Smart Structures and Systems, *Vol. 23, No. 4 (2019) 359-371*
DOI: https://doi.org/10.12989/sss.2019.23.4.359

Real-time geometry identification of moving ships by computer vision techniques in bridge area

Shunlong Li*[1], Yapeng Guo[1], Yang Xu[2] and Zhonglong Li[1]

[1]*School of Transportation Science and Engineering, Harbin Institute of Technology, 73 Huanghe Road, Harbin 150090, China*
[2]*School of Civil Engineering, Harbin Institute of Technology, 73 Huanghe Road, Harbin 150090, China*

(Received September 9, 2018, Revised January 19, 2019, Accepted March 11, 2019)

Abstract. As part of a structural health monitoring system, the relative geometric relationship between a ship and bridge has been recognized as important for bridge authorities and ship owners to avoid ship–bridge collision. This study proposes a novel computer vision method for the real-time geometric parameter identification of moving ships based on a single shot multibox detector (SSD) by using transfer learning techniques and monocular vision. The identification framework consists of ship detection (coarse scale) and geometric parameter calculation (fine scale) modules. For the ship detection, the SSD, which is a deep learning algorithm, was employed and fine-tuned by ship image samples downloaded from the Internet to obtain the rectangle regions of interest in the coarse scale. Subsequently, for the geometric parameter calculation, an accurate ship contour is created using morphological operations within the saturation channel in hue, saturation, and value color space. Furthermore, a local coordinate system was constructed using projective geometry transformation to calculate the geometric parameters of ships, such as width, length, height, localization, and velocity. The application of the proposed method to in situ video images, obtained from cameras set on the girder of the Wuhan Yangtze River Bridge above the shipping channel, confirmed the efficiency, accuracy, and effectiveness of the proposed method.

Keywords: real-time ship detection; geometric parameter identification; single shot multibox detector (SSD); deep learning (DL); computer vision (CV)

1. Introduction

A considerable number of bridges have been constructed over rivers and coastal regions in recent years. Piers of bridges suffer from not only wave forces and water corrosion but also a type of fatal accidental load, i.e., the collision load of ships (Goerlandt *et al*. 2015). Once a ship–bridge collision tragedy occurred, significant losses would be incurred for both the bridges and ships. Thus, real-time geometric parameter identification of moving ships is paramount for pre-disaster warning.

Structural health monitoring systems have been widely implemented in different types of bridges, especially long-span ones (Li *et al.* 2012). Various approaches based on different types of sensors have been developed for environmental load monitoring, global and local structural responses monitoring, and damage detections. For ship-bridge collision monitoring, accelerometers were installed at the bottom of piers and vibration-based methods for piers were employed to evaluate its service state after a collision occurred. Further, a number of visible light sensors (i.e., cameras) have been installed on specific locations of bridges to collect the image data of the shipping channels (Ou and Li 2010). Currently, typical video monitoring practice employs human resources to monitor screens including images and to analyze the risk of such collision accidents. However, humans have inevitable limitations, such as subjectivity and visual fatigue, possibly leading to more misjudgments for ship detection and early warning failure of ship collision.

Recently, computer vision (CV) has exhibited great achievements and such promising results led to the application of related techniques in civil engineering field. CV can be utilized to meet the requirements of long-term continuous service and the independence from a human's subjective experience. For ship collision monitoring, the most suitable technique is object detection, which is one of the most popular research branches in CV and allows machines to identify and locate specific objects (Stockman and Shapiro 2001, Andrew 2004, Szeliski 2011). Object detection techniques can be mainly used for static features recognition (mostly including damage on the surface of structures) and dynamic object identification.

Static feature extraction is a widely used process in object detection. A statistical filter for the crack detection in pipes was proposed using a two-step approach (Sinha and Fieguth 2006). A survey and evaluation of reliable approaches for automatic image-based defect detection of bridge structures were conducted to determine their advantages and disadvantages (Jahanshahi *et al*. 2009). Oh *et al*. (2009) introduced a robotic system for inspecting the safety status of bridges based on machine vision, composed of three parts, namely, a specially designed car, a robot mechanism and control system for mobility, and a machine vision system for automatic crack detection. This system

*Corresponding author, Professor
E-mail: lishunlong@hit.edu.cn

ISSN: 1738-1584 (Print), 1738-1991 (Online)

was only an attempt for crack detection application by using machine vision, but the results were far from that of human judgment. However, this attempt had motivated researchers who focus on crack detection to contribute more to computer vision. An image-based framework was proposed to identify concrete surface cracks by using optical cameras as the source of images (Chen and Hutchinson 2010). A complete procedure for automated crack assessment based on adaptive digital image processing has been proposed, crack objects are extracted from the images using the subtraction with median filter and the local binarization using the Niblack's method (Liu *et al.* 2014). Except for crack identification, static feature extraction also has other applications. Zhu *et al.* (2010) presented a new fault diagnosis method for rotating machinery with artificial immune algorithm by using image recognition technology, improving the accuracy rate and diagnosis system robust quality effectively. An intelligent bridge inspection system by using robot and IT technology that can capture images of the bridge condition to identify its structural safety was also conducted (Lee *et al*. 2011). An automated framework for rapid post-earthquake building evaluation was proposed (German *et al*. 2013). For human loading recognition, Eum *et al*. (2015) proposed a novel method for spotting and recognizing continuous human actions using a vision sensor, composed of tracking the displacement trajectories of individuals and crowds. And the reconstruction of load time histories based on several computer vision techniques had also been explored recently (Celik *et al*. 2018, Celik *et al.* 2019). Due to the lack of information in time domain, static feature recognition has usually been employed on events developing slowly relatively.

