

Article

LFLD-CLbased NET: A Curriculum-Learning-Based Deep Learning Network with Leap-Forward-Learning-Decay for Ship Detection

Jiawen Li ^{1,2,3} , Jiahua Sun ¹ , Xin Li ¹ , Yun Yang ^{1,4} , Xin Jiang ¹  and Ronghui Li ^{1,2,3,*} 

¹ Naval Architecture and Shipping College, Guangdong Ocean University, Zhanjiang 524005, China; jiawen-li@gdou.edu.cn (J.L.); jiahuasun@stu.gdou.edu.cn (J.S.); lxl1@stu.gdou.edu.cn (X.L.); yangyun5@stu.gdou.edu.cn (Y.Y.); xinjiang@stu.gdou.edu.cn (X.J.)

² Technical Research Center for Ship Intelligence and Safety Engineering of Guangdong Province, Zhanjiang 524005, China

³ Guangdong Provincial Key Laboratory of Intelligent Equipment for South China Sea Marine Ranching, Zhanjiang 524005, China

⁴ College of Civil and Transportation Engineering, Shenzhen University, Shenzhen 518060, China

* Correspondence: lironghui@163.com

Abstract: Ship detection in the maritime domain awareness field has seen a significant shift towards deep-learning-based techniques as the mainstream approach. However, most existing deep-learning-based ship detection models adopt a random sampling strategy for training data, neglecting the complexity differences among samples and the learning progress of the model, which hinders training efficiency, robustness, and generalization ability. To address this issue, we propose a ship detection model called the Leap-Forward-Learning-Decay and Curriculum Learning-based Network (LFLD-CLbased NET). This model incorporates innovative strategies as Leap-Forward-Learning-Decay and curriculum learning to enhance its ship detection capabilities. The LFLD-CLbased NET is composed of ResNet as the feature extraction unit, combined with a difficulty generator and a difficulty scheduler. The difficulty generator in LFLD-CLbased NET effectively expands data samples based on real ocean scenarios, and the difficulty scheduler constructs corresponding curriculum training data, enabling the model to be trained in an orderly manner from easy to difficult. The Leap-Forward-Learning-Decay strategy, which allows for flexible adjustment of the learning rate during curriculum training, is proposed for enhancing training efficiency. Our experimental findings demonstrate that our model achieved a detection accuracy of 86.635%, approximately 10% higher than other deep-learning-based ship detection models. In addition, we conducted extensive supplementary experiments to evaluate the effectiveness of the learning rate adjustment strategy and curriculum training in ship detection tasks. Furthermore, we conducted exploratory experiments on different modules to compare performance differences under varying parameter configurations.



Citation: Li, J.; Sun, J.; Li, X.; Yang, Y.; Jiang, X.; Li, R. LFLD-CLbased NET: A Curriculum-Learning-Based Deep Learning Network with Leap-Forward-Learning-Decay for Ship Detection. *J. Mar. Sci. Eng.* **2023**, *11*, 1388. <https://doi.org/10.3390/jmse11071388>

Academic Editor: Mohamed Benbouzid

Received: 6 June 2023

Revised: 29 June 2023

Accepted: 6 July 2023

Published: 8 July 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Ship detection, as one of the most important intelligent maritime perception technologies, has recently attracted significant interest from researchers and scholars in ocean surveillance [1–3]. With the development of artificial intelligence, deep-learning-based ship detection methods are now dominating the maritime domain awareness (MDA) field due to their remarkable data fitting ability [4,5], which has achieved a breakthrough in the accuracy of ship identification compared to the traditional methods.

However, most of the deep-learning-based ship detection methods are trained based on a traditional training mode [6–9], i.e., all the monitoring examples are randomly presented to the model, neglecting the intricacies of the data samples and the current model's learning progress, and inevitably lead to low model robustness and low generalization performance.

Furthermore, the oceanic environment is characterized by its dynamic, high-dimensional, and combinatorially complex nature. The presence of diverse factors, such as fog, clouds, rain, currents, and clutter, further complicates the detection process. Meanwhile, training a deep learning model requires training data, and those training datasets are subject to various historical, representative, measurement, aggregation, and evaluation biases, thereby creating discrepancies between the acquired data and the actual oceanic environment. The aforementioned issue raises higher demands on the robustness and generalization ability of current deep-learning-based ship detection models. Expanding the training dataset is a potential approach to enhance the generalization ability and robustness of a deep-learning-based model. However, it should be noted that this method is not without its challenges, as data collection is a labor-intensive and time-consuming task that requires significant financial and material support to obtain a large number of labeled data [10]. Furthermore, it is important to recognize that increasing the dataset is not a panacea for improving the model's robustness and generalization ability. Rather, the diversity of data is the key factor that can enhance the model's ability to generalize and be robust. Therefore, relying solely on expanding the dataset to improve the model's robustness is not a wise strategy.

Hence, in order to make up for the issues of low robustness and low generalization performance in current deep-learning-based ship detection methods, a novel network, LFLD-CLbased NET, is proposed for monitoring ships in real maritime scenarios.

The LFLD-CLbased NET adopts data augmentation techniques purposefully to construct a diversified training dataset that matches the real maritime environment, establishing a communication platform between the real and training environments. Then, the LFLD-CLbased NET incorporates the principles of curriculum learning (CL), organizing the extended training dataset in a meaningful order, by "starting small" and gradually presenting more complex concepts to acquire greater generalization and robustness in ship detection. Additionally, we propose a Leap-Forward-Learning-Decay (LFLD) strategy to the LFLD-CLbased NET that dynamically adjusts the difficulty and learning rate of curriculum learning, enhancing the model's learning speed, reducing training time, and mitigating the occurrence of gradient explosion.

The experimental results on a real world ship monitoring dataset reveal that LFLD-CLbased NET achieves a state-of-the-art ship identification accuracy and outperforms the modern deep-learning-based methods. Furthermore, we conducted a series of supplementary experiments to assess the efficacy of both the curriculum learning and the Leap-Forward-Learning-Decay strategy. In general, we clarify the contributions of our work as follows:

- One of the main contributions is that we propose a new deep-learning-based model that combines the curriculum learning mechanism for real-scenario ship detection. The curriculum learning mechanism is essentially more suitable for making model generalization ability stronger and the generated proposals more realistic.
- We deliberate design of a novel learning rate decay strategy, Leap-Forward-Learning-Decay strategy, facilitating the expeditious and efficient training of a model. We further evaluate the performance of the Leap-Forward-Learning-Decay strategy, and the supplementary experimental results indicate that it effectively aids in training the model and can even enhance the model's accuracy.
- Another contribution is that we collected a small real maritime ship detection dataset. This dataset is more realistic compared to current ship detection datasets, and is able to reflect the robustness and generalization ability of models.
- Experiments show that LFLD-CLbased NET achieves the highest accuracy and outperforms the current deep-learning based models, achieving 86.64% accuracy in ship detection tasks, significant improving 10% detection accuracy.

The remainder of this article is organized as follows: Section 2 reviews the related work, Section 3 describes the details of LFLD-CLbased NET, the experimental situation and analysis results are discussed in Section 4, and Section 5 concludes with a summary of the findings and future work.

2. Related Work

2.1. Deep-Learning-Based Ship Monitoring Methods

The burgeoning advancement of artificial intelligence has propelled the widespread implementation of deep learning models for ship detection tasks [11–14]. Wu et al. [15], utilizing convolutional neural network (CNN) architecture, started from a search for the relatively distinct ship head, enabling precise determination of ship head location and estimation of approximate vessel directions. Nie et al. [16] used SSD as the basic architecture to construct different layers of feature maps to distinguish between different types of ships, which greatly improved the prediction accuracy. Ma et al. [17] added a fully convolutional network (FCN) as an attention branch to extract ship and ocean environment features, thereby improving ship prediction accuracy, based on the SSD model. Cui et al. [18] utilized a dense attention pyramid network which allows for the extraction of a wide range of resolution and semantic information, resulting in the implementation of high-precision ship monitoring. Chen et al. [19] discussed a novel method for detecting small ships in remote sensing images using a degraded reconstruction enhancement-based approach. Ren et al. [20] devised two novel modules, feature-enhanced structure (FES) and saliency prediction branch (SPB), to enhance the ship detection capabilities in intricate settings. Additionally, a new sampling strategy called the salient screening mechanism (SSM) was proposed to augment the quantity of positive samples. Zhang et al. [21] proposed a state-of-the-art deep learning network, YOLO-FA, for the efficient processing of SAR images. YOLO-FA is equipped with a novel frequency attention module (FAM), which enables the adaptive processing of frequency-domain information. The FAM has been designed to enhance the quality of image processing by selectively attending to frequency components of interest. This innovative approach has the potential to significantly improve the accuracy and efficiency of SAR image analysis. Zhang et al. [22] proposed a novel sensor-related image synthesis framework, dubbed remote sensing-image synthesis pipeline (RS-ISP), which was developed in response to the scarcity of on-orbit remote sensing imagery. Zhang et al. [23] used the feature of DCT blocks to extract horizon information for efficient ship detection. Li et al. [3] proposed a bidirectional ship monitoring system that utilizes satellite devices and near-shore surveillance cameras based on knowledge transfer.

