function is contributed to improve the discriminative capability of classifier by obtaining more contrastive feature representation. Finally, an ensemble learning algorithm based on different image resolutions is constructed to further improve the performance of vessel type recognition. The proposed VAECL model is evaluated using real-world vessel images collected and annotated by ourselves. The VAECL can achieve high accuracy of 91.19% even with empirically estimated parameters, and it can potentially reach 92.96%, exhibiting potential for utilization in various maritime industrial applications. Based on comparisons from different perspectives, C-WDCGAN-GP-based image augmentation improves the accuracy by at least 14.68%, and it performs better than the traditional data augmentation technique. Viewpoint transfer increases the accuracy by as high as 18.50%. Meanwhile, the constructed contrastive learning is effective with an accuracy increment of at least 2.79%. Ensemble learning enhances the accuracy by as high as 1.03%. Moreover, Inception V3 is more advanced than other popular pre-trained CNN models, including VGG19, ResNet101, DenseNet121, Xception, MobileNetV2, NASNet, ConvNeXt and YOLO v5, in terms of recognition accuracy as well as computational cost. The proposed VAECL demonstrates better performance compared to other established baseline approaches in ship type recognition.

In future work, we plan to enhance the DVTR dataset by gathering a wider variety of vessel images to enhance fine-grained recognition capabilities. Furthermore, we intend to assess our recognition model using a broader range of vessel types. Additionally, we will investigate and compare more advanced generative deep learning techniques, such as the diffusion model, for the purpose of data augmentation.

CRediT authorship contribution statement

Xiaocai Zhang: Conceptualization, Data curation, Methodology, Software, Project administration, Writing - original draft. **Zhe Xiao:** Conceptualization, Methodology, Software, Writing - review & editing. **Xiuju Fu:** Conceptualization, Methodology, Visualisation, Writing - review & editing. **Xiaoyang Wei:** Conceptualization, Data curation, Writing - review &



(a) Freezing Layer (b) Oversampling Ratio

(c) Batch Size (d) Pre-training Epoch

(e) Hidden Unit (f) Margin

Figure 8: Classification performance of VAECL with different freezing layers (l_{fz}), oversampling ratios (δ_{or}), batch sizes (BS), pre-training epochs (PT-EP), hidden units (h_1), and margins (m). For the setting of margin (m), the entire network cannot be converged when $m < 2$.

editing. **Tao Liu:** Conceptualization, Methodology, Writing - review & editing. **Ran Yan:** Conceptualization, Software, Writing - review & editing. **Zheng Qin:** Conceptualization, Writing - review & editing. **Jianjia Zhang:** Conceptualization, Methodology, Writing - review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was supported in part by Maritime AI Research Programme (grant number SMI-2022-MTP-06 funded by Singapore Maritime Institute).

References

Agata, K.-G. (2021). The use of drones in the maritime sector–areas and benefits. *Scientific Journals of the Maritime University of Szczecin*, .

Ahmed, M., Afreen, N., Ahmed, M., Sameer, M., & Ahamed, J. (2023). An inception v3 approach for malware classification using machine learning and transfer learning. *International Journal of Intelligent Networks*, *4*, 11–18.

Ahmed, M., Masood, S., Ahmad, M., & Abd El-Latif, A. A. (2021). Intelligent driver drowsiness detection for traffic safety based on multi cnn deep model and facial subsampling. *IEEE Transactions on Intelligent Transportation Systems*, .

Al Husaini, M. A. S., Habaebi, M. H., Gunawan, T. S., Islam, M. R., Elsheikh, E. A., & Suliman, F. (2022). Thermal-based early breast cancer detection using inception v3, inception v4 and modified inception mv4. *Neural Computing and Applications*, *34*, 333–348.

Buscema, P. M., Massini, G., Raimondi, G., Caporaso, G., Breda, M., & Petritoli, R. (2023). A pattern recognition analysis of vessel trajectories. *Algorithms*, *16*, 414.

Chen, X., Qi, L., Yang, Y., Luo, Q., Postolache, O., Tang, J., & Wu, H. (2020a). Video-based detection infrastructure enhancement for automated ship recognition and behavior analysis. *Journal of Advanced Transportation*, *2020*.

Chen, X., Yang, Y., Wang, S., Wu, H., Tang, J., Zhao, J., & Wang, Z. (2020b). Ship type recognition via a coarse-to-fine cascaded convolution neural network. *The Journal of Navigation*, *73*, 813–832.

Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the 2017 IEEE conference on computer vision and pattern recognition* (pp. 1251–1258).

Frederiksen, M. H., & Knudsen, M. P. (2018). Drones for offshore and maritime missions: Opportunities and barriers. *Innovation Fund Denmark*, .