Dynamic object identification is another crucial field for researchers in object detection. Non-contact measurement for responses of structure by using object detection techniques has attracted researchers. Ye *et al.* (2013) developed a vision-based dynamic displacement measurement system with the use of digital image processing technology, which was featured by its distinctive characteristics in non-contact, long-distance, and high-precision structural displacement measurement. They also presented three kinds of image processing algorithms for structural dynamic displacement measurement, i.e., the grayscale pattern matching (GPM) algorithm, the color pattern matching (CPM) algorithm, and the mean shift tracking (MST) algorithm. A vision-based system programmed with the three image processing algorithms was developed for multi-point structural dynamic displacement measurement (Ye *et al*. 2016). Yang *et al.* (2017) explored advanced computer vision and video processing algorithms to develop a novel video measurement and vision-based operational (output-only) modal analysis method that alleviated the need of structural surface preparation associated with existing vision-based methods and can be implemented in a relatively efficient and autonomous manner with little user supervision and calibration. Kong and Li (2018) proposed an approach based on structural surface motion tracking to detect and localize fatigue cracks in a video stream. A novel computer vision technique was proposed for debris flow detection which is feature-based that can be used to construct a debris flow event warning system (Lin *et al.* 2015). For specific macro objects detection, Huang and Ma (2012) proposed a video-based moving object detection algorithm for vehicle localization. Chen *et al*. (2016) presented a method to identify the spatio–temporal distribution of vehicle loads for long-span bridges through computer vision technology by combining the monitoring information of the weigh-in-motion (WIM) system at one cross section and the camera along the bridge. For ship detection, Wang (2011) introduced a robust detection and tracking method that can detect and track ship targets in cluttered forward-looking infrared image sequences. Liu *et al*. (2013) developed a framework that involves an effective pre-processing method based on improved mean shift smoothing and a hierarchical approach, which was applied for ship target and motion direction detections. A method using edge-based segmentation and histogram of oriented gradient (HOG) with the ship size ratio was proposed, which could prevent a marine collision accident by detecting ships at close range (Hyukmin *et al.* 2015). The conventional vision-based methods aim to process the pixels and regions in images directly and provide an output in the form of regions of interest. However, such idea has limitations, for example, the requirement of predesigned filter-based detectors, the need to assume the crack geometry, and restrictions on specific optical devices or images (Xu *et al*. 2018). Further, their performance primarily depends on manual parameter settings and the simplification of actual scenarios to suppress interference.

For a better understanding, deep learning algorithms with computational architectures inspired by human mind have been developed in CV, especially in the branch of object detection. Unlike regular neural network, deep learning can extract high-level features of objects (Lecun *et al.* 2015). The restricted Boltzmann machine (RBM) and deep convolutional neural networks (CNNs) have been widely used for feature learning and classification of images. Xu *et al.* (2017) established an RBM framework for crack identification on steel structure surfaces. Recently, CNNs have been extensively applied for the detection of specific features and moving objects in civil engineering fields (Kulchandani and Dangarwala 2015). Chi and Caldas (2011) proposed an exploratory method for automated object identification by using standard video cameras on construction sites. This method supported a real-time detection and classification of mobile heavy equipment and workers based on a two-layer neural network. Makantasis *et al.* (2015) developed a fully automated tunnel assessment approach by using a CNN to construct high-level features to detect damages. Yeum *et al.* (2018) introduced a novel and powerful method for post-disaster evaluation by autonomously processing and analyzing large visual data by using a CNN algorithm. Narazaki *et al*. (2018a) first described the basic algorithm of the multi-scale convolutional neural networks (multi-scale CNNs), which were implemented to perform pixel-wise classification tasks, to solve the bridge component recognition task. They also investigated automated bridge component recognition using video data, where the information from the past

frames was used to augment the understanding of the current frame. And they creatived a new simulated video dataset to train the machine learning algorithms (Narazaki *et al*. 2018b). A novel damage localization and classification technique, recognizing 6 different types of damage, was proposed to carry out a pixel-wise classification of each image at multiple scales, using a deep convolutional neural network (Hoskere *et al.* 2018a). They also proposed a new framework to generate vision-based condition-aware models, which can serve as the basis for speeding up or automating higher level inspection decisions (Hoskere *et al.* 2018b). For ship detection, Bentes et al. (2017) presented a full workflow for synthetic aperture radar (SAR) maritime target detection and classification on TerraSAR-X high-resolution image, and CNNs were cross evaluated using a typical dataset composed of five maritime classes. Yao et al. (2017) introduced another framework for efficient ship detection through optical remote sensing images based on deep CNNs. With the appearance of region-based convolutional neural network (R-CNN), deep ConvNets with two-stage architectures have significantly improved object detection accuracy (Girshick *et al.* 2014). Fast R-CNN (Girshick 2015) and Faster R-CNN (Ren *et al.* 2015) have been proposed to improve the object detecting speed (He *et al.* 2017). A super-fast algorithm called YOLO with a single neural network could predict the bounding boxes and class probabilities directly from full images in one evaluation (Redmon *et al*. 2016). The algorithm framed object detection as a regression problem to spatially separated bounding boxes and the associated clustering probabilities. Although a series of R-CNN algorithms could yield increasingly high accuracies, they have low speed, whereas YOLO has an opposite condition. To utilize both the accuracy advantage of the R-CNN series and the speed advantage of YOLO, a single shot multibox detector (SSD) was proposed by Liu *et al.* (2016). Although the deep learning algorithms have been successfully used in constrained object categories, such as faces (Vaillant *et al.* 1994, Rowley *et al.* 1998) and pedestrians (Sermanet *et al.* 2013), such approaches are profoundly unsatisfactory (Gu *et al.* 2012) in moving objects, especially for geometrical parameter identification.

This study proposes a novel method for real-time geometric parameter identification of moving ships as part of the bridge health monitoring system based on the SSD, with transfer learning techniques and monocular vision for ship–bridge collision avoidance in the bridge area. Fig. 1 summarizes the identification framework, which is composed of two major steps. In the first step, image sequences obtained from cameras would be processed by using ship detection methodology based on the SSD, where the ship region boxes could be obtained in the coarse scale. The second step is to generate a ship contour by morphological operations within the saturation channel in the hue, saturation, and value (HSV) color space. Furthermore, a local coordinate system was constructed using projective geometry transformation to calculate the geometric parameters of ships in a fine scale (e.g., width, length, height, localization, and velocity). The aforementioned steps would be applied to in situ field tests for demonstration.