The appeals approach offers significant potential for improving ship detection accuracy and represents a valuable contribution to the field of maritime surveillance. There are a few researchers who have conducted research on the robustness and generalization ability of deep-learning-based ship detection models. Nie et al. [24] stated that in adverse weather conditions, the acquired images often suffer from degraded visual quality, which may have negative impacts on target detection in practical applications. To address this issue, they utilized synthetically degraded images to expand the training dataset and employed YOLOv3 to ensure robust ship detection. Liu et al. [1] developed an enhanced convolutional neural network (CNN) that incorporates soft nonmaximum suppression and a reconstructed mixed loss function, resulting in improved ship detection under various weather conditions. Those two studies explored the robustness of models from the perspective of weather changes, which are insufficiently explored in the diversity of data. At the same time, they were all discussed from the perspective of object detection tasks, and research on the robustness and generalization ability of deep-learning-based ship detection models for ship type recognition is still lacking.

2.2. Curriculum Learning

The technique of curriculum learning is a highly effective means of training deep learning models, enhancing their generalization capabilities, and improving their performance on new data [25,26]. Bengio et al. [27] first proposed the concept of curriculum learning to the field of artificial intelligence. The curriculum learning training strategy involves progressively increasing the difficulty level of the data or subtasks, beginning with simpler subsets and gradually incorporating more complex ones until the entire training dataset or target task(s) is covered.

The effectiveness of curriculum learning in improving the efficiency of deep learning model training has led scholars to widely apply this approach across various domains. It has been applied in supervised learning tasks in natural language processing (NLP) [28,29], healthcare prediction [30], reinforcement learning (RL) tasks [31–33], graph learning [34,35], and neural architecture search (NAS) [36].

Curriculum learning has also found applications in the field of computer vision [37,38]. Hacohen et al. [39] conducted an analysis of the impact of curriculum learning on the training of deep networks, with a focus on CNNs trained for image recognition. Their experiments provided further evidence of the effectiveness of curriculum learning in this domain. Mosavi et al. [40] employed a curriculum learning strategy to train a convolutional neural network, achieving effective classification of fully polarimetric synthetic aperture radar (PolSAR) data. Inspired by the concept of curriculum learning, Wang et al. [41] proposed a unified framework, dynamic curriculum learning (DCL), which adaptively adjusts the sampling strategy and loss weight in each batch. This approach enhances the ability of generalization and discrimination in the task of human attribute analysis. Wang et al. [42] employed the curriculum learning strategy to train a Faster R-CNN using weakly and semisupervised data. Their research revealed that curriculum learning significantly reduces the labeling efforts required to obtain reliable object detectors. Goyal et al. [43] revealed that the curriculum-learning-based approach demonstrates a considerable improvement in recall compared to non-curriculum-learning methods, underscoring its robustness in addressing false negatives that arise from overlapping predictions of motorcycle and rider boxes.

Currently, there is a paucity of research on the automatic ship detection model utilizing curriculum learning technology. Nevertheless, given the impressive accomplishments of curriculum learning in various other domains, it is reasonable to posit that its capacity to facilitate progress in the present ship detection model, which relies on deep learning, is significant. Consequently, we proactively leveraged this potential by applying curriculum learning to the vessel monitoring field.

3. LFLD-CLbased NET

This paper introduces a novel Leap-Forward-Learning-Decay neural network, namely, LFLD-CLbased NET, which integrates curriculum learning for ship detection in a more realistic scenario. The LFLD-CLbased NET is a two-step pipeline model that consists of curriculum generation phase and curriculum learning phase. Figure 1 shows the details of LFLD-CLbased NET. The LFLD-CLbased NET is founded upon the ResNet34 framework and employs a difficulty generator to produce a range of realistic and varied sample data. The resultant data are systematically arranged and converted into curricula through the assistance of a difficulty scheduler. The curriculum datasets are subsequently arranged in a manner that aligns with the learning progression of LFLD-CLbased NET, thereby enabling a structured course training and optimizing the performance of curriculum learning. During the training phase, the LFLD mechanism was integrated and a learning rate decay factor was devised to accommodate varying levels of learning complexity. By employing a Leap-Forward of the learning rate, the data information is continuously modified, thereby mitigating the issues of slow convergence rate and local optima entrapment. The capacity of the model to explore local optima is strengthened, and its generalization performance is enhanced, making the LFLD-CL-based NET an effective approach for maritime monitoring.

Subsequently, a comprehensive exposition of the model is presented, encompassing the problem statement, curriculum generation phase, curriculum learning, and LFLD mechanism.

3.1. Problem Statement

In this study, we approached the task as a supervised classification problem, with the objective of recognizing images of ship types. The model was designed to classify the type of ship shown in an outboard profile image captured by optical surveillance cameras.

Within the provided dataset, $D = \{(x_i, y_i)\}_{i=1}^N$, x_i corresponds to the i th outboard profile image, while N signifies the aggregate quantity of images in the dataset. The primary aim of this endeavor is to prognosticate the classification of ship, which is identified by the label y_i . The task's output was characterized as $f_{ship-detect} : x_i \rightarrow \{y_i\}$.

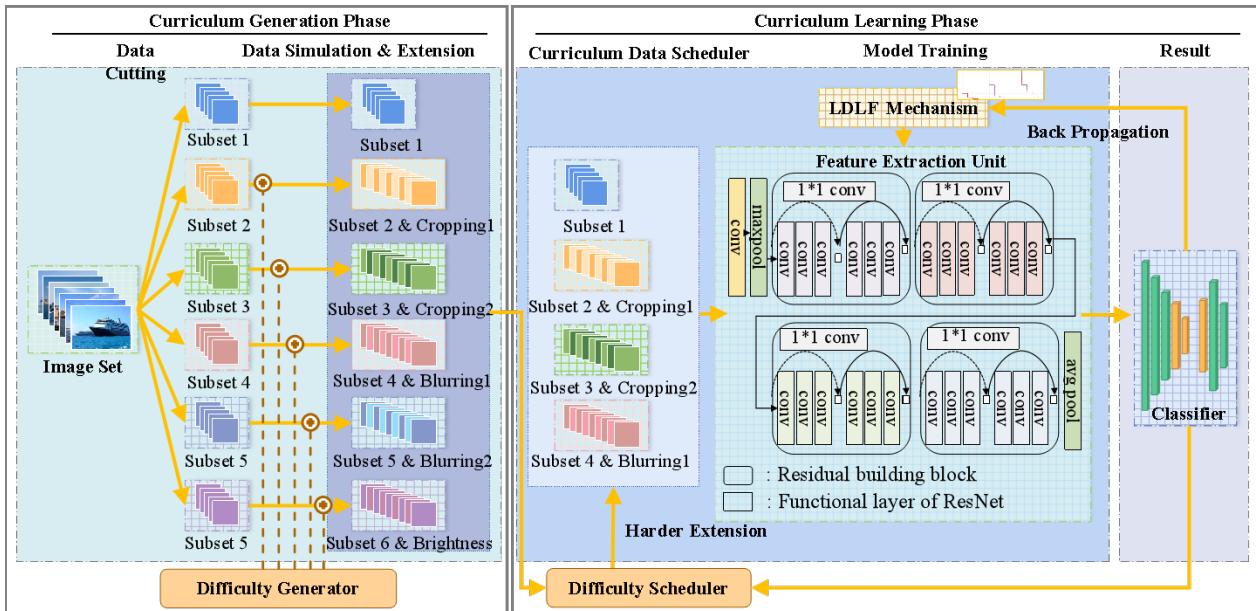


Figure 1. The architecture of LFDF-CLbased NET.

3.2. Curriculum Generation Phase

The curriculum generation phase is the preprocess of curriculum training. During this phase, the original outboard profile ship detection dataset is segmented into multiple independent subsets, and then the subsets are sequentially fed into a difficulty generator for data transformation. The difficulty generator is a module consisting of random cropping unit, noise addition unit, and brightness adjustment unit, which are composed of data augmentation techniques.

The random cropping unit performs image cropping to enhance the model's ability to process images with only a small portion of ship features. The formula of random cropping unit is

$$x_{crop} = f_{crop}(x), \quad (1)$$

where x represents a original image, and x_{crop} denotes the image obtained by applying a cropping function represented by f_{crop} .

f_{crop} is expressed as

$$f_{crop} = random_crop(x, size), \quad (2)$$

where $size$ represents the size of the clipped image.

The noise addition unit is utilized to simulate noise interference and enhance the model's robustness against noise. The commonly used noise distributions include Gaussian noise, salt noise, and so on. Taking salt noise addition as an instance, the salt noise can be regarded as the random addition of white pixels to an image. The pixel points in the original image x can be randomly replaced with white pixels. The density of salt noise can be represented as p , which is typically within the range of $[0, 1]$. The method of adding salt noise is mathematically described as follows:

$$x_{noisy}(i, j, k) = (0, with probability of p), \quad (3)$$

where $x(i, j, k)$ represents the pixel value of the i th channel in the j th row and k th column of the image. It should be noted that the density value is able to affect the quality of the

generated data. When the value of p is too high, the image may be overly distorted by noise, which can negatively affect the performance of the model. Conversely, when the value of p is too low, the diversity of the data may not be effectively increased.

