SeaShips: A Large-Scale Precisely Annotated Dataset for Ship Detection

Zhenfeng Shao, Wenjing Wu, Zhongyuan Wang, Wan Du, and Chengyuan Li

***Abstract*—In this paper, we introduce a new large-scale dataset of ships, called SeaShips, which is designed for training and evaluating ship object detection algorithms. The dataset currently consists of 31 455 images and covers six common ship types (ore carrier, bulk cargo carrier, general cargo ship, container ship, fishing boat, and passenger ship). All of the images are from about 10 080 real-world video segments, which are acquired by the monitoring cameras in a deployed coastline video surveillance system. They are carefully selected to mostly cover all possible imaging variations, for example, different scales, hull parts, illumination, viewpoints, backgrounds, and occlusions. All images are annotated with ship-type labels and high-precision bounding boxes. Based on the SeaShips dataset, we present the performance of three detectors as a baseline to do the following: 1) elementarily summarize the difficulties of the dataset for ship detection; 2) show detection results for researchers using the dataset; and 3) make a comparison to identify the strengths and weaknesses of the baseline algorithms. In practice, the SeaShips dataset would hopefully advance research and applications on ship detection.**

***Index Terms*—Object detection, ship dataset, neural networks, ship detection.**

I. Introduction

THE detection of inshore and offshore ships is an essential task for a large variety of applications in both military and civilian fields. For example, in the civil field, ship detection plays a strong supervisory role in monitoring and managing marine traffics, transportation, fisheries dumping of pollutants and illegal smuggling. In the military field, through obtaining ship location, size, direction, speed and other information, one can determine whether there exists ship cross-border or other abnormal behaviors to ensure the safety of coast and sea. Traditional ship detection in video heavily relies on the staffs' manual manipulation and provided that maritime staffs need to monitor the screen, it is inefficient and costly. Besides, due to the complexity of sea environment, it is very challenging for maritime staffs to keep focusing on the screen for a long time.

Computer-aided ship detection methods greatly free up human resources and typically include two steps: extracting image features, and then using classifiers for classification and localization. Zhu [1] extracts ships based on local binary patterns, shape features and gray intensity features. Yang [2] uses sea surface feature and a linear classifier to detect ships. Shen [3] firstly extracts ship proposals and then uses a directional gradient histogram feature to distinguish them. These methods can produce stable results under calm sea conditions. However, when disturbances such as waves, clouds, rain, fog, and reflections happen, the extracted low-level features are not robust. Besides, manual selection of features is time-consuming and strongly dependent on the expertise and characteristics of the data itself.

Therefore, later studies began to focus on how to integrate more ship features into detection and how to detect ships more precisely and quickly. In recent years, a lot of breakthroughs have been made owning to the convolution neural networks (CNN) [4]–[8]. Through a series of convolution and pooling layers, convnets can extract more distinguishable features. However, since convent is a data-driven approach, its ship detection performance has to rely on large-scale high-quality training dataset. Although a variety of open datasets, e.g., ImageNet [9], PASCAL VOC [10] and COCO [11], are available for the identification and detection of multiple static targets, they are designed for general object detection, but not specific for ship detection. We will show in our experiments that the performance of these general datasets in ship detection is unsatisfactory. In addition, there are also many unique datasets for specific object detection scenarios, e.g., face detection datasets (including CAS-PEAL [12], LFW [13] and FDDB [14]), pedestrian detection datasets (including Caltech-USA [15], KITTI [16] and CityPersons [17]), street sign dataset FSNS [18], fish image dataset LFIW [19], and bird image dataset Caltech-UCSD Birds 200 [20]. The disclosure of these large datasets greatly accelerates the development of object detection areas. However, public datasets specific for sea ship detection remain unavailable.

Manuscript received January 30, 2018; revised May 26, 2018 and July 13, 2018; accepted August 9, 2018. Date of publication August 17, 2018; date of current version September 18, 2018. This work was supported in part by the National High-Resolution Earth Observation System Major Projects of China under Grant 02-Y30B19-9001-15/17, in part by the National Natural Science Foundation of China under Grants 61671332, 41771452, and 41771454, in part by the Guangzhou Science and Technology Project under Grant 201604020070, and in part by the Key Research and Development Program of Hubei Province of China under Grant 2016AAA018. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Xuan Song. *(Corresponding author: Wenjing Wu.)*

Z. Shao, W. Wu, and C. Li are with the State Key Laboratory for Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430079, China (e-mail: shaozhenfeng@whu.edu.cn; wuwenjing94@163.com; lichengyuan@whu.edu.cn).

Z. Wang is with the National Engineering Research Center for Multimedia Software, Wuhan University, Wuhan 430072, China (e-mail: wzy_hope@163.com).

W. Du is with the Computer Science and Engineering, University of California, Merced, CA 95340 USA (e-mail: wdu3@ucm.edu).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org. The file contains five demo videos on ship detection, through which one can see the effect of the SeaShips dataset applied to actual situations more intuitively. The material is 42.5 MB in size.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2018.2865686

TABLE I
COMPARISON WITH OTHER DATASETS

Dataset	Images[a]	Objects[b]	Types	Image size
VOC2007	363	791	1(boat)	random
CIFAR-10	6000	6000	1(ship)	32x32
Caltech-256	418	418	4[c]	random
SeaShips	31455	40077	6	1920x1080

For this reason, our paper presents a new large-scale ship dataset, named as SeaShips, which consists of 31455 1920 $\times$ 1080 images of 6 principal ship types, including ore carrier, bulk cargo carrier, general cargo ship, container ship, fishing boat and passenger ship. Every image in our SeaShips dataset is precisely annotated with ship labels and bounding boxes. We build the SeaShips dataset based on the images captured by an in-field video monitoring system deployed around the Hengqin Island, Zhuhai city, China. In this project, 156 cameras are deployed in 50 different locations around the northwest border of the Hengqin Island, covering a total of 53 km^2 of coastal areas. We select images for our dataset from 45 cameras which are deployed at 45 different locations around the seashore. For each camera, we acquire videos in January, April, August and October across Year 2017 and Year 2018, from 6:00 a.m. to 20:00 p.m. every day. To make our dataset more diverse to contain all possible situations, the selected images cover a set of features, including different ship types, hull parts, scales, viewpoints, illumination and different occlusion degrees in a variety of complex environments.

We compare our SeaShips dataset against the existing general multi-object detection datasets which have ship target images. Table I presents their differences.

The PASCAL VOC2007 dataset [10] has 20 object classes, among which ship class only has 791 objects and one label 'boat'. The CIFAR-10 dataset [21] has 6000 32 $\times$ 32 images for label 'ship', but each image is far too small and only contains one target. Obviously, it is not suitable for practical applications and occlusion studies. The Caltech-256 dataset [22] has 4 ship categories including canoe, kayak, ketch, and speed boat, with each image being roughly 300 $\times$ 200 pixels. However, as objects in the dataset are all placed in the middle, this tendency means Caltech-256 is probably not an ideal candidate for object localization but classification.

To evaluate the usage of our proposed SeaShips dataset, we conduct experiments with three baseline detectors on SeaShips. Based on the experiment results, we summarize the advantages, difficulties and weaknesses of each detector and find out some possible directions in future research. For example, we find that YOLO v2 can achieve a proper tradeoff between accuracy and speed in practical applications. We also perform cross-validation experiments on SeaShips and the PASCAL VOC2007 dataset. The results demonstrate that YOLO v2 model trained by our SeaShips enjoys a good generalization ability on other test sets.