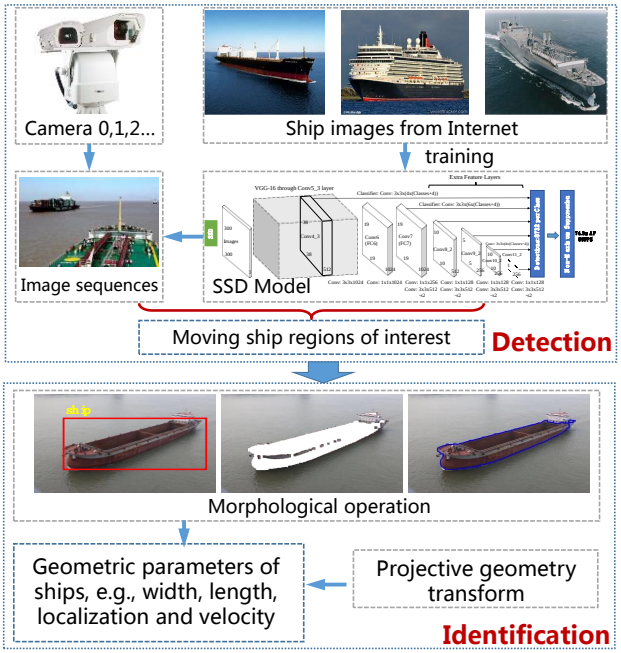


Fig. 1 Framework of real-time geometric parameter identification of moving ships in bridge area using novel computer vision techniques

2. Ship detection based on a SSD

2.1 SSD modelling and training

The SSD is based on a feedforward CNN (Fig. 2). It produces a series of fixed-size bounding boxes together with scores of object category instances. The non-maximum suppression technique is employed for filtering to produce the final detection results (Liu *et al*. 2016).

According to the base network VGG-16 (Simonyan and Zisserman 2014), the SSD surrenders two fully connected layers and transforms FC_6 and FC_7 into convolutional layers Conv_6 and Conv_7, respectively. Thereafter, another eight convolutional layers, namely, Conv6_1, Conv6_2, Conv7_1, Conv7_2, Conv8_1, Conv8_2, Conv9_1, and Conv9_2, are added to the modified VGG-16. The detailed architecture of the SSD is shown in Fig. 2, but the ReLU activation function is not shown for brevity. The most innovative part of the SSD algorithm is to directly predict the category and coordinates of the bounding box for multiple targets via integrated multiscale feature maps. Six different feature maps that generated from convolution layers (i.e., Conv4_3, Conv_7, Conv6_2, Conv7_2, Conv8_2, and Conv9_2) are convoluted using two separate 3 × 3 convolution filters. One convolution output is the confidence for classification, and each default box generates 2 confidences. The other convolution output is the localization for regression, and each default box generates four coordinate values (center point coordinate x, y, width, height). In addition, the aforementioned six different

convolution layers are used for a fixed number of default box generations (i.e., 8732 in this study) through the prior-box layer. Finally, the confidence, localization, and default box generation results are merged separately and passed to the loss layer.

The entire training process can be described as follows. The SSD only requires an input image and ground truth boxes for each object during training. In a convolutional manner, a small set of default boxes of various aspect ratios at each location in several feature maps with different scales are evaluated. For each default box, both the shape offsets and confidences for all object categories would be predicted. During the training process, these default boxes are determined to correspond to the ground truth boxes, and the SSD network are trained accordingly. The SSD training objective is derived from the multibox objective, but it is extended to handle multiple object categories. The overall objective loss function is a weighted sum of the localization loss and the confidence loss in Eq. (1)

$$L(x,c,l,g)=\frac{1}{N}\left[L_{conf}(x,c)+\alpha L_{loc}(x,l,g)\right]$$

$$L_{loc}(x,l,g)=\sum_{i\in Pos}\sum_{m\in\{cx,cy,w,h\}}x_{ij}^{k}\,\mathrm{smooth}_{L1}\left(l_i^m-\hat{g}_j^m\right)$$

$$\hat{g}_j^{cx}=\left(g_j^{cx}-d_i^{cx}\right)/d_i^{w}\qquad \hat{g}_j^{cy}=\left(g_j^{cy}-d_i^{cy}\right)/d_i^{h} \tag{3}$$

$$\hat{g}_j^{w}=\log\left(\frac{g_j^{w}}{d_i^{w}}\right)\qquad \hat{g}_j^{h}=\log\left(\frac{g_j^{h}}{d_i^{h}}\right)$$

$$L_{conf}(x,c)=-\sum_{i\in Pos}^{N}x_{ij}^{p}\log\left(\hat{c}_i^{p}\right)-\sum_{i\in Neg}\log\left(\hat{c}_i^{0}\right)\quad\text{where}\quad\hat{c}_i^{p}=\frac{\exp\left(c_i^{p}\right)}{\sum_{p}\exp\left(c_i^{p}\right)}$$

where N is the number of matched default boxes and the weight term α is set to 1 by cross validation.

2.2 Model transfer learning for moving ship detection

2.2.1 Training ship images preparation

Inspired by ImageNet (Deng *et al.* 2009), ship images used for the SSD training were searched using Microsoft Bing and downloaded from the Internet.

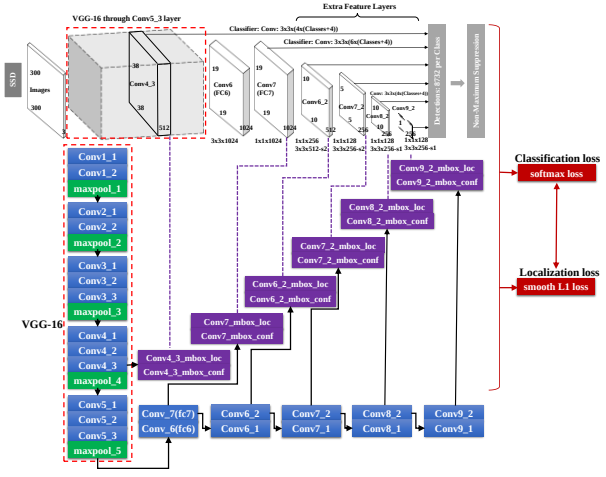


Fig. 2 Detailed SSD network architecture

(a) cargo ships (b) passenger ships (c) navy ships

Fig. 3 Representative raw images from training dataset

A total of 500 images including cargo, passenger, and navy ships of different sizes were captured from different angles of view under various light conditions. The representative raw images from the training dataset are shown in Fig. 3.