The brightness adjustment unit enhances the diversity of the data by adjusting the brightness of the image, and the formula is as follows:

$$x_{bright}(i, j, k) = \left\{ x_{i,j,k} + \Delta, 0 < x_{i,j,k} + \Delta \leq 255 \right\}, \quad (4)$$

where Δ represents the adjusted value of brightness.

The difficulty generator serves as a bridge between real-world data and experimental data, and the data processed through difficulty generator obtain diversity. Its detailed operations are described in Algorithm 1.

Algorithm 1 Detection Difficulty Generator Algorithm.

Input: Outboard profile ship detection dataset $D = \{(x_i, y_i)\}_{i=1}^N$, hyper-parameters $num, size_list, p, \Delta, flag_list$

Require: Dataset Cutting function $Cutting()$; Random Cropping function, $RandomCropping()$; Noise Addition function, $AddNoising()$; Brightness Adjustment function, $BrightnessAdjustment()$.

```

1: # obtain multiple subsets
2:  $D^i, \forall i \in \{1, \dots, num\} \leftarrow Cutting(D, num)$ 
3: for  $j$  in  $1, \dots, num$  do
4:   flag = flag_list[j], # flag_list is a hyperparameter of the data enhancement type
5:   if flag==Cutting then
6:      $\tilde{D}^j \leftarrow Cutting(D^j)$ 
7:   if flag==Cropping then
8:      $\tilde{D}^j \leftarrow RandomCropping(D^j, size\_list)$ 
9:   if flag==Noising then
10:     $\tilde{D}^j \leftarrow AddNoising(D^j, p)$ 
11:   if flag==Brightness then
12:      $\tilde{D}^j \leftarrow BrightnessAdjustment(D^j, \Delta)$ 
13:    $\bar{D}^j = D^j \cup \tilde{D}^j$ 
14: endfor
Output: preprocessed subsets of outboard profile ship detection dataset:  $D^i, \forall i \in \{1, \dots, num\}$ 

```

3.3. Curriculum Learning Phase

The output from the curriculum generation phase was subsequently fed into the difficulty scheduler to generate a training curriculum from easy to difficult, which will be transmitted to a information processing unit for curriculum training. The training process of curriculum learning phase is shown in Figure 2. The complexity of ship monitoring in real sea conditions can lead to poor model performance and low generalization ability when directly training the model for automatic monitoring; however, this problem can be effectively alleviated by utilizing curriculum training.

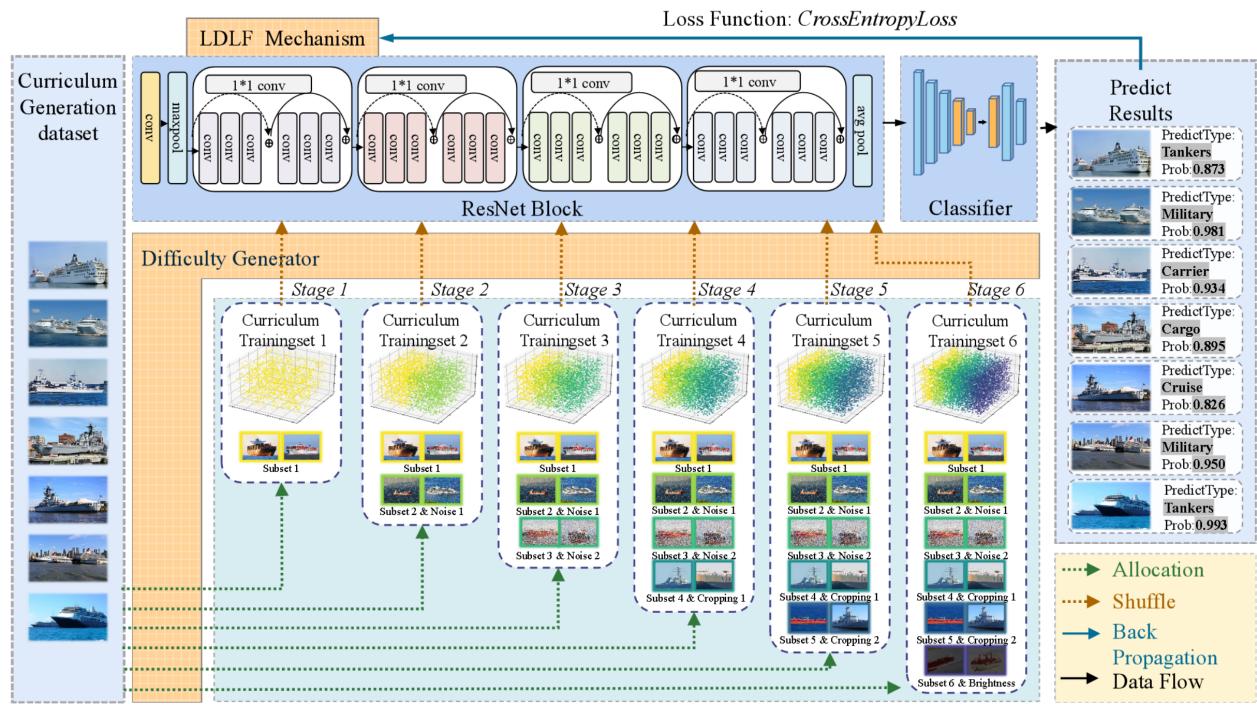


Figure 2. The training process of the curriculum learning phase.

The definition of curriculum learning in this paper is followed by [26]; a curriculum is a sequence of training criteria over T training steps: $C = \{c_1, \dots, c_t, \dots, c_T\}$, and each criterion c_t is a reweighting of the target training distribution $P(z)$.

$$P_t(z) \propto W_t(z)P(z), \quad (5)$$

such that the following three conditions are satisfied:

- (1) The entropy of distributions gradually increases, i.e., $H(c_t) < H(c_{t+1})$.
- (2) The weight for any example increases, i.e., $W_t(z) < W_{t+1}(z), \forall z \in D$.
- (3) $c_T = P(z)$.

In this definition, in accordance with Condition (1), the diversity and information of the training set should progressively increase. This is achieved by increasing the probability of sampling slightly more difficult examples through the reweighting of examples in later steps. Condition (2) involves gradually adding more training examples, either in a binary or soft manner, to expand the size of the training set. Finally, Condition (3) requires uniform reweighting of all examples and training on the target training set. The definition of curriculum learning serves the purpose of providing a framework for designing a training curriculum that gradually increases in difficulty to better facilitate the learning process for a model. It allows for a more effective and efficient training process by introducing examples in a way that is more conducive to learning, and can lead to improved performance and generalization ability of the model.

The difficulty scheduler is a crucial processing unit in curriculum learning phase, which continuously generates a curriculum training set based on the model's training process. The concept of complexity in curriculum learning refers to the level of difficulty or sophistication of the training examples. Specifically, the level of difficulty of the training examples varies depending on the entropy of distributions in the curriculum training set based on Condition (1). In other words, the concentration or variety of the entropy samples in the next stage of the curriculum training set is greater than that in the previous stage. It is worth noting that the new curriculum training set is composed of the fusion and reorganization of the training set from the previous curriculum training and an untrained preprocessed subset.

The feature extraction unit is another important processing unit that is designed based on the principles of deep learning, possessing both learning and recognition capabilities. We utilized a residual network as the underlying framework for this unit.

In the feature extraction unit, the image sample of the curriculum training set was first delivered through a convolution layer for performing feature extraction and obtaining higher level representations. The convolution layer comprises a kernel scanning layer and activation layer that extract advanced information by scanning through kernels and applying an activation function. The most essential features are obtained through a maxpooling layer and are then passed through multiple residual building blocks for deeper extraction. Each residual building block includes a convolution layer, a batch normalization layer, and an activation layer.

A breakdown of the curriculum learning phase applied to the ship detection task is presented in Algorithm 2, which includes essential specifics.

Algorithm 2 Curriculum learning algorithm.

Input: Preprocessed subsets: $\bar{D}^i, \forall i \in \{1, \dots, num\}$
Require: Residual(), Conv(), Maxpool(), Avgpool(), and FC() functions, which encompass the essential elements of residual networks, such as convolution layers, max pooling functions, average pooling functions, and fully connected layers.

```

1: # Begin Curriculum training
2:  $c = []$ 
3: for  $t$  in  $\text{len}(\bar{D})$  do
4:    $c_t \leftarrow c_t \cup \bar{D}^t$ , # obtaining the  $t$ th Curriculum train set
5:    $c_t \leftarrow \text{Conv}(c_t)$ 
6:    $c_t \leftarrow \text{Maxpool}(c_t)$ 
7:   for  $i$  in  $[3, 4, 6, 3]$  do, #  $[3, 4, 6, 3]$  is the number of the residual basic block
8:     for  $j$  in 1 to  $i$  do
9:        $c_t \leftarrow \text{Residual}(c_t)$ 
10:      endfor
11:      endfor
12:       $c_t \leftarrow \text{Avgpool}(c_t)$ 
13:       $c_t \leftarrow \text{FC}(c_t)$ , # Dimensionality reduction
14:      # Classifier:
15:       $\hat{y} \leftarrow \text{softmax}(W \cdot x_i + b), \forall x_i \in c_t$ 
16:    endfor
Ouput: ship detection labels  $\hat{y}_i$ 
```

3.4. LFLD Mechanism

The curriculum learning strategy is able to effectively facilitate the model to achieve high-quality learning, but compared to the traditional training strategy, curriculum learning requires more training time. As can be seen from Algorithm 2, curriculum learning needs to repeatedly train the model based on data subsets, resulting in a multiple-fold increase in training time.