Research achievements based on the proposed dataset can be applied to following practical applications: 1) improving the detection accuracy of maritime vessel identification system to achieve better ship positioning and finer category information. 2) assisting real-time ship tracking technology to monitor sailing status, especially useful in transportation and fishery. 3) spurring the development in automatic identification of unusual events such as beaching, pulling back, turning around and abnormal speed in the future to ensure maritime safety.

The remainder of this paper is organized as follows. The related work about ship dataset and object detection algorithms are described in Section II. The acquisition and annotation process of ship images is introduced in Section III. We describe the detailed dataset design and analyze statistics of the newly produced dataset in Section IV. Experimental results of three baseline detection algorithms on our dataset are presented in Section V. We conclude this paper in Section VI.

II. RELATED WORKS

This paper mainly discusses datasets and neural networks for ship detection, and so we summarize the related works for these two aspects.

A. Object Detection Datasets

In the past decades, many datasets have been created for multi-objects and specific-object detection. ImageNet [9], PASCAL VOC [10], and COCO [11] are famous datasets for the identification and detection of multiple static targets. Although these datasets also contain ship targets, the number is relatively small and the categories are not rich, usually with only one label ship or boat.

In specific object detection scenarios, CAS-PEAL [12] advances face recognition technologies by providing a large-scale face database of Mongolian. CityPersons [17] which has large portions of occluded people not only constructs a more challenging pedestrian detection dataset, but also proposes five modifications to Faster R-CNN to improve pedestrian detection performances on general Caltech dataset [15]. Caltech-UCSD Birds 200 dataset [20] divides birds into 200 finer categories and adds new part localization annotations, which makes multi-class categorization and part localization possible. The disclosure of these datasets together with others like LFW [13], FDDB [14], KITTI [16], FSNS [18], fish dataset LFIW [19], etc. can not only enables models trained on them to better generalize to other test sets, but also reveals more challenges such as occlusions, thus greatly accelerates object detection researches.

However, public datasets specific for sea ship detection remain unavailable. Therefore, the quest for large ship dataset has become more urgent.

B. Object Detection Methods

In the light of deep learning paradigm, recent research has been focused on three mainstream branches on boosting the performance of object detection networks.

The first branch relies on improving convolutional neural network itself, either adjusting the base architecture or increasing the network depth. Initial attempt to adjust network architecture was given by ZF network in 2014 [23]. Other representative efforts attribute to Google's Inception series [24]–[26]. Based on the idea that deeper networks should lead to higher object detection accuracy, some studies are committed to deepen the network

layers. The representative works in this branch include VGGNet [27] and ResNet [28]. In addition, Inception ResNet [29] and ResNetXt [30] combine advantages of these two branches at the same time, resulting in better detection results.

The second branch has been focused on the optimization of deep-learning-based object detection algorithms, including region-based and regression-based detection algorithms. Region-based algorithms begin with the R-CNN [31], and afterwards researchers have put forward a series of variants like SPP-net [32], Fast R-CNN [33], Faster R-CNN [34], R-FCN [35] and Mask R-CNN [36]. This kind of algorithms' computation amount is large although the detection precision is very high. End-to-end object detection algorithms typically cover YOLO [37], SSD [38] and YOLO v2 [39]. They accomplish the determination of the position and category directly by a single network. As a result, one can quickly detect multi-targets in an image, although at the same time sacrifices some position accuracy.

The above two branches both require high-quality training data as support. It is generally believed that the recent success of object detectors is a product of the availability of more large-scale training data. Therefore, the third branch can be attributed to making full use of the data itself, such as the data sets described in Section II-A.

Our work is related to many researches in designing datasets [9]–[22], we are committed to reach a line that some particular databases such as pedestrian and face have achieved. The difference is that we are aiming at a new domain—sea ships.

III. Data Acquisition

We build the SeaShips dataset from the images captured by the cameras of a real-deployed video monitoring system. In the Zhuhai Hengqin New Area roundabout electronic fence project, 156 cameras are deployed in 50 different locations around the northwest border of the Hengqin Island, covering a total of 53 km^2 of coastal areas. At each location, three cameras are normally installed, including one low-light high-definition dome camera and two high-definition bolt cameras. The other six cameras are panoramic cameras. These cameras provide high-quality surveillance videos, from which our proposed dataset images are extracted.

A. Video Camera System

There are four main types of images commonly used in the detection of sea ships: optical remote sensing images [40]–[42], radar images [43], [44], infrared images [45], [46] and visible video images [47]. Among them, optical remote sensing images easily suffer from weather conditions like waves and cloud, which makes it difficult to achieve real-time monitoring in long operation period. Radar images can cover a wide range and penetrate occlusions, but their imaging resolution are poor, so that the captured ship targets only take up a few pixels in the entire image. This will bring a lot of inconveniences to object detection and tracking. Moreover, high cost of radar systems makes it difficult to achieve uninterrupted work within 24 hours. Infrared images have obvious advantages mainly in the night or under the circumstances lack of light, but they fail to provide rich color information. Visible light video images which can realize the clear monitoring in the short distance sea area enjoy lots of advantages, such as high resolution, rich in color and texture information, low price, low power consumption, and all-weather real-time operation. All these features allow it the best data resource for ship inspection.

Fig. 1. Camera equipments used to extract ship images.

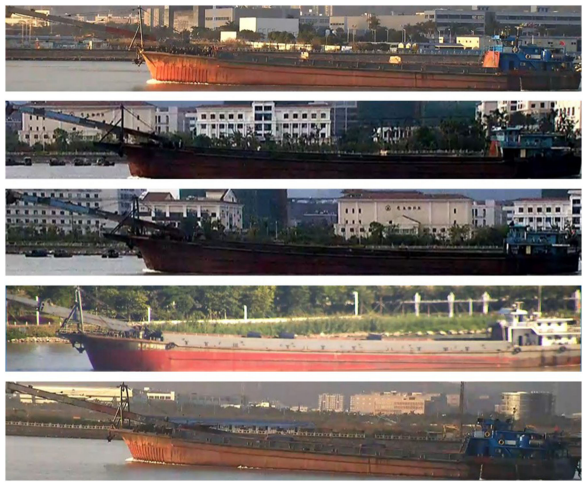


Fig. 2. Bulk cargo ships under different backgrounds.

Therefore, our proposed dataset refers to visible image, mainly obtained by frames within video sequences recorded by front-end cameras. Each video clip is divided into 1 minute long, including 1500 images. Fig. 1 shows the three video cameras used in this paper. Among them, HD bolt camera can only shoot at one direction, while it enjoys high video clarity. HD dome cameras can not only rotate at any angles to acquire videos from different viewpoints, but also switch focus to adapt to different scales. Panoramic camera is used to capture video data in a wide range. These three types of cameras can provide rich video data for extracting ship images.