To establish a highly robust SSD training model to various input object sizes and shapes, the data augmentation described by Liu *et al.* (2016) is employed in this study. Each downloaded ship image is randomly sampled by using one of the following three options (Liu *et al.* 2016): (1) use the entire original input image; (2) sample a patch such that the minimum Jaccard overlapping with the objects is 0.1, 0.3, 0.5, 0.7, or 0.9; and (3) randomly sample a patch. Thereafter, each sampled patch is adjusted to a fixed size and horizontally flipped with a probability of 0.5. Thus, the training ship images can be increased from 500 to 1500. Further, all the sampled ship images are resized to 300 × 300 pixels as the inputs.

2.2.2 Transfer learning fine-tuning process

Transfer learning allows rapid progress or improved performance of modelling new tasks through the transfer of knowledge, already learned from approximate tasks (Pan and Yang 2010). In this study, the first five convolutional parts of VGG-16 (Simonyan and Zisserman 2014) pre-trained model by ImageNet (Deng *et al.* 2009) is employed for fine-tuning, which contains lots of universal low-level features for distinguishing different objects. And SSD fine-tuned its own weights for added layers based on weights in the aforementioned convolutional parts.

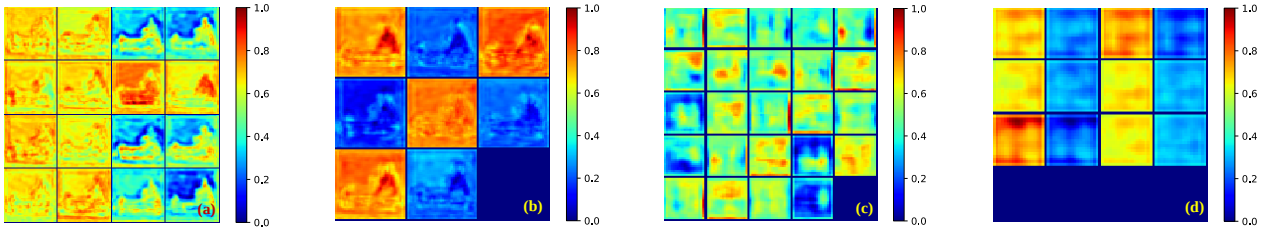
Deep learning frameworks, such as Caffe, Tensorflow, and PyTorch, are generally recognized to dramatically improve the efficiency on building networks. The SSD fine-tuning model employed in this study was conducted using the Caffe framework. However, configuring the optimal hyperparameters before fine-tuning is tedious and time consuming. Thus, most of the hyperparameters used herein adopted the default configuration recommended by Liu *et al.* (2016). Particularly, the learning rate is a crucial parameter for the SSD network training process. Caffe provides a series learning rate decay policies, such as fixed,

(a) Input raw ship image (b) resized image for visualization

Fig. 4 Ship image for visualization

(a) Feature maps of layer Conv1_1 (b) Learned filters of layer Conv1_1

Fig. 5 Feature maps and learned filters of layer Conv1_1



(a) Conv4_3_mbox_loc (b) Conv4_3_mbox_conf (c) Conv7_mbox_loc (d) Conv7_mbox_conf

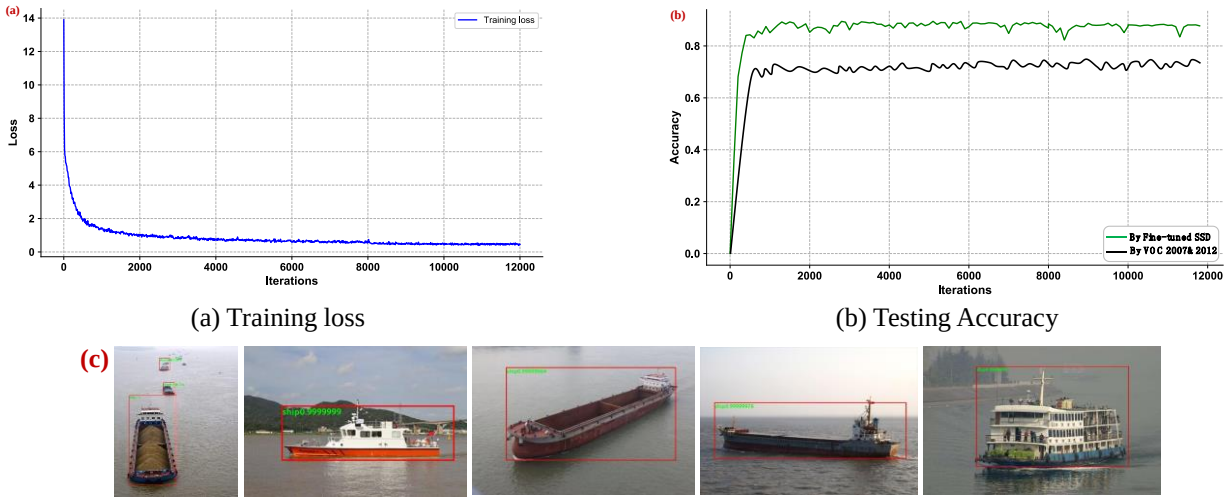
Fig. 6 Feature maps of layer Conv4_3_mbox and Conv7_mbox

step, inv, multistep, and sigmoid. The typically employed policy is the multistep for piecewise. That is, the entire SSD network training process incorporating N iterations can be divided into several parts (N_i, i = 1, 2, 3…). When the optimization iteration number reaches Ni, the learning rate decreases as designated. When using the multistep method, the training process can be optimized to obtain a smooth decreasing loss. In this study, the learning rate variation followed a piecewise function. The fundamental learning rate was set to λ=0.01, the total iteration was set to 12,000, N_1 = 8000 with λ=0.001, and N_2 = 10000 with λ=0.0001. The training, validation, and test sets were independent and randomly sampled from the entire dataset of 1500 images mentioned previously with a proportion of 70%, 10%, and 20%, respectively (1500 in total, i.e., 1050 for training, 150 for validation, and 300 for testing).

Intuitionally visualizing SSD network features was recognized as a common practice for deep understanding of different layers and network improvements. The assumed input raw and resized ship image is shown in Fig. 4. Subsequently, the input ship image data starts flowing in the network from layer Conv1_1 and Conv1_2 and to layer Conv9_1 and Conv9_2, respectively.