Gao, X., Deng, F., & Yue, X. (2020). Data augmentation in fault diagnosis based on the wasserstein generative adversarial network with gradient penalty. *Neurocomputing*, *396*, 487–494.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial nets. *Advances in Neural Information Processing Systems*, *27*.

Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J. et al. (2018). Recent advances in convolutional neural networks. *Pattern Recognition*, *77*, 354–377.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., & Courville, A. C. (2017). Improved training of wasserstein gans. *Advances in Neural Information Processing Systems*, *30*.

Haijoub, A., Hatim, A., Arioua, M., Hammia, S., Eloualkadi, A., & Guerrero-González, A. (2023). Fast yolo v7 based cnn for video streaming sea ship recognition and sea surveillance. In *Modern Artificial Intelligence and Data Science: Tools, Techniques and Systems* (pp. 99–109). Springer.

Han, Y., Yang, X., Pu, T., & Peng, Z. (2021). Fine-grained recognition for oriented ship against complex scenes in optical remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, .

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the 2016 IEEE conference on computer vision and pattern recognition* (pp. 770–778).

Hou, X., Ao, W., Song, Q., Lai, J., Wang, H., & Xu, F. (2020). Fusar-ship: building a high-resolution sar-ais matchup dataset of gaofen-3 for ship detection and recognition. *Science China Information Sciences*, *63*, 1–19.

Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In *Proceedings of the 2017 IEEE conference on computer vision and pattern recognition* (pp. 4700–4708).

Huang, H., Hong, F., Liu, J., Liu, C., Feng, Y., & Guo, Z. (2018). Fvid: Fishing vessel type identification based on vms trajectories. *Journal of Ocean University of China*, (pp. 1–10).

Jocher, G., Chaurasia, A., Stoken, A., Borovec, J., Kwon, Y., Michael, K., Fang, J., Wong, C., Yifu, Z., Montes, D. et al. (2022). ultralytics/yolov5: v6. 2-yolov5 classification models, apple m1, reproducibility, clearml and deci. ai integrations. *Zenodo*, .

Karabayır, O., Yücedağ, O. M., Kartal, M. Z., & Serim, H. A. (2017). Convolutional neural networks-based ship target recognition using high resolution range profiles. In *Proceedings of 2017 18th international radar symposium* (pp. 1–9). IEEE.

Lampe, J., & Hamann, R. (2018). Probabilistic model for corrosion degradation of tanker and bulk carrier. *Marine Structures*, *61*, 309–325.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*, 436–444.

Liu, B., Wang, S. Z., Xie, Z., Zhao, J., & Li, M. (2019). Ship recognition and tracking system for intelligent ship based on deep learning framework. *TransNav: International Journal on Marine Navigation and Safety of Sea Transportation*, *13*.

Liu, T., Jia, Z., Lei, Z., Zhang, X., & Huo, Y. (2022a). Unsupervised depth estimation for ship target based on single view uav image. *International Journal of Remote Sensing*, *43*, 3216–3235.

Liu, X., Li, Y., Wu, Y., Wang, Z., He, W., & Li, Z. (2021). A hybrid method for inland ship recognition using marine radar and closed-circuit television. *Journal of Marine Science and Engineering*, *9*, 1199.

Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., & Xie, S. (2022b). A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 11976–11986).

Marzuki, M. I., Gaspar, P., Garello, R., Kerbaol, V., & Fablet, R. (2017). Fishing gear identification from vessel-monitoring-system-based fishing vessel trajectories. *IEEE Journal of Oceanic Engineering*, *43*, 689–699.

Meng, H., Tian, Y., Ling, Y., & Li, T. (2022). Fine-grained ship recognition for complex background based on global to local and progressive learning. *IEEE Geoscience and Remote Sensing Letters*, *19*, 1–5.

Messinger, M., & Silman, M. (2016). Unmanned aerial vehicles for the assessment and monitoring of environmental contamination: An example from coal ash spills. *Environmental Pollution*, *218*, 889–894.

Ren, Y., Yang, J., Zhang, Q., & Guo, Z. (2021). Ship recognition based on hu invariant moments and convolutional neural network for video surveillance. *Multimedia Tools and Applications*, *80*, 1343–1373.

Salem, M. H., Li, Y., Liu, Z., & AbdelTawab, A. M. (2023). A transfer learning and optimized cnn based maritime vessel classification system. *Applied Sciences*, *13*, 1912.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the 2018 IEEE conference on computer vision and pattern recognition* (pp. 4510–4520).

Santos, C. E., & Bhanu, B. (2018). Dyfusion: dynamic ir/rgb fusion for maritime vessel recognition. In *Proceedings of 2018 25th IEEE international conference on image processing* (pp. 1328–1332). IEEE.

Saufi, S. R., Ahmad, Z. A. B., Leong, M. S., & Lim, M. H. (2020). Gearbox fault diagnosis using a deep learning model with limited data sample. *IEEE Transactions on Industrial Informatics*, *16*, 6263–6271.