B. Dataset Diversity

A good detection model should maintain sensitivity to interclass differences while giving stable test results. Due to the complexity of sea environments, all influencing factors need to be considered to ensure diversity of the dataset. Although adopting proper data augmentation methods would generate some formatting/distortions that may be present in the ship detection problem, some real-world data, by its very nature, can be hard to predict yet. So, real data is the first choice in sea ship detection. We will take the following steps to ensure diversity of our dataset:

1) Background Selection: In most detection tasks especially face recognition, the detection accuracy is rarely affected by

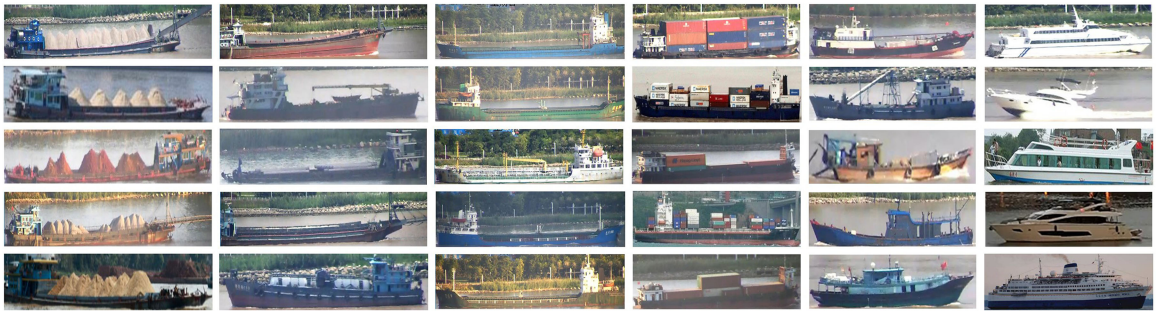


Fig. 3. Example images of six ship types.

background variations due to the fact that face area is filled in a regular rectangle, easily separated from background. But unlike human faces, the shape of ship is very irregular, which will lead to a lot of background information in the labeled bounding box (see Fig. 2 for an example). Background information will be identified as ship features and compromises final detection accuracy.

To avoid the impact of a single background, we collect ship data under 45 different backgrounds, five of which are shown in Fig. 2. This can be accessed by selecting cameras deployed in different locations.

2) Lighting Environment: Because the video cameras used to extract ship images are placed in natural environments, the illumination differences from different temporal periods are particularly significant. We collect images under different lighting conditions by selecting videos at different periods.

3) Visible Proportion: As most sea ships in cameras are moving, only part of ship hulls would appear on the screen when entering and leaving the camera's field of view. Actually, they still belong to the objects that need to be detected. We thus need to annotate both complete ship and incomplete ship hull parts at different visible proportion.

4) Occlusion: Due to the fact that SeaShips data was collected from broad sea area, we found that there are more than one ships sailing in some images, highly occluded by each other. It is obviously unreasonable to ignore occlusion. So, we collect as much occlusion data as possible in order that the subsequent training model can readily cope with occlusions.

C. Annotation

Like other large datasets, we use manual annotation methods to label our acquired images. As described below, we follow three steps to improve annotation quality.

1) On one hand, to cover all the factors described in Section III-B, the selected cameras should cover most of the sea areas and provide data at different viewpoints across the day. We thereby selected 168 videos from 45 cameras in different locations. Each video is one hour in duration and consists of 60 clips where every clip lasts 60 s, about 1500 frames. We took one image every 50 frames (approximately two seconds) and finally had 302400 original images in jpg format.

TABLE II
NUMBER OF IMAGES OF EACH SHIP CATEGORY

Ship Category	Images	Percentage
Ore carrier	5126	0.1630
Bulk cargo carrier	5067	0.1610
Container ship	3657	0.1163
General cargo ship	5342	0.1698
Fishing boat	5652	0.1797
Passenger ship	3171	0.1008
Mixed type[a]	3440	0.1094
Total	31455	1

2) On the other hand, since many images actually do not have ship objects, or there are few changes between images, the redundancy is thus unavoidable. So, we discarded those empty or repeated images to reduce the dataset to 31455 applicable images, which are named from 000001.jpg to 031455.jpg.
3) For each image in the dataset, we drew a bounding box tightly around the ship object using labeling tool. The generated xml file follows PASCAL VOC2007 format, but without 'difficult' tag. Then, the total 31455 annotated files named from 000001.xml to 031455.xml are divided into two parts, training set and testing set.

All the above steps were accomplished manually. Although this image selecting and annotating process costs a lot of time and manpower, it is worthwhile in the sense of the obtained high-quality dataset.

IV. Design and Statistics of SeaShips Dataset

By collecting video data from cameras described in Section III-A, 6 variations are investigated in order to construct the SeaShips dataset: hull part, scale, viewpoint, illumination, background, and occlusion.

A. Ship Classification System

Generally, objects in the image are labeled as either ship or background. Although different ship types share some basic elements such as deck and stern, different ship types vary greatly

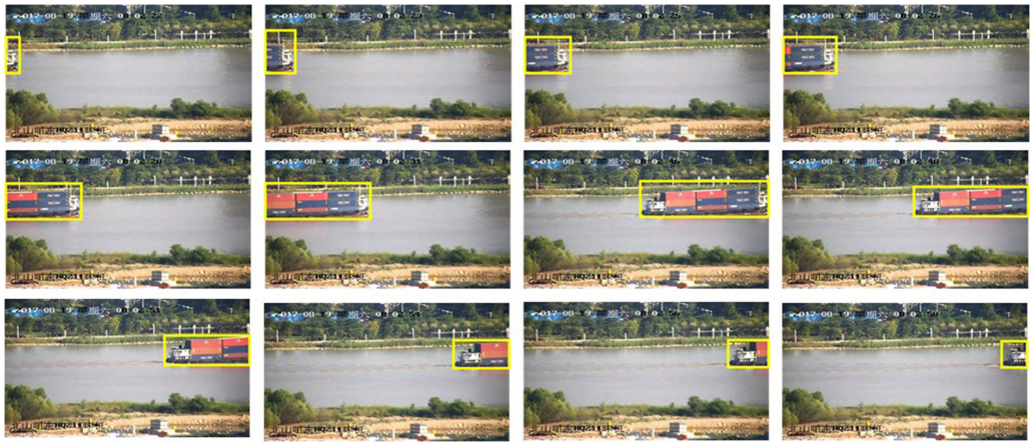


Fig. 4. Different annotated hull parts of container ship.

in shape and appearance. This intra-class differences make it difficult to perform ship detection with exact category.

In this paper, we need to provide further fine-grained labels for ships. According to the classification of civil ships in Introduction to Ship and Ocean Engineering [48] and the actual situation of sea environment in Hengqin Island, we group all ships in the sea into six categories, namely labels. They are ore carrier, bulk cargo carrier, general cargo ship, container ship, fishing boat, and passenger ship. Other types like oil tanker and barge carrier rarely appear in the sea areas, and they also belong to part of cargo ships. Therefore, these six types of ships roughly cover all ship categories that appear in the monitoring system. Fig. 3 shows some example images of each category. Images are all adjusted to a similar size for the convenience of display. Table II lists the number of images of each ship category in our proposed dataset. The numbers of container ship and passenger ship are relatively smaller than other types. We have 31455 images for now and there are about 60 images for each same ship to cover most variations. In this way, our proposed SeaShips dataset contains 500 different ships approximately.

B. Visible Proportion Variation

In addition to the situation where the entire ship hull is present in the cameras, we also annotated ships that only have a part of hulls appearing in the camera. Take container ship as an example, Fig. 4 shows 12 annotated ship bounding boxes from entering to leaving camera's range. Similarly, we applied this annotation process to all types of ship.

C. Scale Variation

By controlling HD dome cameras and panoramic cameras described in Section III-A, we collected sample bounding boxes of the same ship at different scales. Fig. 5 shows three scales of bulk cargo carrier when the camera is facing the sea. Fig. 6 shows 3 scales of bulk cargo carrier when the camera is looking side the sea. In this way, we collect images of different scales for all the 6 defined ship categories.

The size of different ships also varies greatly. The smallest ship is in 34 $\times$ 8 pixels and the largest one is in

(a) (b) (c)

Fig. 5. Three different scales when the camera is facing the sea.

1920 $\times$ 424 pixels, where the width ranges from 28 to 1920 pixels and the height ranges from 8 to 486 pixels, with the ratio (width/height) ranges from 0.39 to 17.98. Thus, the scales are enough to cover all sizes of ships.