To the best of our knowledge, the first several convolutional layers, such as Conv1_1, typically contains low-level features that a human can decipher. As shown in Fig. 5(a), the output of the convolutional layer Conv1_1 (i.e., the Conv1_1 feature maps) indicates that the primary function of this layer is to extract the texture of objects, background, and certain colour information. This inference is also proven in Fig. 5(b), from the learned filters of layer Conv1_1.

Layers Conv4_3_mbox, Conv7_mbox, Conv6_2_mbox, Conv7_2_mbox, Conv8_2_mbox, and Conv9_2_mbox were employed to generate default boxes as multiscale feature maps. Further, the 3 × 3 convolution filter was applied twice to extract the classification and localization information, separately. Thus, twelve feature maps would be available to make the decisions, and the receptive fields of these feature maps are gradually increasing. As shown in Figs. 6(a) and 6(b), the feature maps of layer Conv4_3_mbox for the localization and confidence can dominantly show the original shape contours of the investigated ships. Hence, the primary role of layer



(a) Training loss (b) Testing Accuracy

(c) Ship detection

Fig. 7 Training loss of model training process

Conv4_3_mbox was to detect relatively small objects. As also shown in the figures, the scales of the localization coordinates of every pixel in the feature map typically reach the extremum in the ship zone and the scale of the classification confidence. Each feature map corresponded with a convolutional filter and a different weight to contribute to the final results. Through layers Conv4_3_mbox_loc and Conv4_3_mbox_conf, the predictions and ground truth could flow into the loss function. The subsequent feature map layers worked in a similar manner, such as Conv7_mbox (Figs. 6(c) and 6(d). However, they exhibited larger receptive fields and captured larger objects.

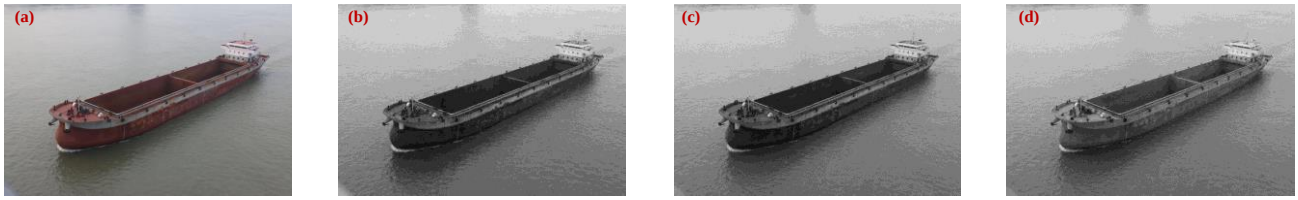
The recognition accuracy was defined as the proportion of the correct predictions to evaluate how the trained SSD network matched the overall targets. When a correct prediction occurs after comparing the output prediction label with the ground truth, the number of correct predictions adds 1, and the proportion is subsequently changed accordingly. This training network was tested after every 100 iterations. The training loss and testing recognition accuracy are shown in Fig. 5. As shown in Figs. 7(a) and 7(b), the training loss decreases from 13.8966 to 0.4255 with the iteration increases, whereas the testing recognition accuracy increases from 0% to 88.13%, which is higher than the model directly trained by using the VOC 2007&2012 dataset (Liu *et al.* 2016). Fig. 7(c) shows the intuitive results of ship detection by using the testing images. The ship detection results illustrate that the label-assigned boxes were as accurate as the ground truth and classification was performed at a relatively high probability. Thus, the trained SSD network can detect the ship region to a certain extent. However, for a more accurate calculation of the geometric parameter, the bounding boxes defined by only four coordinates are insufficient. Hence, segmentation techniques can be employed to obtain highly accurate bounding contours of ships.

3. Geometric parameter identification of moving ships

To ensure recalling rate of ships, bounding boxes predicted by the SSD network are always larger than ground truth, leading to inaccurate SSD detection results. But for geometric parameter identification, rough bounding boxes cannot meet the accuracy requirement for pixel-distance of ships. Thus, more accurate ship contours would be needed to calculate target parameters.

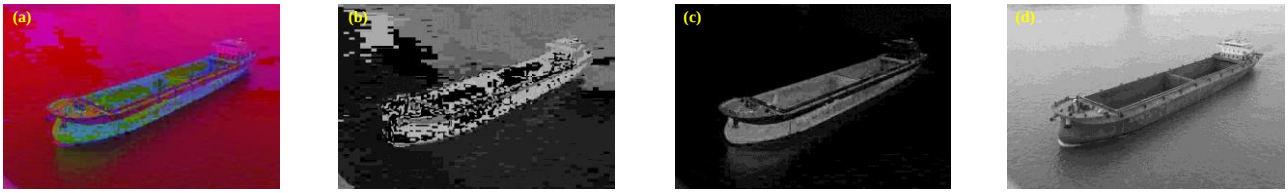
3.1 Ship contours segmentation

With hundreds of ships to be observed on the Yangtze River, a simple summary is concluded that almost all ships for civilian use are coated with colours that are noticeably different from the water, that is, the background (Fig. 8). Thus, utilizing the colour feature to realize segmentation is feasible in this study. The RGB colour space was initially considered. However, this description methodology was not intuitive, and the channel occupying the dominant position (Fig. 8) is difficult to distinguish. A colour space based on the concept of HSV is subsequently employed, as shown in Fig. 9. As presented in the figure, the saturation channel describes the purity of colour, and the purity of the water colour is relatively low compared with that of the ships. Thus, saturation information could be used to segment much more accurate contours of the investigated ships. To enhance the algorithm and eliminate the influence of the surroundings of the target ships, contour extractions should be performed in the detected region of interest. As shown in Fig. 10, a mask for the ship was created using the saturation channel image by the morphological operation. Subsequently, the contours could be captured by combining the bounding box and the saturation channel with a threshold area of 500 pixels. Given the water reflection interference, the contours extracted in the entire image perform worse than that in the bounding box. If the