In order to enhance the efficiency of model training, we developed the LFLD Mechanism, integrating the concept of learning rate decay. The LFLD Mechanism is capable of dynamically adapting the learning rate, whereby in the initial stages of course training, the learning rate undergoes a significant increase, facilitating the model to approach the optimal solution rapidly. However, during the middle and later stages of course training, a high learning rate may result in the model experiencing gradient explosion or gradient disappearance, or failing to converge to the optimal solution. Under the influence of the LFLD Mechanism, the learning rate gradually decreases with a fixed step size, thereby precluding the model from encountering issues such as gradient explosion or gradient disappearance during the course of training, and improving the stability of gradient descent.

Learning rate is a hyperparameter in deep learning model training that determines the size of weight updates made by the model during each iteration. The magnitude of the learning rate can affect the speed and effectiveness of model training. The formula for the learning rate is as follows:

$$w_{t+1} = w_t - \eta \nabla_{w_t}, \quad (6)$$

where η means the learning rate, w_t represents the weight value of the t th iteration, and ∇_{w_t} represents the gradient of the loss function with respect to the weight.

As the training progresses, the LFLD Mechanism sets a sequence of increasing initial learning rates, and then adopts a fixed step size linear decay strategy for the learning rate in the middle and later stages of each curriculum, according to the following formula:

$$\eta_t = \eta_{t-1} \times \alpha, \quad (7)$$

where η_t represents the learning rate of iteration t , and α denotes the attenuation coefficient.

4. Experiments and Results

4.1. Dataset

Two ship detection datasets, Deep Learning Vessel Dataset and Real-World Ship Detection Dataset, were utilized to train or evaluate the performance of the LFLD-CLbased NET. The Deep Learning Vessel Dataset is a publicly available dataset for an image recognition task (<https://www.kaggle.com/datasets/arpitjain007/game-of-deep-learning-ship-datasets>), consisting of 6252 images of five different types of vessel. This dataset contains a significant number of high-quality images, with clear contrasts and sharp focus, and an adequate number of images for each type, making it a suitable choice for training deep learning models. The Real-World Ship Detection Dataset, on the other hand, is a small dataset that we collected specifically for testing the robustness and generalization ability of models. The image quality in this dataset is more reflective of nearshore camera conditions, including various weather-related factors, providing a more accurate and representative assessment. The data in this dataset was captured using optical surveillance cameras. Compared to the Deep Learning Vessel Dataset, the images in this dataset are more realistic and can effectively reflect the robustness and generalization ability of models. Table 1 and Table 2 present the mathematical statistics for both datasets, and the corresponding figures (Figures 3 and 4) provide a visual representation of the data.

Table 1. Deep learning vessel dataset statistics.

Dataset	Category	Original—All	Balanced—All	Training	Test
Deep learning vessel dataset	Cargo	2120	1949	1881	68
	Carrier	916	1974	1908	66
	Cruise	832	1945	1879	66
	Military	1167	1954	1891	63
	Tankers	1217	2025	1962	63



Figure 3. Example images from Deep Learning Vessel Dataset.



Figure 4. Examples from the Real-World Ship Detection Dataset.

Table 2. Real-World Ship Detection Dataset statistics.

Dataset	Category	All	Training	Test
Real-World Ship Detection Dataset	Cargo	137	69	68
	Carrier	131	65	66
	Cruise	115	58	57
	Military	126	63	63
	Tankers	127	64	63

4.2. Evaluation Metrics

The LFLD-CLbased NET ship identification performance is evaluated quantitatively in terms of accuracy, precision, recall, and F1, which are as follows:

$$\left\{ \begin{array}{l} \text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{FP} + \text{FN} + \text{TN}), \\ \text{Recall} = \text{TP} / (\text{TP} + \text{FN}), \\ \text{Precision} = \text{TP} / (\text{TP} + \text{FP}), \\ \text{F1} = 2 \times (\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall}). \end{array} \right. \quad (8)$$

Positive samples correctly predicted as positive are represented by TP, while negative samples correctly predicted as negative are represented by TN. FP and FN denote negative and positive samples, respectively, that are incorrectly predicted.

4.3. Experiment Setup

The model architecture was based on ResNet34. The adopted optimizer was Adam. The whole model was optimized with the proposed loss function that integrates the probabilistic classification loss with the multiclass cross-entropy loss. The Adam optimizer was utilized with a batch size of 64, and the learning rate decay step size was set to 20 epochs based on empirical evaluation.

4.4. Results of Experiments

In order to thoroughly investigate the performance of the LFLD-CLbased NET, comparative experiments were conducted on multiple widely utilized deep learning models for image classification. These models include the renowned AlexNet, GoogLeNet, VGG, ResNet, Wide-ResNet, GoogleNet, DenseNet, and MobileNet, as well as their variant forms. All the baseline models and their variant forms were trained and evaluated by the Real-World Ship Detection Dataset. Model selection was grounded in their established reputation in the field of image classification, as well as their potential to achieve high accuracy in detection tasks.

Table 3 presents comparison experiment results. The experimental results indicate that LFLD-CLbased NET significantly improved upon the baseline models in the real maritime scenarios ship detection task, achieving a detection accuracy rate of 86.635%, coupled with a precision score of 87.090%, recall score of 86.226%, and F1 score of 86.431%. The LFLD-CL-based NET demonstrated a significant increase in accuracy when compared to the baseline models, which ranged from 37.855% to 72.555%.

To further validate the impact of the curriculum learning and LFLD mechanisms on the monitoring and detection performance of the model, we employed the control variable method and selected our ResNet34 skeleton model as the reference point. By establishing ResNet34-Without CL-320 and ResNet34-Without CL-9437, we aimed to conduct more comprehensive comparative experiments. The ResNet34 model trained on the Real-World Ship Detection Dataset is denoted as ResNet34-Without CL-320, while ResNet34-Without CL-9437 refers to the ResNet34 model trained on the Deep Learning Vessel Dataset. It is worth noting that neither of these models underwent curriculum training strategy nor LFLD learning rate decay strategy during training. In comparison, LFLD-CLbased NET-9437 was trained on the Deep Learning Vessel Dataset with the incorporation of curriculum training strategy and LFLD learning rate decay strategy. The experimental outcomes are

presented in the last three rows of Table 3. It is noteworthy that the results in Table 3 were evaluated using the test set of the Real-World Ship Detection Dataset.

Table 3. Evaluation metrics comparison with other approaches.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1 (%)
LeNet-5 [44]	37.855	37.126	37.845	36.479
AlexNet [45]	67.823	69.367	67.925	68.101
VGG-11 [46]	61.830	63.798	61.833	62.271
VGG-13 [46]	60.883	61.301	61.201	60.446
VGG-16 [46]	45.426	45.478	45.512	43.612
ResNet-18 [47]	66.246	69.169	66.268	65.674
ResNet-50 [47]	62.461	64.420	62.326	62.691
ResNet-101 [47]	51.735	54.297	52.064	52.646
ResNet-152 [47]	48.580	52.325	48.623	48.786
ResNext-50-32x4d [48]	58.675	62.455	58.319	58.635
ResNext-101-32x8d [48]	61.199	63.842	61.431	60.766
Wide-ResNet-50 [48]	58.675	62.455	58.319	58.635
Wide-ResNet-101 [48]	66.562	68.661	66.321	66.264
GoogleNet [49]	41.325	39.819	41.534	39.626
DenseNet-121 [50]	72.555	73.222	72.583	72.762
DenseNet-161 [50]	68.770	72.505	68.351	69.053
DenseNet-169 [50]	70.347	70.547	70.540	70.417
DenseNet-201 [50]	69.716	72.457	69.438	69.769
MobileNet-v2 [51]	57.729	59.267	57.855	58.012
MobileNet-v3-Large [52]	63.722	64.406	64.114	63.954
MobileNet-v3-Small [52]	52.681	53.665	52.856	52.913
ResNet34-Without CL-320	73.501	74.971	73.525	73.946
ResNet34-Without CL-9437	76.656	75.290	74.075	74.678
LFLD-CLbased NET-9437	86.635	87.090	86.226	86.431

The inferior performance of ResNet34-Without CL-320 was primarily due to the limited size of the training set in the Real-World Ship Detection Dataset, resulting in overfitting despite a more similar distribution between the training and test sets. ResNet34-Without CL-9437 compared to ResNet34-Without CL-320 revealed a 3% increase in detection accuracy for the latter, highlighting the potential benefits of expanding the dataset for enhanced model convergence. Notably, LFLD-CLbased NET-9437 exhibited a significant improvement in detection accuracy by 13% and 10% when compared to ResNet34-Without CL-320 and ResNet34-Without CL-9437, respectively, providing further evidence of the effectiveness of curriculum training and LFLD.