D. Viewpoint Variation

In fact, adjacent cameras can capture pictures from different viewpoints. By turning the HD dome cameras, we get pictures of the same ship from different viewpoints. In this paper, we mainly use three viewpoints described as looking at left, looking at middle and looking at right. Fig. 7 shows example images of one fishing boat at 6 different sub-viewpoints recorded by one camera. Table III lists the number of images annotated at each three viewpoints. We can see that abnormal viewpoints (looking left and looking right) take up nearly one third of the entire dataset, which makes the dataset a good resource for viewpoint learning.

E. Illumination Variation

To acquire image data under different illumination conditions, we collected videos from 6:00 am to 20:00 pm in January, April, August and October. We think that these 4 months are enough to cover all possible illumination variations caused by either seasonal change or time change. Fig. 8 illustrates example images of different lighting conditions. All these conditions are considered in our proposed dataset.

F. Background Variation

The background diversity can be investigated according to the cameras deployed on different locations. Fig. 9 shows nine camera images with the position and orientation of the respective

Fig. 6. Three different scales when the camera is looking side the sea.

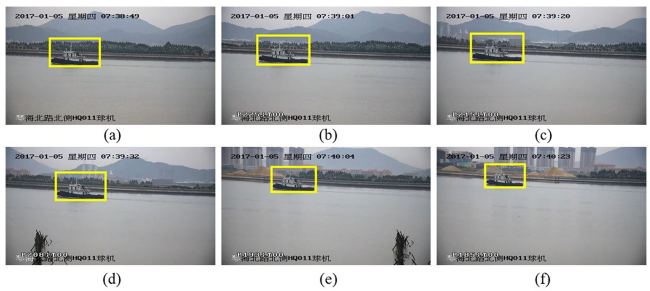


Fig. 7. Six sub-viewpoints of the same fishing boat below the left viewpoint.

TABLE III
NUMBER OF IMAGES AT THREE DIFFERENT VIEWPOINTS

Viewpoint	Images	Percentage
L^a	5347	0.17
M^b	21076	0.67
R^c	5032	0.16
Total	31455	1

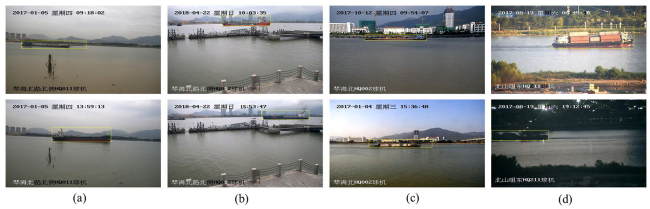


Fig. 8. Example images of different illuminations. Each column represents one month: January, April, August, and October.

cameras. Fig. 10 shows the number of images under each background, where B1 to B45 correspond to individual camera in counterclockwise order.

G. Occlusion Variation

In order to investigate which kinds of occlusions we have on SeaShips, we group all those images in the presence of occlusions together and roughly compare their occlusion degrees. We found that more than 10% images contain occlusion in the whole dataset (see Table II mixed type), with some of them even having a high occlusion ratio (>0.9). This makes SeaShips a more challenging dataset for occlusion handling. Fig. 11 illustrates the distribution of ships at different occlusion levels.

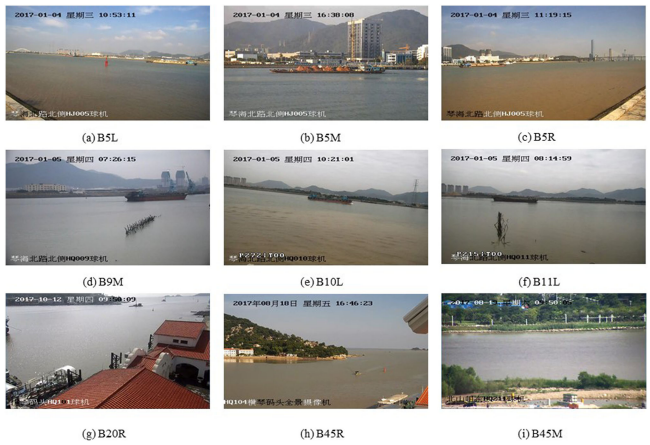


Fig. 9. Some backgrounds in the dataset. (a) B5L means background number five where the camera is looking at left. Similarly, (b), (c), (d), (e), (f), (g), (h), (i) show the backgrounds associated with the respective cameras.

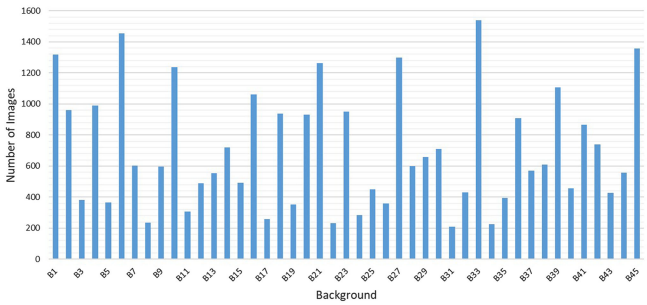


Fig. 10. Number of images under each background.

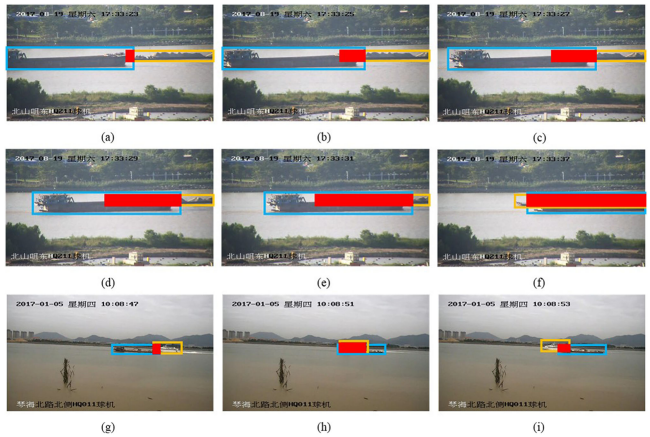


Fig. 11. Examples images of different occlusion degrees. The first and third rows correspond to reasonable occlusion situations. The second row is highly occluded by each other.

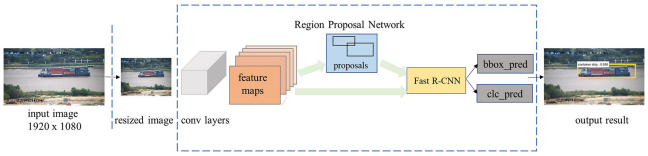


Fig. 12. The flowchart of Faster R-CNN applied in ship detection.

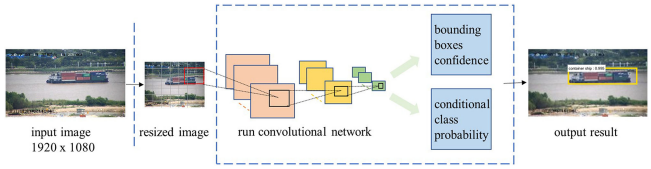


Fig. 13. The flowchart of YOLO v2 applied in ship detection.

V. Baseline Experiments on the SeaShips Dataset

To validate the functionalities of ship detection on the proposed SeaShips and provide reference evaluation results for researchers using the dataset, we retrain and evaluate three different baseline detectors on four Titan Xp. We also compare performance of Seaships with another general object dataset.

A. Baseline Ship Detection Algorithms

The three baseline detectors are Faster R-CNN [34], YOLO v2 [39], and SSD [38]. Faster R-CNN acts as the state-of-the-art detector. YOLO v2 uses end-to-end training network to achieve the purpose of real-time detection. SSD has made some progress in achieving good accuracy and speed at the same time using a pyramidal feature hierarchy structure.