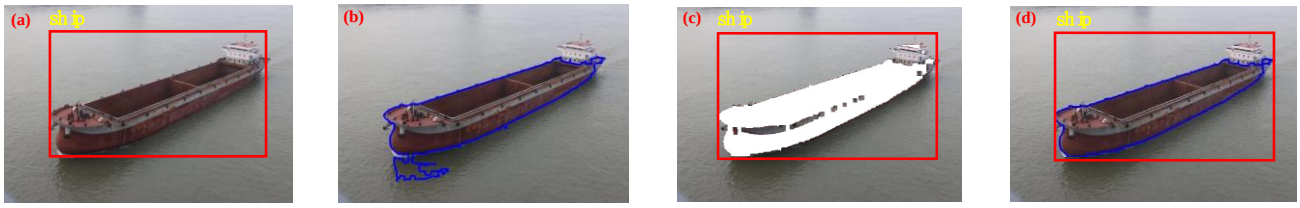
(a) Raw image (b) Blue channel (c) Green channel (d) Red channel

Fig. 8 Image of ships on Yangtze River in BGR color space



(a) Raw image (b) Hue channel (c) Saturation channel (d) Value channel

Fig. 9 Image of ships on Yangtze River in HSV color space



(a) detected box (b) extracted contour in the whole image (c) mask in box (d) contours in box

Fig. 10 Difference between contours extracted in the whole image and bounding boxes

backgrounds of the ships are more complex, then the method of contour extracting by detecting the bounding boxes will achieve a clearer advantage.

The specific processes for the morphological operation are as follows: (1) thresholding the saturation channel image by using the Otsu algorithm (Otsu 1979) and (2) morphological operation for the binary image from (1) with different kernel sizes according to different types of ship. After performing the aforementioned processes, the bounding boxes would be transformed into contours.

3.2 Target geometric parameters identification

Stereo vision techniques are frequently used to acquire more accurate data of the target depth. However, high computing resource consumption of stereo renders real-time monitoring with high accuracy impossible. On the basis of the detection module with a monocular vision (Steger *et al.* 2018), a novel distance estimation methodology has been proposed to obtain the estimated geometric parameters of the target ships in this study. A camera is fixed on the girder of the bridge in the middle of the shipping channel. Fig. 11 illustrates the estimated geometric parameters. Different from stereo vision, additional constraints must be considered in monocular vision for the geometrical parameter identification, such as the width of the shipping

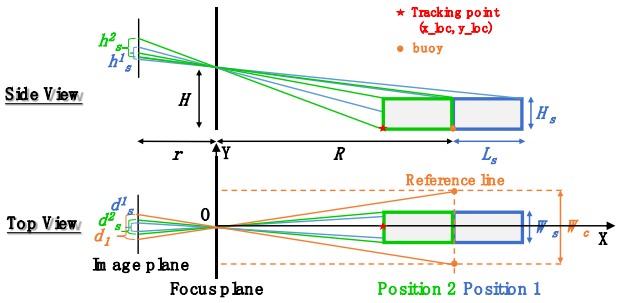


Fig. 11 Brief illustration for rough geometric parameters estimation based on pinhole model

channel, which could be easily obtained from the maritime bureau or measured by a laser rangefinder. In this study, a reference line is defined to control the horizontal distance where ships should be detected and is typically set to 200-400 m from the bridge. With these two aforementioned constraints, similar triangles could be utilized on the basis of pinhole cameras.

As shown in Fig. 11, from the hypothesis of the pinhole model, the width, length, height, location, and velocity of the target ships can be calculated.

(a) Width (W_s).

$$\frac{d_1}{W_c}=\frac{d^1_s}{W_s}=\frac{r}{R} \tag{2}$$

(b) Length (L_s) and height (H_s).

$$\begin{cases} h^1_s=\dfrac{r}{R}(H_s+\dfrac{H\cdot L_s}{L_s+R}) \\ h^2_s=\dfrac{r}{R'}(H_s+\dfrac{H\cdot L_s}{L_s+R'}) \end{cases} \tag{3}$$

in which

$$\frac{d^2_s}{W_s}=\frac{r}{R'} \tag{4}$$

(c)Locations of tracking point (***x_loc***, ***y_loc***).

$$\begin{cases} \dfrac{r}{x\_loc}=\dfrac{d_s}{W_s} \\ \dfrac{d_1}{W_c}=\dfrac{d_{y\_loc}}{y\_loc} \end{cases} \tag{5}$$

(d) Velocity (V_s).

$$V_s=\sqrt{(\frac{d(x\_loc)}{dt})^2+(\frac{d(y\_loc)}{dt})^2} \tag{6}$$

where W_s, H_s, L_s represent width, height and length of target ship; d_s, h_s are pixel width and pixel height of target ship; d^1_s, h^1_s indicate pixel width and pixel height of target ship at Position 1 (where the prow reaches Reference line (RL) on image); d^2_s, h^2_s are pixel width and pixel height of target ship at Position 2 (where the stern reaches RL on image); W_c , d_1 are the real and pixel width of shipping channel respectively; ***R*** is distance between focus plane and reference line; ***R'*** indicates distance between focus plane and ship prow at Position 2; ***r*** is equivalent focal length of camera; ***H*** indicates height of focus centre; ***x_loc***, ***y_loc*** indicates coordinates of tracking point; d_{y_loc} is pixel y coordinate of tracking point.

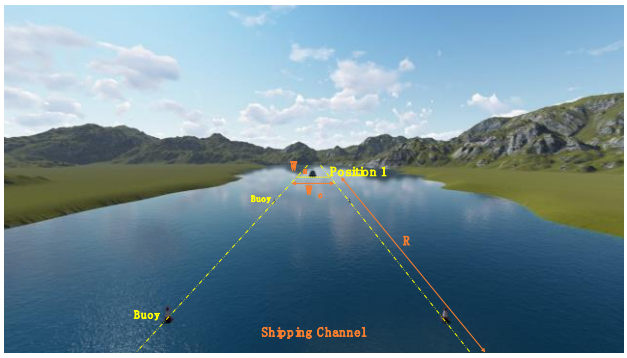


Fig. 12 Illustration for parameters at application stage

As shown in Fig. 12, at application stage, W_c, ***R*** and ***H*** could be known in advance. d_1 could be acquired from monitored image manually, approximately as constant. d_s, h_s, and d_{y_loc} would be identified directly from images. Except for these seven variables, all the others are calculated by Eqs. (2)-(6) automatically.

4. Field testing results and discussions

The Wuhan Yangtze River bridge is employed for field testing, which is a nine-span continuous beam bridge combining the highway and railway. It is the first bridge crossing the Yangtze River in China. The nine spans are all 128 m. The placement of the camera is shown in Fig. 13.