The results of the experiment provide additional evidence supporting the superior generalization ability and robustness of the LFLD-CLbased NET. Both LFLD-CLbased NET-9437 and ResNet34-Without CL-9437 were trained on the Deep Learning Vessel Dataset, but the testing was conducted using the Real-World Ship Detection Dataset, which exhibits a distribution bias compared to the training data. Thus, the testing results can better reflect the model's generalization ability and stability. The experimental outcomes revealed that the F1 score for ResNet34-Without CL was 74.678, while that for LFLD-CLbased NET was 86.431, indicating a significant improvement of 11.753%.

We also analyzed the balance relationship between model accuracy rate and recall rate, and the experimental results are shown in Figure 5. It can be seen that the AUC-PR value of LFLD-CLbased NET is significantly higher than that of other models, with better generalization ability and robustness. The AUC-PR value of 0.94 of LFLD-CLbased NET indicates that the model has a very good balance between accuracy rate and recall rate, and can effectively distinguish positive and negative examples, so it has high predictive power. In contrast, LeNet's AUC-PR value is 0.37, which makes it difficult to effectively distinguish positive and negative examples.

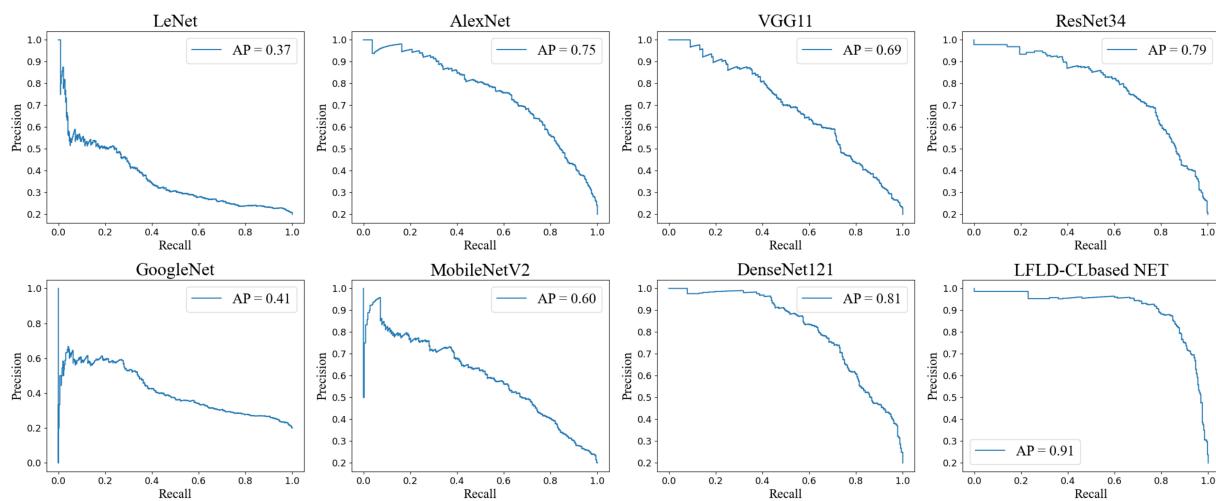


Figure 5. The ship detection PR (precision–recall) curve.

4.5. Comparison of the Training Process for LFLD-CLbased NET

We conducted an analysis of the training process for the ResNet34-Without CL-320, ResNet34-Without CL-9437, and LFLD-CLbased NET-9437, and the experimental results are shown in Figure 6.

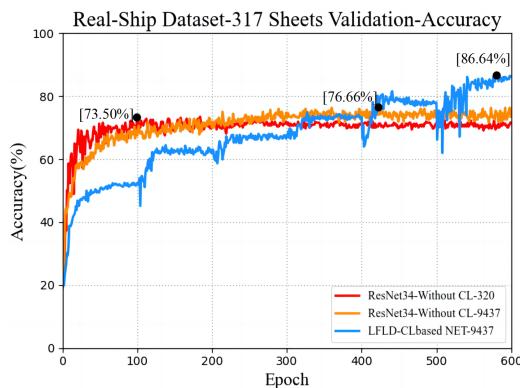


Figure 6. Comparative plot of model training evolution.

During the initial stages of the curriculum training, the testing accuracy of ResNet34-Without CL-320 exhibited a significant increase, surpassing that of ResNet34-Without CL-9437, and LFLD-CLbased NET-9437 at certain points. The improvement observed can be ascribed to the small real-world dataset used to train ResNet34-Without CL-320, which had a relatively uniform distribution between the training and test sets, enabling the model to converge rapidly in the early training phase. As the number of training epochs increased, we observed that the accuracy of ResNet34-Without CL-320 reached a peak value of 73.501 at epoch 99 before displaying a slow decline, consistent with our earlier analysis. This decline can be attributed to the dataset's limited number of training samples, which constrained the model's learning efficiency from making further improvements.

In contrast to ResNet34-Without CL-320, ResNet34-Without CL-9437 employed a more extensive and varied dataset for training, contributing to a gradual but consistent improvement in accuracy throughout a prolonged period as the number of training epochs increased. Notably, the model achieved its highest accuracy at epoch 422.

In contrast to the previous two models, LFLD-CLbased NET-9437 exhibited a step-like increase in accuracy during training, following the approach of curriculum learning. The curriculum learning mechanism involved training the model on a dataset that gradually increased in difficulty, with the difficulty level advancing by one step every 100 epochs. As a result, the accuracy of the model exhibited a cyclic increase with a period of 100 epochs.

Simultaneously, our findings indicate that during the early stages of each 10-epoch period (i.e., the initial phase of a new curriculum), the model's accuracy occasionally exhibited a rapid decline, attributable to the LFLD mechanism and curriculum learning mechanism. Those two mechanisms prompted the model to undergo a phase of rapid parameter adjustment, resulting from the addition of numerous challenging courses and learning rate adjustments, leading to a momentary drop in prediction accuracy. However, as the model progressed to learn and deepen its understanding of the new courses, its accuracy steadily increased, surpassing the previous highest point. These experimental outcomes provide further evidence of the effectiveness of curriculum learning mechanism and LFLD mechanism, which can enhance the model fitting process, learning efficiency, and robustness of the model.

4.6. Error Investigation

We carried out a comprehensive examination of errors in the baseline deep learning model and LFLD-CL-based NET using precision–recall curves and confusion matrices. The experimental results, shown in Figures 7 and 8, reveal that the overall misclassification rate of our model is still significantly lower than that of the baseline models.

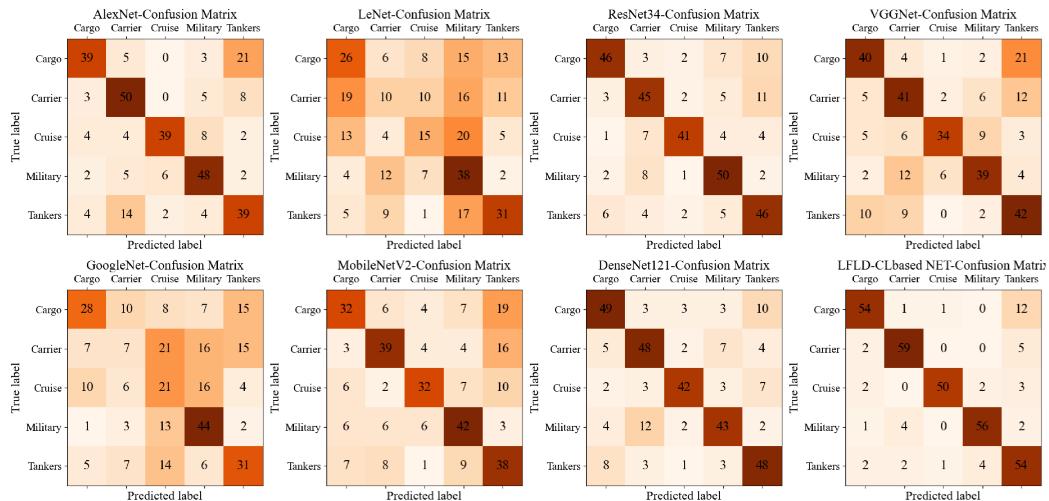


Figure 7. Confusion matrices of each model.