1) Fast/Faster R-CNN: Fast R-CNN [33] solves the repetitive calculation problem caused by R-CNN and SPP-net. It uses RoI pooling, multi-task training, and mini-batch sampling to improve on speed and accuracy. However, using selective search method to extract proposals is still time-consuming.

Instead, Faster R-CNN creatively adopts a CNN to extract proposals: Region Proposal Network (RPN). RPN module can realize extraction of proposals through sharing characteristics with the convolution layer. The following Fast R-CNN module detects targets based on proposals extracted by RPN network. This makes the object detection network much faster.

Original RPN version usually generates 9 anchors in three different scales (8,16,32) and three different ratios (0.5,1,2) in one location. However, ships in our dataset are all threadlike, the ratio of width and height is rarely smaller than 1. Therefore, in our training process, we do not use the ratio of 2 to generate anchors. In this way, we decrease anchors in one location to 6 and thus reduce the time required for training.

Fig. 12 shows the flowchart of Faster R-CNN applied in ship detection. Firstly, the original images are resized to an appropriate size. Then, through a series of convolution and pooling layers, we get feature maps of the image. Thirdly, feature maps are input into RPN to generate proposals. Together with feature maps, it performs a Fast R-CNN process to decide whether a box belongs to a defined class. Then it uses a regression step to further adjust its position.

2) YOLO/YOLO v2: YOLO and YOLO v2 algorithms let proposal generation, feature extraction, object classification and localization be unified in one single neural network. They thus turn ship detection into a regression problem to achieve end-to-end detection. YOLO detection algorithm is much faster than region-based methods. However, it has relatively low recall and more localization errors.

YOLO v2 uses a few tricks to improve localization while maintaining classification accuracy. Detailed novel ideas to improve YOLO's performance can be found in [39]. As YOLO is somewhat outdated, we adopt YOLO v2 as our second baseline algorithm and its flowchart applied in ship detection is shown in Fig. 13. Firstly, the original images are resized to an appropriate size. Then, the resized image is divided into several grids to perform a single network, with each grid predicting several (for example, 5) anchor boxes and confidence scores. For those boxes containing objects, the network calculates conditional class probabilities. At last, conditional class probabilities and box confidence predictions jointly give class-specific confidence scores for each box.

3) SSD: Similar to anchors in Faster R-CNN, SSD generates several prior boxes of different ratios and scales on each feature map. Learning the idea of converting detection to regression from YOLO, SSD completely eliminates proposal generation subsequent pixel or feature resampling stage and unifies all computation in one single network. This makes fine-tuning process much easier. Furthermore, SSD makes full use of pyramidal feature hierarchy structure which combines predictions from multiple feature maps in different resolutions to naturally handle objectives of various sizes.

However, basic parameters of prior box, including its size, shape and ratios, cannot be obtained directly by learning, but by manual setting. In addition, the prior box size and shape used by each feature map in the network happen to be different, which will lead debugging process to rely on experience heavily.

B. Evaluation Protocol

For a given test set, there are some common quantitative indicators to evaluate a ship detection model. In this paper, we follow the same evaluate protocol as used by PASCAL VOC [10] which is briefly described below.

1) Intersection Over Union: Intersection over union, also IOU, defines the overlap degree of two bounding boxes. It can be computed as:

$$\mathrm{IOU} = \frac{\left|B_{gt} \cap^{B_p}\right|}{\left|B_{gt} \cup^{B_p}\right|} \tag{1}$$

As shown in Fig. 14, $\mathrm{B_{gt}}$ is the area of annotated ground truth box and $\mathrm{B_p}$ is the area of predicted bounding box.

Selecting an overlap threshold $\mathrm{threshold}$, the detector will decide whether the box belongs to background or not according to (2), where class 0 means background, and class i refers to our

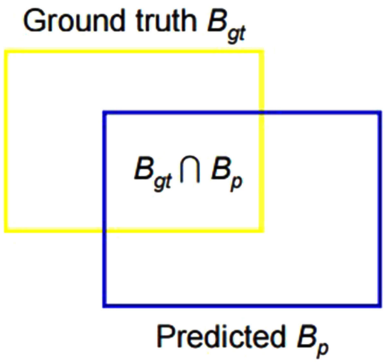


Fig. 14. Intersection Over Union.

category system.

$$class = \left\{ \begin{array}{l} 0, \quad (if\ IOU < threshold) \\ i, \quad (if\ IOU > threshold) \end{array} \right\} \tag{2}$$

2) Average Precision: Given an IOU threshold, there are two concepts, recall and precision. Recall is the proportion of the correctly detected boxes number to the ground truth boxes number. Precision means the ratio of the correctly detected boxes number to the total detected boxes number. For each type, we can draw a precision-recall curve according to recall and precision values. AP is the area surrounded by the curve (3).

$$\mathrm{AP} = \int_0^1 P\left(R\right) dR \tag{3}$$

3) Mean Average Precision: Each class i corresponds to an AP value AP_i, with mAP denoting their mean.

$$\mathrm{mAP} = \frac{\sum_{i=1}^{n} \mathrm{AP}_i}{n} \tag{4}$$

where n is the number of classes that need to be detected.

4) Frames Per Second: In addition to evaluating accuracy, we would also compare the running time of each detector. FPS symbolizes the number of frames/images that detector can detect in one second. We would adopt this indicator to measure speed.

C. Results and Analysis

All experiments are performed on pre-trained models based on ImageNet, and then fine-tuned for ship detection. For Faster R-CNN, we use ZF net [23] (5 convolutional layers and 3 fully-connected layers), VGG16 net [27] (13 convolutional layers and 3 fully-connected layers) and ResNet (ResNet18, ResNet50, ResNet101) [28] as the base network architectures. We also train Fast R-CNN algorithm with VGG16 for a comparison. For SSD, we use MobileNet [49] and VGG16 net as base network architecture. For YOLO v2, we adopt darknet19 and use some usual data augmentation methods such as hue, saturation, and exposure shifts. The training implementations, like number of iterations and learning rate strategies, keep the same in all experiments to ensure fair comparison.

1) Quantitative Results: We record the quantitative performances of each detector in Table IV. Fig. 15 shows AP curves of each ship type when we set IOU threshold to 0.5.

As it can be seen from Table IV, Fast R-CNN is worse than others by a wide margin in terms of mAP performance. mAP of Faster R-CNN series outperforms YOLO and SSD. On average, mAP of Faster R-CNN is 16.22% higher than YOLO and 12.58% higher than SSD.

For Faster R-CNN, detection accuracy and speed will increase when we use ResNet18, ResNet 50 and ResNet 101 instead of VGG16, especially ResNet 101. Considering their same experimental setup, the improvements can be attributed to better base networks.

In the situation where original images are resized to 300 $\times$ 300 pixels, SSD with VGG16 network promotes the accuracy by about 2% compared with MobileNet. Adjusting the images to larger sizes like 608 $\times$ 608 and 512 $\times$ 512 pixels would lead to improvements in mAP performance. This is especially obvious when adopting VGG16 network. Furthermore, SSD adopts a pyramidal feature hierarchy structure to combine feature maps of different layers, which to some extent alleviates the problem of small targets detection. This fact justifies why SSD outperforms YOLO.

When training YOLO v2 without any tricks other than its original paper, using multi-scale training (setting random $=$ 1) increases mAP by almost 2%. This operation would randomly change image size from 320 $\times$ 320 to 608 $\times$ 608 every 10 batches. While random $=$ 0 would fix the input size to 416 $\times$ 416, which is too small for 1920 $\times$1080 images in our proposed dataset.