Shen, S., Yang, H., Yao, X., Li, J., Xu, G., & Sheng, M. (2020). Ship type classification by convolutional neural networks with auditory-like mechanisms. *Sensors*, *20*, 253.

Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition, . (pp. 1–14).

Su, J., Zhu, X., Li, S., & Chen, W.-H. (2022). Ai meets uavs: A survey on ai empowered uav perception systems for precision agriculture. *Neurocomputing*, .

Sun, S., Gu, Y., & Ren, M. (2022a). Fine-grained ship recognition from the horizontal view based on domain adaptation. *Sensors*, *22*, 3243.

Sun, Z.-H., Luo, X., Wu, E. Q., Zuo, T.-Y., Tang, Z.-R., & Zhuang, Z. (2022b). Monitoring scheduling of drones for emission control areas: An ant colony-based approach. *IEEE Transactions on Intelligent Transportation Systems*, *23*, 11699–11709.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2818–2826).

Voinov, S., Krause, D., & Schwarz, E. (2018). Towards automated vessel detection and type recognition from vhr optical satellite images. In *Proceedings of the 2018 IEEE international geoscience and remote sensing symposium* (pp. 4823–4826). IEEE.

Wang, F., Liang, H., Zhang, Y., Xu, Q., & Zong, R. (2022). Recognition and classification of ship images based on sms-pcnn model. *Frontiers in Neurorobotics*, *16*.

Wang, S., Liu, Y., Xu, Y., Zhu, C., & Zeng, M. (2021a). Want to reduce labeling cost? gpt-3 can help. *arXiv preprint arXiv:2108.13487*, .

Wang, X., & Qi, G.-J. (2022). Contrastive learning with stronger augmentations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, .

Wang, X., Zhang, R., Shen, C., Kong, T., & Li, L. (2021b). Dense contrastive learning for self-supervised visual pretraining. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 3024–3033).

Wu, Y.-L., Shuai, H.-H., Tam, Z.-R., & Chiu, H.-Y. (2021). Gradient normalization for generative adversarial networks. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 6373–6382).

Xu, L., Chen, X., Wen, B., Ma, J., Wang, Y., & Xu, Q. (2023). Ship classification based on trajectories data and lightgbm considering offshore distance feature. In *International Conference on Spatial Data and Intelligence* (pp. 115–127). Springer.

Yang, C.-S. (2019). Investigating uavs applications and intention to use in the maritime shipping in taiwan. *Maritime Policy & Management*, *46*, 982–994.

Yang, J., Liu, J., Xie, J., Wang, C., & Ding, T. (2021). Conditional gan and 2-d cnn for bearing fault diagnosis with small samples. *IEEE Transactions on Instrumentation and Measurement*, *70*, 1–12.

Yang, T., Wang, X., & Liu, Z. (2022). Ship type recognition based on ship navigating trajectory and convolutional neural network. *Journal of Marine Science and Engineering*, *10*, 84.

Yildirim, M. E. (2023). Ship type recognition using deep learning with fft spectrums of audio signals. *El-Cezeri*, *10*, 57–65.

Zhang, S., Zhang, J., Pei, K., Tang, X., Hou, J., Tang, F., Yang, S., & Zhang, H. (2021). Fishing vessel type recognition based on ship position data. In *Proceedings of 2021 IEEE 4th international conference on electronics and communication engineering* (pp. 93–97). IEEE.

Zhang, X., Zhao, Z., Zheng, Y., & Li, J. (2020). Prediction of taxi destinations using a novel data embedding method and ensemble learning. *IEEE Transactions on Intelligent Transportation Systems*, *21*, 68–78.

Zhao, W., Syafrudin, M., & Fitriyani, N. L. (2023). Crasyolo: A novel multi-category vessel detection and classification model based on yolov5s algorithm. *IEEE Access*, *11*, 11463–11478.

Zheng, J., Cao, J., Yuan, K., & Liu, Y. (2023). A small fishing vessel recognition method using transfer learning based on laser sensors. *Scientific Reports*, *13*, 5931.

Zheng, Y., Peng, H., Zhang, X., Gao, X., & Li, J. (2018). Predicting drug targets from heterogeneous spaces using anchor graph hashing and ensemble learning. In *Proceddings of 2018 international joint conference on neural networks* (pp. 1–7). IEEE.

Zhu, J., Qiu, X., Pan, Z., Zhang, Y., & Lei, B. (2016). Projection shape template-based ship target recognition in terrasar-x images. *IEEE Geoscience and Remote Sensing Letters*, *14*, 222–226.

Zoph, B., Vasudevan, V., Shlens, J., & Le, Q. V. (2018). Learning transferable architectures for scalable image recognition. In *Proceedings of the 2018 IEEE conference on computer vision and pattern recognition* (pp. 8697–8710).