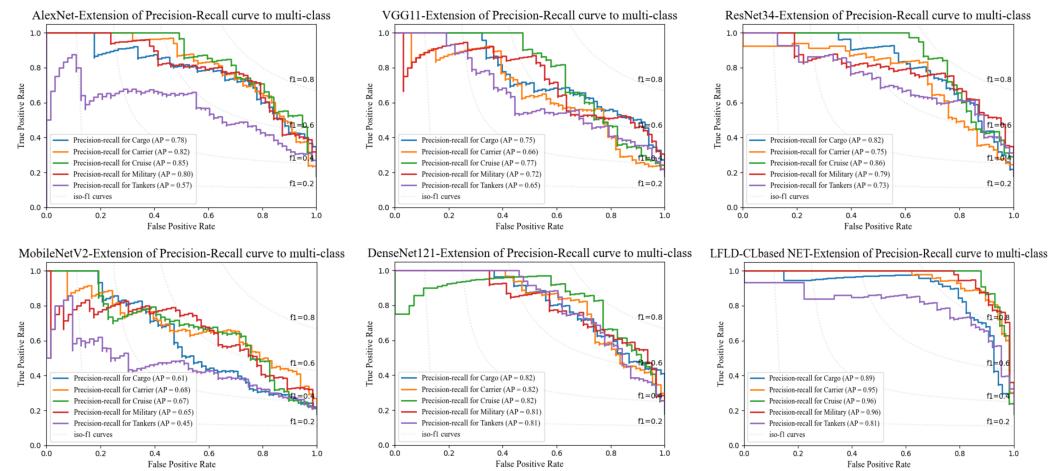


Figure 8. Precision–recall curves of each model.

The outcomes of the ship type prediction error analysis for different models are presented in Figures 7 and 8. Overall, LFLD-CLbased NET outperformed other deep learning baseline models. However, we identified a probability of LFLD-CLbased NET misclassifying

ing cargo as tankers. Nonetheless, this tendency was also observed among other models, with most other models exhibiting a misclassification rate approximately two times higher than our model. Furthermore, most deep-learning-based models demonstrated a tendency to misclassify tankers as carriers, whereas LFLD-CLbased NET showed significant improvement in this regard. Additionally, we were pleasantly surprised to find that our model exhibited high accuracy in predicting carriers, with significantly lower misclassification rates compared to other models.

4.7. Exploratory Analysis of the Impact of Noise Types

An extensive exploration was carried out to investigate how different noise types impact ship monitoring performance in real-world scenarios. We tested the effects of four of the most common noise types, namely Gaussian noise, pepper noise, s&p noise, and salt noise, on the model. Figure 9 illustrates sample images of these four noise types.

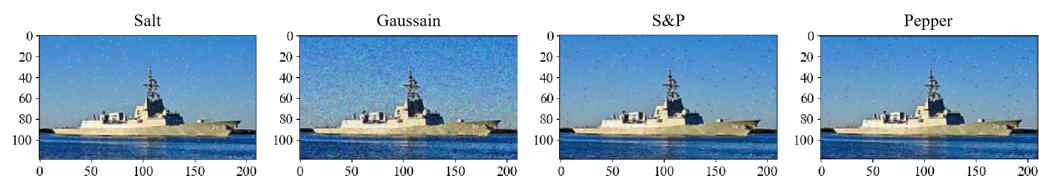


Figure 9. Illustration of noise types.

The ResNet-34 served as the primary test model in this study. To enable comparison, we utilized the curriculum learning mechanism to train the model with course learning taking place every 100 epochs. The course design method remained consistent with previous experiments, except for the substitution of the third and fourth course data with corresponding noise type data. It is worth noting that our model did not adopt the LFLD updating method in this experiment.

In the experiment, the amount values of noise data were set to 0.01 and 0.03 for the third and fourth course, respectively. The outcomes are illustrated in Figure 10. The results demonstrate that noise type salt is the most suitable for real-world scenarios, yielding a model accuracy of up to 86.64% when utilizing salt noise for course learning. This represents an approximate 3% improvement compared to the testing results obtained using Gaussian noise. The second-best noise type is pepper noise, with the model accuracy at 85.22%, slightly lower than that achieved with salt noise.

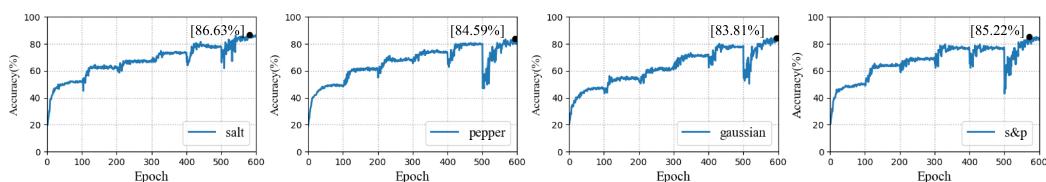


Figure 10. Accuracy curves of training results for different types of noise.

4.8. Analysis of the Impact of Noise Intensity

We conducted a further analysis of the impact of various noise intensities on the model. In the context of data augmentation techniques, noise data commonly refers to the incorporation of a certain degree of random perturbation into the original data. The amount value is a frequently used parameter for noise data, which aims to regulate the intensity and quantity of noise. Specifically, the amount value typically denotes the magnitude of the noise amplitude, which can be leveraged to control the proportion and intensity of noise in the data. For instance, higher amount values produce more pronounced noise perturbations, while lower amount values result in weaker noise perturbations. In the data augmentation process, the noise intensity can be precisely controlled by adjusting the amount value to enhance the effectiveness and stability of data augmentation. Figure 11 presents the noise data generated by different amount values.

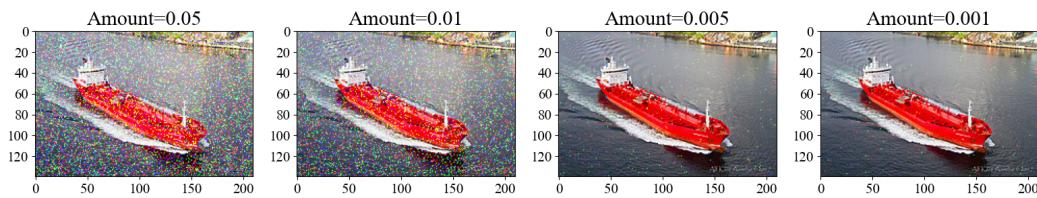


Figure 11. Illustration of noise intensity adjustment.

In this study, we persisted in using ResNet-34 as the reference model, employed salt noise as the primary noise source, adopted the curriculum learning as the fundamental model training approach, and designated the duration of each basic course learning to 50 epochs. We investigated various combinations of noise intensities, and the experimental outcomes are depicted in Figure 12. Notably, the icons in the figure represent different intensities of noise, where the icon for test 1 signifies that the amount value is set to 0.001 for the third course learning stage and 0.005 for the fourth course learning stage in test 1.

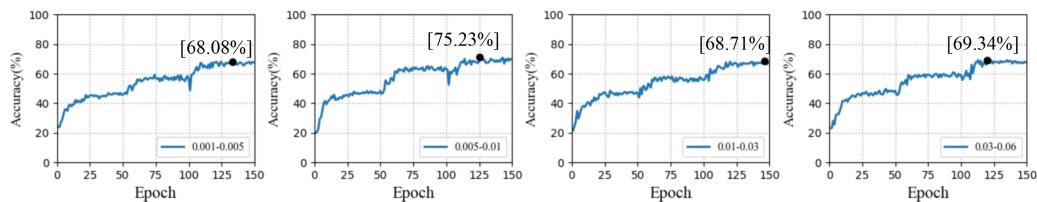


Figure 12. Accuracy curves of training results for different levels of noise intensity.

From Figure 12, it is evident that varying noise intensities also exert a substantial influence on the model's performance. The combination of amount values 0.005 and 0.01 can enable the model to better learn the distribution pattern of the data, achieving an accuracy of 71.23%, which is the highest among all noise combinations. The experimental results also reveal that adding a larger amount of noise information to the model does not necessarily lead to better performance. For instance, test 4, which adopts noise amount values 0.03 and 0.06, can enhance the model's adaptability to noise and interference to some extent, thereby improving its robustness. However, this combination does not conform to the noise distribution in real-world scenarios. Additionally, we also observed that weak noise has limited assistance to the model, as demonstrated by the results of test 1.

4.9. Analysis of the Effectiveness of Learning Rate Decay Strategy

To further investigate the effectiveness of learning rate decay strategy, we conducted an ablation study to confirm the role of learning rate decay strategy in the ship detection task. We devised two models, LD-CLbased NET and CLbased NET, by leveraging our original model as the backbone. LD-CLbased NET employed the conventional constant decay strategy for learning rate decay, while CLbased NET did not incorporate any learning rate decay strategy. The experimental outcomes are demonstrated in Figure 13.

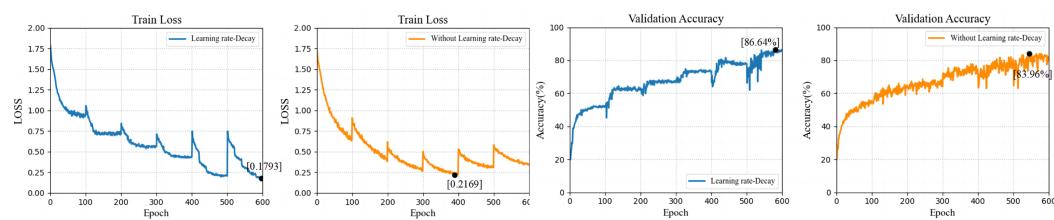


Figure 13. Comparison of training process with and without learning rate decay strategy.