In the defined six ship categories, the performance of fishing boat is worse than other types. The main reason is that fishing boat is generally small and only occupies 70 $\times$ 130 pixels within 1920 $\times$ 1080 image. After many forward convolutional layers, features of small targets become vague. Detectors often fail to handle such small targets which are dominant on fishing boat. Performance of passenger ship is also not very good, mainly because the number of passenger ships is relatively small. On the contrary, ore carrier and container ship can achieve much better results. It is because these two kinds of ships are mainly used for transporting goods like ore and containers. These goods have very distinctive features to be separated.

In terms of speed, although adopting end-to-end training method enables SSD to achieve considerable effects as Faster R-CNN, its detection speed improvement is not noticeable as far as real-time performance is concerned. In contrast, YOLO v2 is far better in detection speed than other methods, with FPS being 91. It can meet real-time detection requirements.

2) Qualitative Results: In Fig. 16, we show some visual results of detection algorithms, where each column corresponds to one particular situation: oblique viewpoint, very dark light, small ship target, slight occlusion, medium occlusion and high occlusion.

Evidently, most of the detection algorithms enjoy good results, with few cases of misdetection and missed detection. Impressively, the very perfect detection performance is achieved under unusual perspective (second column) and at night (third column).

Even though the proposed SeaShips dataset contains many samples of small ships, small object detection remains a

TABLE IV
DETECTION RESULTS ON THE SEASHIPS DATASET

Model	mAP	ore carrier	bulk cargo carrier	general cargo ship	container ship	fishing boat	passenger ship	FPS(Titan Xp)
Fast[a] (VGG16)	0.7103	0.7709	0.7133	0.7705	0.8681	0.6170	0.5220	0.5
Faster[b] (ZF)	0.8916	0.9050	0.9001	0.9077	0.9091	0.8568	0.8706	15
Faster (VGG16)	0.9012	0.8944	0.9034	0.9073	0.9087	0.8876	0.9057	6
Faster (ResNet18)	0.9063	0.9037	0.8978	0.9045	0.9091	0.8717	0.8893	21
Faster (ResNet50)	0.9165	0.9238	**0.9088**	0.9246	0.9291	0.8927	0.9093	17
Faster (ResNet101)	**0.9240**	**0.9368**	0.9022	**0.9387**	**0.9341**	**0.8996**	**0.9178**	7
SSD 300(MobileNet)	0.7766	0.6477	0.7669	0.8743	0.9077	0.7100	0.7532	16
SSD 608 (MobileNet)	0.7950	0.7827	0.7998	0.8515	0.8884	0.6730	0.7735	12
SSD 300 (VGG16)	0.7937	0.7503	0.7666	0.8766	0.9071	0.7179	0.7435	7
SSD 512 (VGG16)	0.8673	0.8399	0.8300	0.8708	0.9081	0.8585	0.8965	5
YOLO v2 random=0	0.7751	0.8301	0.7936	0.8060	0.8890	0.6270	0.7048	83
YOLO v2 random=1	0.7906	0.8316	0.8207	0.8321	0.8831	0.6474	0.7289	**91**

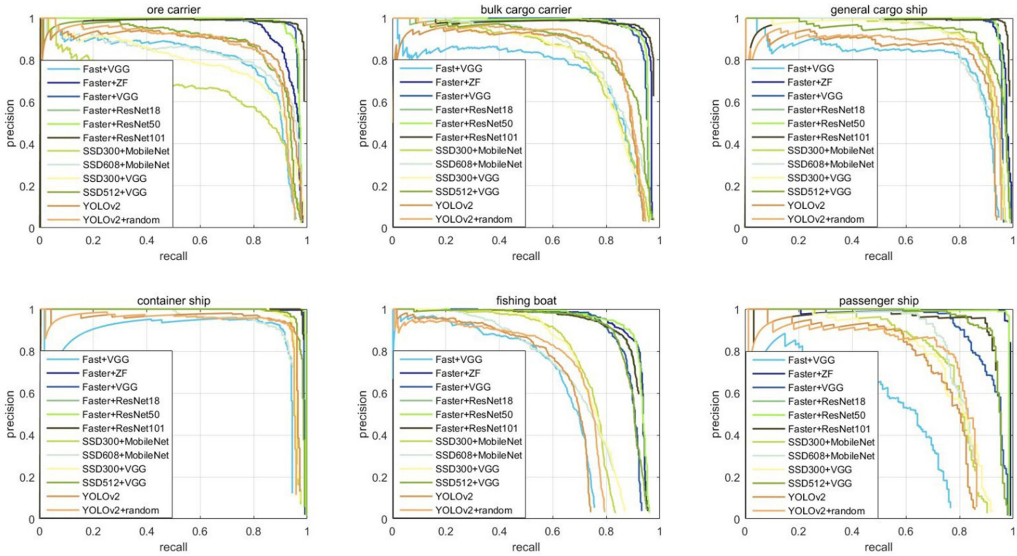


Fig. 15. Precision-recall curve for each detector on six ship types.

challenge. For example, the fishing boat (fourth column) only occupies 175 × 55 pixels, so that features of small ships will become insignificant even disappear.

Occlusion problem is another unavoidable challenge. Experiments have shown that all algorithms can correctly detect multiple ships at a suitable occlusion rate. However, when two ships are highly shaded from each other (the last column), it is difficult for the detectors to distinguish them.

Furthermore, we validate ship detection results under practical extreme weather conditions, shown in Fig. 17. It can be seen that fog, rain, wind and wave do not affect the detection a lot. However, reflection and cloud shadow can create confusion and thus disturb the detection of ships.

D. Comparison With a General Dataset

To illustrate generalization ability of the proposed SeaShips dataset, we compare its performance with a general dataset, PASCAL VOC2007 [10], which includes 363 images containing boat targets. We do not compare with the other two datasets in Table I, because CIFAR-10 only contains images of 32 × 32 pixels which is too small for real-world situations. And objects in Caltech-256 are all placed in the middle of the image, which makes it unsuitable for object localization.

Based on the experimental results of three baseline detectors in Section V-C, we find that YOLO v2 can achieve a proper tradeoff between accuracy and speed in practical application. Therefore, we use YOLO v2 as the base detection algorithm in this comparison experiment. To validate the universality of the datasets, we conduct three experiments: VOC2007 as training data and SeaShips as testing data, SeaShips as training data and VOC2007 as testing data, SeaShips as training data and SeaShips as testing data. Although the SeaShips dataset can be used to identify six types of ships, since the PASCAL VOC2007 dataset contains only one class of boat, we will use one category for experiments for fair competition (i.e., six fine ships are all defined as 'ship').