4.1 Ship detection results

A consumer-grade camera (SONY α6000) was used to acquire images and videos from different views with different focal lengths. In order to test the robustness of the proposed ship detection model, six representative images (1920×1080 pixels) were selected to identify the locations of ships. The six images are described as following, shown as Fig 14(a) single cargo ship from front view, (b) single cargo ship from oblique view, (c) two different-size ships, (d) two ships with wakes, (e) two ships with occlusion, (f) two passenger ships with reflection.

These images first were resized to 300×300 pixels to feed into the detection network. The testing hardware environment includes: an Intel Xeon E5-2620 v4 CPU, a NVIDIA GTX 1080Ti GPU, and 128GB memory. Fig. 14 illustrated the detection results. It could be seen from Fig. 14 that for each target ship, the network could identify satisfactory ship locations in less than 0.03s. However, small white ship in Fig. 14(d) could not be identified well enough, because of the influence from white wake.

4.2 Ship contours segmentation results

In the field testing, Otsu method was applied to each image first. Then according to binary results from Otsu, dilation operations were employed.

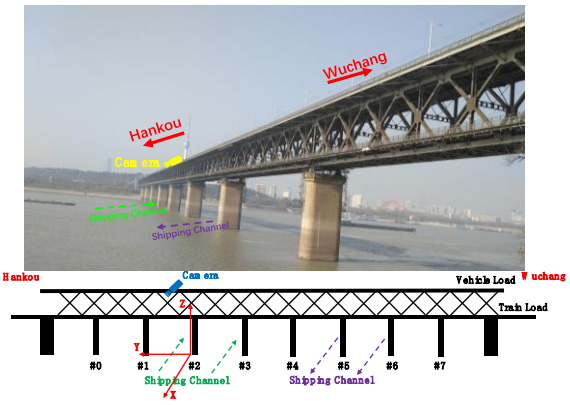


Fig. 13 Wuhan Yangtze River bridge and placement of camera

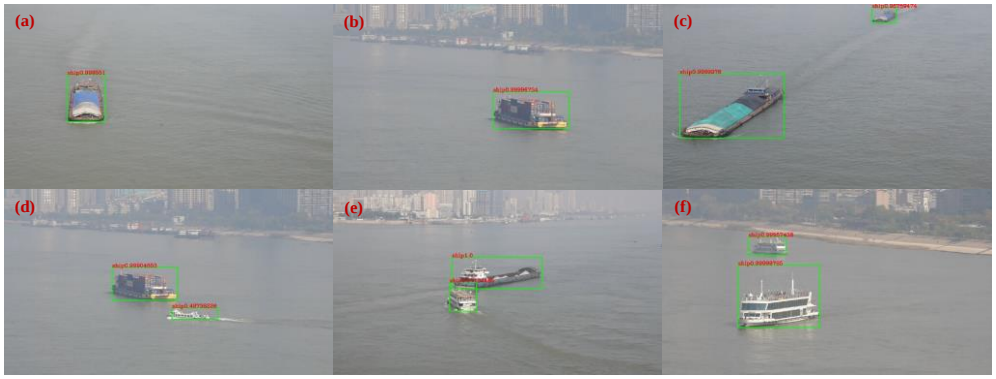


Fig. 14 Ship detection results in field testing

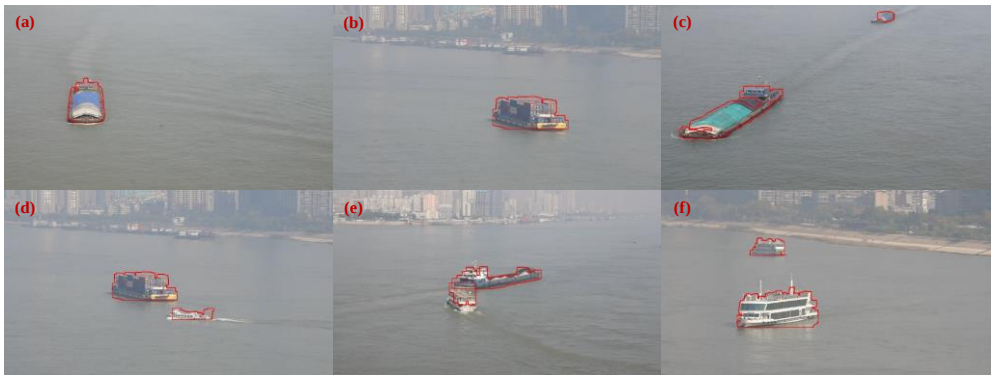


Fig. 15 Segmentation results in field test

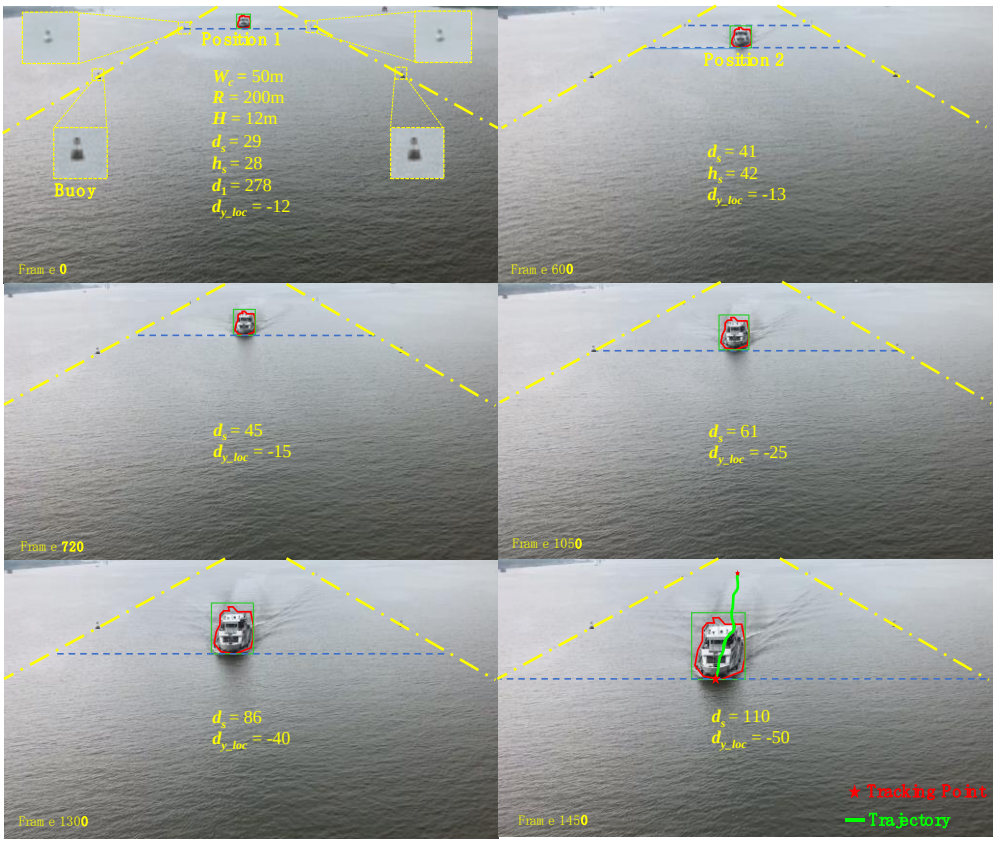


Fig. 16 Ship geometric parameters identification results in field test

Table 1 Kernel sizes and iterations used in morphological operations

No.	(a)	(b)	(c)	(d)	(e)	(f)
Kernel size	(5, 5)	(7, 7)	(7, 7)	(5, 5)	(5, 5)	(5, 5)
iterations	5	5	5	7	5	7

Table 2 Ship geometric parameters identification results

Items	Frame 0	Frame 600	Frame 720	Frame 1050	Frame 1300	Frame 1450
d_s (pixel)	29	41	45	61	86	110
h_s (pixel)	28	42	--	--	--	--
d_1 (pixel)	278	--	--	--	--	--
d_{y_loc} (pixel)	-12	-13	-15	-25	-40	-50
r (m)	10.96					
W_s (m)	5.30 (err = 3.64%)					
L_s (m)	14.46 (err = 9.62%)					
H_s (m)	4.31 (err = 4.22%)					
x_loc (m)	200	141.46	128.90	95.08	67.44	52.73
y_loc (m)	-2.19	-1.68	-1.77	-2.17	-2.47	-2.41
V_s (m/s)	--	2.93	3.14	3.07	3.3	2.94

Table 1 summarized the different dilation kernel sizes and iterations for aforementioned 6 images.

Fig. 15 gives the results of contours extraction. In most cases, the proposed algorithm could make satisfactory segmentations, such as Fig. 15(a). However, for ships with occlusions, the algorithm might not get ideal contours, like Fig. 15(e). Because in Otsu threshold process, the part of occlusion would be recognized for both emerging ships, leading to unsatisfactory edges division.

4.3 Ship geometric parameters identification results