The experimental results indicate that the model's accuracy is higher when the learning rate decay strategy is employed, achieving a performance of 86.3%, which is 3% higher than the model without learning rate decay strategy. This implies that the learning rate

decay strategy can assist the model in surpassing local optimal solutions and enhancing its detection performance in ship detection tasks. Moreover, it can be observed from the training loss plot that the implementation of the learning rate decay strategy promotes quick attainment of the optimal solution in the early stages of training, while also preventing overfitting in the later stages, resulting in improved training efficiency. This outcome is also evident from the valid acc plot, where it can be seen that the model's valid acc exhibits a rapid increase during the first half of each course learning stage, followed by a stable maintenance in the latter half, when the learning rate decay strategy is employed.

4.10. Exploratory Analysis of Learning Rate Initialization for LFLD Mechanism

In Section 4.6, we presented evidence supporting the efficacy of the learning rate decay strategy in ship detection task. In this section, we conduct an exploratory analysis of the performance of the LFLD mechanism. Specifically, we investigate the influence of various combinations of initial learning rates on model performance across six course learning stages for the LFLD mechanism. Notably, the learning rate decay values applied in this section are set uniformly to 1×10^{-2} , with each course unit comprising 50 epochs. The initial learning rate settings for every course stage are tabulated in Table 4, and the experimental outcomes are illustrated in Figure 14.

Table 4. Different combinations of learning rate initialization.

LR1	LR2	LR3	LR4
Lr = 0.000005	Lr = 0.000001	Lr = 0.000005	Lr = 0.00001
Lr = 0.00001	Lr = 0.000005	Lr = 0.000001	Lr = 0.00005
Lr = 0.000005	Lr = 0.0000025	Lr = 0.000001	Lr = 0.00005
Lr = 0.000005	Lr = 0.0000025	Lr = 0.000001	Lr = 0.00005
Lr = 0.00005	Lr = 0.00001	Lr = 0.000005	Lr = 0.0001
Lr = 0.00001	Lr = 0.0000075	Lr = 0.000001	Lr = 0.00005

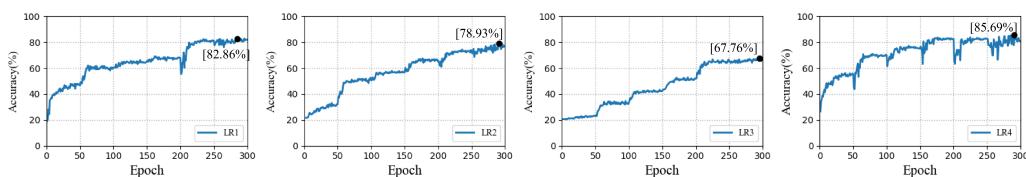


Figure 14. Accuracy curves for different combinations of initial learning rates.

The results reveal that different initial learning rate values have a certain impact on the model's learning process. It can be observed that when the initial learning rate is set to a large value, the model's performance exhibits significant oscillations during the early stages of course learning, and even experiences short-term performance degradation due to the large adjustment of weights caused by the learning rate. However, this decline is transient, and with the model's learning progress, a larger learning rate can help the model rapidly fit the data information. The most representative example of this phenomenon is the LR4 learning rate scheme, where the learning rate value is relatively large compared to other schemes, resulting in oscillations in the model's performance during the learning process. Meanwhile, the model's performance increases rapidly during the oscillations.

After continuous adjustments, the optimal learning rate value and the corresponding learning process for LFLD-CLbased NET are shown in Figure 15. It can be observed that the learning rate exhibits a leapfrog-like increase across different course learning stages, while within each course learning stage, the learning rate decays continuously with a fixed step size.

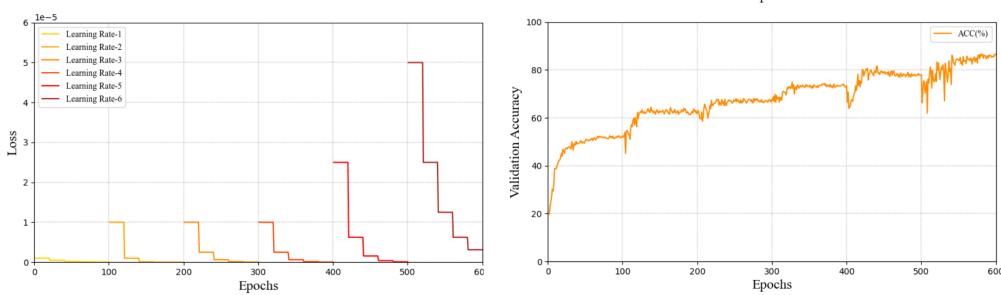


Figure 15. Learning rate variation curve and accuracy curve for LFLD-CLbased NET.

5. Conclusions

Existing deep-learning-based ship detection approaches typically encounter the training samples in a random or unstructured manner, resulting in a less stable, overfit model with poor performance, and difficulty in optimization. In this article, we introduce a novel model, LFLD-CLbased NET, which incorporates the concept of curriculum learning. Specifically, the model leverages a difficulty generator and scheduler in order to present training data to the model in a structured and gradually increasing order of complexity, tailored to real oceanic scenarios. Experimental findings demonstrate that curriculum learning strategy leads to a significant improvement in the model's training performance and detection accuracy, increasing its robustness and generalization capability to effectively handle noise and variations in the data. Additionally, to further enhance the model's training efficiency, we introduce the LFLD learning rate decay strategy, which dynamically adjusts the model's backpropagation update rate, accelerates the convergence speed, and improves its generalization ability and training efficiency by enabling the model to overcome local optima. Notably, both curriculum learning and LFLD are extremely valuable for deep-learning-based ship detection models. In future research, we will integrate other information generation and scheduling techniques for obtaining a more comprehensive training dataset to further enhance the robustness and generalization capability of the model. Additionally semisupervised learning or PU-learning techniques will be incorporated into our model to overcome no-ground-truth ship images problems.

Author Contributions: Conceptualization, software, validation, supervision, writing—original draft preparation, and writing—review and editing: J.L. Methodology, conceptualization, and writing—review and editing: J.S. Data curation and validation: X.L. Methodology and resources: Y.Y. Validation and resources: X.J. Funding acquisition, resources, methodology, and supervision: R.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Ocean Young Talent Innovation Programme of Zhanjiang City (Grant No. 2022E05002), the Young Innovative Talents Grants Programme of Guangdong Province (Grant No. 2022KQNCX024), the National Natural Science Foundation of China (Grant No. 52171346), the Natural Science Foundation of Guangdong Province (Grant No. 2021A1515012618), and the special projects of key fields (Artificial Intelligence) of Universities in Guangdong Province (Grant No. 2019KZDZX1035), the program for scientific research start-up funds of Guangdong Ocean University, and the College Student Innovation Team of Guangdong Ocean University (Grant No. CXTD2021013).

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Liu, R.W.; Yuan, W.; Chen, X.; Lu, Y. An enhanced CNN-enabled learning method for promoting ship detection in maritime surveillance system. *Ocean Eng.* **2021**, *235*, 109435. [[CrossRef](#)]
2. Er, M.J.; Zhang, Y.; Chen, J.; Gao, W. Ship detection with deep learning: A survey. *Artif. Intell. Rev.* **2023**, *1–41*. [[CrossRef](#)]
3. Li, J.; Yang, Y.; Li, X.; Sun, J.; Li, R. Knowledge-Transfer-Based Bidirectional Vessel Monitoring System for Remote and Nearshore Images. *J. Mar. Sci. Eng.* **2023**, *11*, 1068. [[CrossRef](#)]