Table V shows the experimental results in terms of recall, precision and AP, on conditions that the IoU threshold is set to 0.5. It is obvious that the trained model of the original

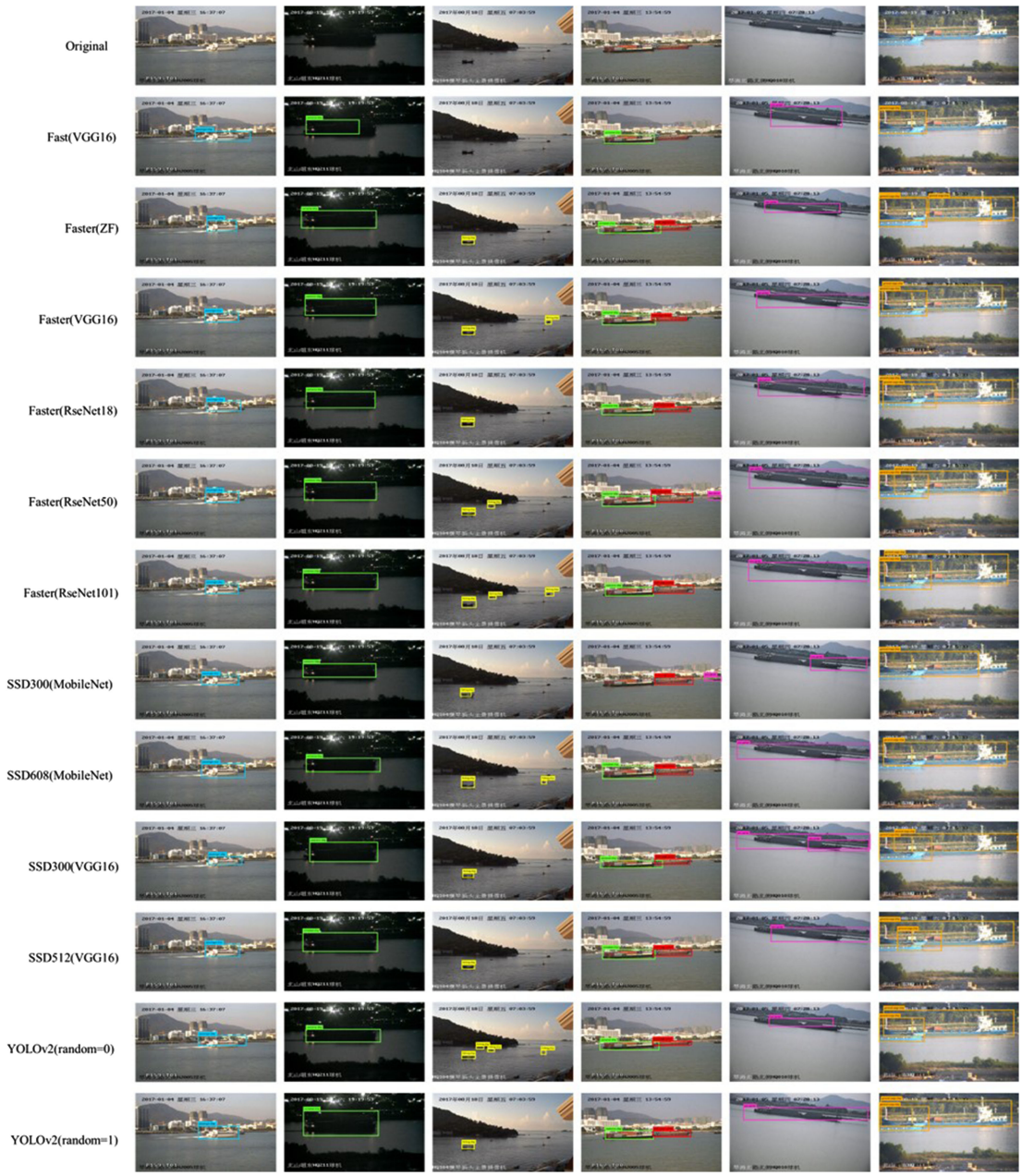


Fig. 16. Ship detection results (including successful and failure cases). The original images are shown in the first row. The next few rows are the experimental results of the corresponding method in the first column.

PASCAL VOC dataset yields very poor detection results on ships under the real sea scenarios. In constrast, the trained model of SeaShips still results in good detection results on both testing PASCAL VOC dataset and testing SeaShips dataset. This is a good evidence that our dataset shows strong generalization ability across dataset experiments.

E. Real-Time Video Scenario

Finally, we implement the above trained model on different datasets in the Hengqin roundabout monitoring system for real-time ship detection. In theory, they can be fully used as a tracking system. Because in the monitoring system, every camera produces a video of 25 FPS continuously, the real-time performance needs to be satisfied.

In this test, we obtain RTSP addresses of the cameras in the Hengqin roundabout monitoring system. Aiming at detecting different ship types in real time, we then connect the trained YOLO v2 models on SeaShips and PASCAL VOC2007 to these addresses. We calculate the time to get images from RTSP and display detections on the screen together, obtaining real-time detection speed of 91 FPS. We also select 50 video clips (75,000 frames) and measure their misdetection rate and omission rate (opposite to precision and recall). Table VI shows the statistics for real-time video detection. Experimentally, our model is able to achieve real-time detection and maintain good accuracy at the same time.

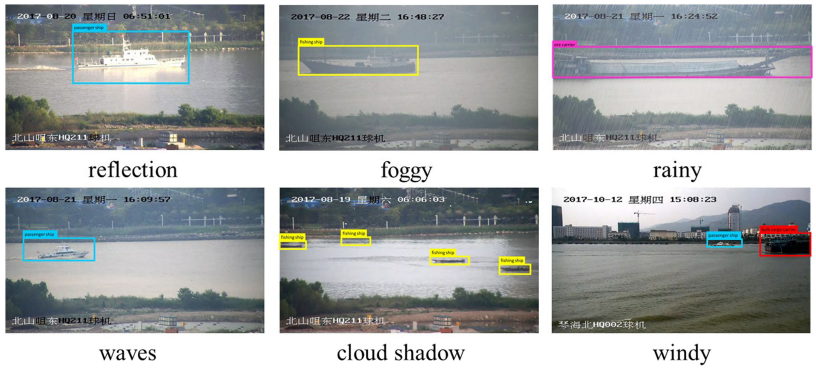


Fig. 17. The ship detection results under some extreme weather conditions.

TABLE V
DETECTION RESULTS ON DIFFERENT TRAINING SETS

Training	Testing	Recall	Precision	mAP
VOC 2007	SeaShips	0.4144	0.2793	0.2803
SeaShips	VOC2007	0.6886	0.6526	0.6051
SeaShips	SeaShips	0.8537	0.8085	0.7906

TABLE VI
DETECTION PERFORMANCE FOR REAL-TIME VIDEO

Threshold	Training	Misdetection	Omission
0.5	VOC 2007	0.1069	0.3934
	SeaShips	0.0653	0.0475
0.7	VOC 2007	0.0571	0.5793
	SeaShips	0.0107	0.0533

Although YOLO v2 model trained on PASCAL VOC2007 can detect some boats, the omission rate is high (nearly 40%). Under oblique or non-orthographic viewpoints, the model often fails to detect any objects. And when we set the detection threshold to a higher value like 0.7, the omission rate also increases (over 50%). On the contrary, with a detection threshold of 0.7, the omission rate of our model trained on the proposed SeaShips does not increase, because the model has already fully learned the unique characteristics of various ships. Considering the real-time detection performance of two datasets, it is better to use our proposed dataset for practical marine monitoring applications.

VI. CONCLUSION

In this paper, aiming to support further research in ship detection field, we propose a new diverse ship dataset named SeaShips. It contains accurate bounding box annotations for six ship types in the PASCAL VOC format. We also described the detailed design of the dataset, including its acquisition procedure, annotation method, and different variations. SeaShips consists of 31455 images of six ship types across 4 months, various scales, viewpoints, backgrounds, illumination and diverse occlusion conditions. Therefore, it can be used as a benchmark dataset for ship detection. Adopting similar evaluation protocol of PASCAL VOC, we provided experimental results of three baseline detectors on the dataset. Through analyzing the results, we concluded the performances of each detector and the difficulties of ship detection.

Future research can be focused on the following aspects: 1) With the help of established data sets, improve the detector to better handle small vessels such as fishing boats; 2) Advance novel detection algorithms for ship detection under occlusion; 3) Apply our data set to diverse learning-based detection algorithms and practical maritime decision-making systems.