4. Thombre, S.; Zhao, Z.; Ramm-Schmidt, H.; García, J.M.V.; Malkamäki, T.; Nikolskiy, S.; Hammarberg, T.; Nuortie, H.; Bhuiyan, M.Z.H.; Särkkä, S.; et al. Sensors and AI techniques for situational awareness in autonomous ships: A review. *IEEE Trans. Intell. Transp. Syst.* **2020**, *23*, 64–83. [[CrossRef](#)]
5. Zhu, X.X.; Tuia, D.; Mou, L.; Xia, G.S.; Zhang, L.; Xu, F.; Fraundorfer, F. Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE Geosci. Remote Sens. Mag.* **2017**, *5*, 8–36. [[CrossRef](#)]
6. Li, Y.; Zhang, S.; Wang, W.Q. A lightweight faster R-CNN for ship detection in SAR images. *IEEE Geosci. Remote Sens. Lett.* **2020**, *19*, 4006105. [[CrossRef](#)]
7. Zhang, T.; Zhang, X.; Ke, X.; Liu, C.; Xu, X.; Zhan, X.; Wang, C.; Ahmad, I.; Zhou, Y.; Pan, D.; et al. HOG-ShipCLSNet: A novel deep learning network with hog feature fusion for SAR ship classification. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 5210322. [[CrossRef](#)]
8. Jiang, X.; Xie, H.; Chen, J.; Zhang, J.; Wang, G.; Xie, K. Arbitrary-oriented ship detection method based on long-edge decomposition rotated bounding box encoding in SAR images. *Remote Sens.* **2023**, *15*, 673. [[CrossRef](#)]
9. Zhou, Y.; Fu, K.; Han, B.; Yang, J.; Pan, Z.; Hu, Y.; Yin, D. D-MFPN: A Doppler Feature Matrix Fused with a Multilayer Feature Pyramid Network for SAR Ship Detection. *Remote Sens.* **2023**, *15*, 626. [[CrossRef](#)]
10. Del Prete, R.; Graziano, M.D.; Renga, A. Unified Framework for Ship Detection in Multi-Frequency SAR Images: A Demonstration with COSMO-SkyMed, Sentinel-1, and SAOCOM Data. *Remote Sens.* **2023**, *15*, 1582. [[CrossRef](#)]
11. Zha, M.; Qian, W.; Yang, W.; Xu, Y. Multifeature transformation and fusion-based ship detection with small targets and complex backgrounds. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 4511405. [[CrossRef](#)]
12. Qin, C.; Wang, X.; Li, G.; He, Y. An Improved Attention-Guided Network for Arbitrary-Oriented Ship Detection in Optical Remote Sensing Images. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 6514805. [[CrossRef](#)]
13. Zhang, J.; Xing, M.; Sun, G.C.; Li, N. Oriented Gaussian function-based box boundary-aware vectors for oriented ship detection in multiresolution SAR imagery. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 5211015. [[CrossRef](#)]
14. Zhang, T.; Zhang, Z.; Yang, H.; Guo, W.; Yang, Z. Ship Detection of Polarimetric SAR Images Using a Nonlocal Spatial Information-Guided Method. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 4513805. [[CrossRef](#)]
15. Wu, F.; Zhou, Z.; Wang, B.; Ma, J. Inshore ship detection based on convolutional neural network in optical satellite images. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2018**, *11*, 4005–4015. [[CrossRef](#)]
16. Nie, G.H.; Zhang, P.; Niu, X.; Dou, Y.; Xia, F. Ship detection using transfer learned single shot multi box detector. In Proceedings of the ITM Web of Conferences, EDP Sciences, Messina, Italy, 24–27 October 2017; Volume 12, p. 01006.
17. Ma, X.; Li, W.; Shi, Z. Attention-based convolutional networks for ship detection in high-resolution remote sensing images. In *Proceedings of the Pattern Recognition and Computer Vision: First Chinese Conference, PRCV 2018, Guangzhou, China, 23–26 November 2018; Proceedings, Part IV 1*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 373–383.
18. Cui, Z.; Li, Q.; Cao, Z.; Liu, N. Dense attention pyramid networks for multi-scale ship detection in SAR images. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 8983–8997. [[CrossRef](#)]
19. Chen, J.; Chen, K.; Chen, H.; Zou, Z.; Shi, Z. A degraded reconstruction enhancement-based method for tiny ship detection in remote sensing images with a new large-scale dataset. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5625014. [[CrossRef](#)]
20. Ren, Z.; Tang, Y.; He, Z.; Tian, L.; Yang, Y.; Zhang, W. Ship detection in high-resolution optical remote sensing images aided by saliency information. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5623616. [[CrossRef](#)]
21. Zhang, L.; Liu, Y.; Zhao, W.; Wang, X.; Li, G.; He, Y. Frequency-Adaptive Learning for SAR Ship Detection in Clutter Scenes. *IEEE Trans. Geosci. Remote Sensing* **2023**, *early access*. [[CrossRef](#)]
22. Zhang, W.; Zhang, R.; Wang, G.; Li, W.; Liu, X.; Yang, Y.; Hu, D. Physics Guided Remote Sensing Image Synthesis Network for Ship Detection. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 4700814. [[CrossRef](#)]
23. Zhang, Y.; Li, Q.Z.; Zang, F.N. Ship detection for visual maritime surveillance from non-stationary platforms. *Ocean Eng.* **2017**, *141*, 53–63. [[CrossRef](#)]
24. Nie, X.; Yang, M.; Liu, R.W. Deep neural network-based robust ship detection under different weather conditions. In Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand, 27–30 October 2019; pp. 47–52.
25. Soviany, P.; Ionescu, R.T.; Rota, P.; Sebe, N. Curriculum learning: A survey. *Int. J. Comput. Vis.* **2022**, *130*, 1526–1565. [[CrossRef](#)]
26. Wang, X.; Chen, Y.; Zhu, W. A survey on curriculum learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 4555–4576. [[CrossRef](#)]
27. Bengio, Y.; Louradour, J.; Collobert, R.; Weston, J. Curriculum learning. In Proceedings of the 26th Annual International Conference on Machine Learning, Montreal, QC, Canada, 14–18 June 2009; pp. 41–48.
28. Platanios, E.A.; Stretcu, O.; Neubig, G.; Poczos, B.; Mitchell, T.M. Competence-based curriculum learning for neural machine translation. *arXiv* **2019**, arXiv:1903.09848.
29. Tay, Y.; Wang, S.; Tuan, L.A.; Fu, J.; Phan, M.C.; Yuan, X.; Rao, J.; Hui, S.C.; Zhang, A. Simple and effective curriculum pointer-generator networks for reading comprehension over long narratives. *arXiv* **2019**, arXiv:1905.10847.
30. El-Bouri, R.; Eyre, D.; Watkinson, P.; Zhu, T.; Clifton, D. Student-teacher curriculum learning via reinforcement learning: Predicting hospital inpatient admission location. In Proceedings of the International Conference on Machine Learning, PMLR, Virtual, 13–18 July 2020; pp. 2848–2857.
31. Florensa, C.; Held, D.; Wulfmeier, M.; Zhang, M.; Abbeel, P. Reverse curriculum generation for reinforcement learning. In Proceedings of the Conference on Robot Learning, PMLR, Mountain View, CA, USA, 13–15 November 2017; pp. 482–495.

32. Narvekar, S.; Sinapov, J.; Stone, P. Autonomous Task Sequencing for Customized Curriculum Design in Reinforcement Learning. In Proceedings of the IJCAI, Melbourne, Australia, 19 August 2017; pp. 2536–2542.
33. Ren, Z.; Dong, D.; Li, H.; Chen, C. Self-paced prioritized curriculum learning with coverage penalty in deep reinforcement learning. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *29*, 2216–2226. [CrossRef]
34. Gong, C.; Yang, J.; Tao, D. Multi-modal curriculum learning over graphs. *ACM Trans. Intell. Syst. Technol. TIST* **2019**, *10*, 1–25. [CrossRef]
35. Qu, M.; Tang, J.; Han, J. Curriculum learning for heterogeneous star network embedding via deep reinforcement learning. In Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, Los Angeles, CA, USA, 9 February 2018; pp. 468–476.
36. Guo, Y.; Chen, Y.; Zheng, Y.; Zhao, P.; Chen, J.; Huang, J.; Tan, M. Breaking the curse of space explosion: Towards efficient nas with curriculum search. In Proceedings of the International Conference on Machine Learning, PMLR, Virtual, 13–18 July 2020; pp. 3822–3831.
37. Guo, S.; Huang, W.; Zhang, H.; Zhuang, C.; Dong, D.; Scott, M.R.; Huang, D. Curriculumnet: Weakly supervised learning from large-scale web images. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 135–150.
38. Jiang, L.; Meng, D.; Mitamura, T.; Hauptmann, A.G. Easy samples first: Self-paced reranking for zero-example multimedia search. In Proceedings of the 22nd ACM international conference on Multimedia, Orlando, FL, USA, 3–7 November 2014; pp. 547–556.
39. Hacohen, G.; Weinshall, D. On the power of curriculum learning in training deep networks. In Proceedings of the International Conference on Machine Learning, PMLR, Long Beach, CA, USA, 9–15 June 2019; pp. 2535–2544.
40. Mousavi, H.; Imani, M.; Ghassemian, H. Deep curriculum learning for polsar image classification. In Proceedings of the 2022 International Conference on Machine Vision and Image Processing (MVIP), Macau, China, 12–14 January 2022; pp. 1–5.
41. Wang, Y.; Gan, W.; Yang, J.; Wu, W.; Yan, J. Dynamic curriculum learning for imbalanced data classification. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 5017–5026.
42. Wang, J.; Wang, X.; Liu, W. Weakly-and semi-supervised faster r-cnn with curriculum learning. In Proceedings of the 2018 24th International Conference on Pattern Recognition (ICPR), Beijing, China, 20–24 August 2018; pp. 2416–2421.
43. Goyal, A.; Agarwal, D.; Subramanian, A.; Jawahar, C.; Sarvadevabhatla, R.K.; Saluja, R. Detecting, Tracking and Counting Motorcycle Rider Traffic Violations on Unconstrained Roads. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 4303–4312.
44. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
45. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
46. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
47. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
48. Xie, S.; Girshick, R.; Dollár, P.; Tu, Z.; He, K. Aggregated residual transformations for deep neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1492–1500.
49. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
50. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
51. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 4510–4520.
52. Howard, A.; Sandler, M.; Chu, G.; Chen, L.C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for mobilenetv3. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1314–1324.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.