REFERENCES

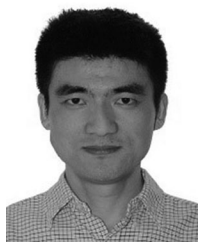
[1] C. Zhu, H. Zhou, R. Wang, and J. Guo, "A novel hierarchical method of ship detection from spaceborne optical image based on shape and texture features," *IEEE Geosci. Remote Sens. Lett.*, vol. 48, no. 9, pp. 3446–3456, Sep. 2010.
[2] G. Yang *et al.*, "Ship detection from optical satellite images based on sea surface analysis," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 3, pp. 641–645, Mar. 2014.
[3] S. Sheng *et al.*, "Unsupervised ship detection based on saliency and S-HOG descriptor from optical satellite images," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 7, pp. 1451–1455, Jul. 2015.
[4] X. Wang, A. Shrivastava, and A. Gupta, "A-fast-RCNN: Hard positive generation via adversary for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3039–3048.
[5] S. Azadi, J. Feng, and T. Darrell, "Learning detection with diverse proposals," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 7369–7377.
[6] K. Kang, W. Ouyang, H. Li, and X. Wang, "Object detection from video tubelets with convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 817–825.
[7] W. Ouyang, X. Wang, C. Zhang, and X. Yang, "Factors in finetuning deep model for object detection with long-tail distribution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 864–873.
[8] J. Cao, Y. Pang, and X. Li, "Pedestrian detection inspired by appearance constancy and shape symmetry," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1316–1324.
[9] J. Deng *et al.*, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.
[10] M. Everingham, L. V. Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, 2010.
[11] T. Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.
[12] W. Gao *et al.*, "The CAS-PEAL large-scale Chinese face database and baseline evaluations," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 38, no. 1, pp. 149–161, Jan. 2008.
[13] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," in *Proc. Workshop Faces Real-Life Images, Detect., Alignment, Recognit.*, 2008, pp. 1–15.
[14] V. Jain and E. Learned-Miller, "FDDB: A benchmark for face detection in unconstrained settings," Univ. Massachusetts Amherst, Amherst, MA, USA, Tech. Rep. UM-CS-2010-009, 2010.

[15] P. Dollar, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 743–761, Apr. 2012.
[16] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 3354–3361.
[17] S. Zhang, R. Benenson, and B. Schiele, "CityPersons: A diverse dataset for pedestrian detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4457–4465.
[18] R. Smith *et al.*, "End-to-end interpretation of the French street name signs dataset," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 411–426.
[19] G. Cutter, K. Stierhoff, and J. Zeng, "Automated detection of rockfish in unconstrained underwater videos using haar cascades and a new image dataset: Labeled fishes in the wild," in *Proc. Appl. Comput. Vis. Workshops*, 2015, vol. 38, no. 2, pp. 57–62.
[20] P. Welinder *et al.*, "Caltech-UCSD birds 200," California Inst. Technol., Pasadena, CA, USA, Tech. Rep. CNS-TR-2010-001, 2010.
[21] A. Krizhevsky, "Learning multiple layers of features from tiny images," Technical report, University of Toronto, vol. 1, no. 4, p. 7, 2009.
[22] G. Griffin, A. Holub, and P. Perona, "Caltech-256 object category dataset," California Inst. Technol., Pasadena, CA, USA, Tech. Rep. CNS-TR-2007-001, 2007.
[23] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 818–833.
[24] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1–9.
[25] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.
[26] C. Szegedy *et al.*, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2818–2826.
[27] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014.
[28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[29] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-ResNet and the impact of residual connections on learning," 2016, *arXiv:1602.07261*.
[30] S. Xie, R. Girshick, P. Dollr, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," 2016, arXiv:1611.05431.
[31] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Region-based convolutional networks for accurate object detection and segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 1, pp. 142–158, Jan. 2016.
[32] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015.
[33] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1440–1448.
[34] S. Ren, K. He, R. Girshick, and J Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.
[35] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. Conf. Neural Inf. Process. Syst.*, 2016, pp. 379–387.
[36] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," 2017, *arXiv:1703.06870*.
[37] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 779–788.
[38] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.*, 2015, pp. 21–37.
[39] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6517–6525.
[40] R. Zhang, J. Yao, K. Zhang, C. Feng, and J. Zhang, "S-CNN ship detection from high-resolution remote sensing images," in *Proc. Int. Congr. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, 2016, pp. 423–430.
[41] J. Xu, X. Sun, D. Zhang, and K. Fu, "Automatic detection of inshore ships in high-resolution remote sensing images using robust invariant generalized hough transform," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 12, pp. 2070–2074, Dec. 2014.
[42] G. Liu *et al.*, "A new method on inshore ship detection in high-resolution satellite images using shape and context information," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 3, pp. 617–621, Mar. 2014.
[43] M. Tello, C. Lopez-Martinez, and J. J. Mallorqui, "A novel algorithm for ship detection in SAR imagery based on the wavelet transform," *IEEE Geosci. Remote Sens. Lett.*, vol. 2, no. 2, pp. 201–205, Apr. 2005.
[44] X. Xing, K. Ji, L. Kang, and M. Zhan, "Review of ship surveillance technologies based on high-resolution wide-swath synthetic aperture radar imaging," *J. Radars*, vol. 4, no. 1, pp. 107–121, 2015.
[45] W. Tao, H. Jin, and J. Liu, "Unified mean shift segmentation and graph region merging algorithm for infrared ship target segmentation," *Opt. Eng.*, vol. 46, no. 12, pp. 127002-1–127002-7, 2007.
[46] S. R. Rotman, "Region-of-interest-based algorithm for automatic target detection in infrared images," *Opt. Eng.*, vol. 44, no. 7, pp. 166–169, Jul. 2005.
[47] M. Ren and Z. Tang, "One effective method for ship recognition in ship locks," *Proc. SPIE*, vol. 3720, pp. 467—472, Apr. 1999.
[48] J. Wu, "Ship classification," in *Introduction to Ship and Ocean Engineering*, 1st ed. Guangzhou, China, South China University of Technology Press, Nov. 2013.
[49] A. G. Howard *et al.*, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.

Zhenfeng Shao received the Ph.D. degree from Wuhan University, Wuhan, China, in 2004. He is currently a Professor with the State Key Laboratory of Information Engineering in Surveying, Mapping, and Remote Sensing, Wuhan University. His research interest focuses on computer vision.

Wenjing Wu received the B.Eng. degree in remote sensing science and technology from Wuhan University, Wuhan, China, in 2016, where she is currently working toward the M.Eng. degree in photogrammetry and remote sensing at the State Key Laboratory of Information Engineering in Surveying, Mapping, and Remote Sensing. Her research interests include image processing and object detection.

Zhongyuan Wang received the Ph.D. degree in communication and information systems from Wuhan University, Wuhan, China. He is currently a Professor with the Computer School, Wuhan University. His research interests include video compression, image processing, and multimedia big data analytics.



Wan Du received the Ph.D. degree in electronics from the University of Lyon (Ecole centrale de Lyon), Lyon, France, in 2011. He was a Research Fellow with the School of Computer Science and Engineering, Nanyang Technological University, Singapore, from 2011 to 2017. He is currently an Assistant Professor with the Computer Science and Engineering Department, University of California, Merced, CA, USA. His research interests include the Internet of Things, cyber-physical system, wireless networking systems, and mobile computing systems.

Chengyuan Li received the B.Eng. degree in geographic information systems from Zhengzhou University, Zhengzhou, China, in 2014. He is currently working toward the M.Eng. degree in surveying and mapping engineering at the State Key Laboratory of Information Engineering in Surveying, Mapping, and Remote Sensing, Wuhan University, Wuhan, China. His research interests include image processing and object detection.