To validate the effectiveness of proposed geometric parameters identification method, a water region nearby the Wuhan Yangtze River Bridge was employed. The real width of channel W_c, the distance between camera and reference line R and the height of camera H are 50m, 200m and 12m respectively. The placement of camera obeyed the requirements in Fig. 11. In other words, the image plane is perpendicular to horizontal plane, meanwhile optical axis should be parallel with the border line of shipping channel.

A representative 50-second surveillance video (30 frames per second) was employed for illustration with 1080×720 pixels. First, the bounding boxes of ship locations was detected by the transfer-learned SSD. Then Otsu and dilation operations with a 5×5 kernel and 5 iterations were applied to get ship contour, as shown in Fig. 16. Shipping channel was identified by four buoys in the video. Width (d_s) was defined as the distance between two edge points in the contour in Y-axis, height (h_s) in X-axis. The tracking point was defined as the center of d_s. d_s, h_s, d_1 and d_{y_loc} can be acquired directly from images automatically, as shown in Table 2.

Six representative frames including frame 0, frame 600, frame 720, frame 1050, frame 1300, and frame 1450 are plotted in Fig. 16. The ground truth parameters of the target ship are 16m×5.5m×4.5m (L×W×H) with a velocity of about 3m/s (provided by the ship owner). Estimated geometric parameters and corresponding errors are listed in Table 2.

The average processing time of one frame, including SSD ship detection, contours segmentation and geometric parameters identification, is 0.0682s, i.e., 14.66 FPS, with a CPU (Intel Xeon E5-2620 v4), a GPU (NVIDIA GTX 1080Ti) and 128GB memory. Considering that ships always move slowly and continuously, 14 FPS processing speed would be fast enough for real-time ship tracking in practice.

4.4 Discussions

From formula derivations mentioned in Section 3, the orientation of camera is crucial to geometric parameters identification results. Most probable reasons causing slight movements of cameras are installation accuracy and vibration of bridge. These movements can be decomposed as offset and rotations. Installation accuracy of camera offset could be controlled into 0.1m easily. And offset caused by bridge vibration is usually smaller than 0.2m for the investigated bridge. Thus, compared to the width of ship channel and camera installation height, offset makes limited influence to final identification results. However, because the amplification effects of angles, rotation would generate bigger influence on results.

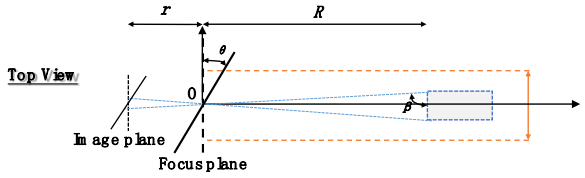


Fig. 17 Illustration for horizontal rotation of image plane

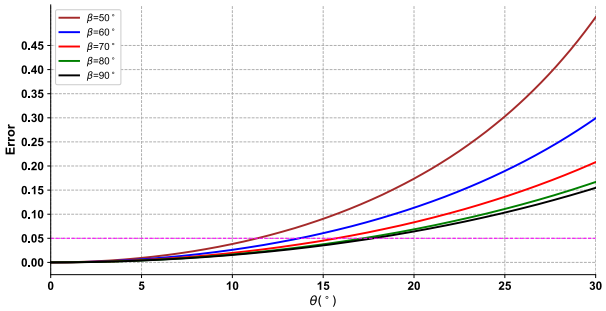


Fig. 18 Curve of error with the change of θ and β

Take horizontal rotation of the camera for example, the vertical rotation of the camera is the same as that of the horizontal rotation analysis. As shown in Fig. 17, a rotation angle $\boldsymbol{\theta}$ was introduced into the calculation model, and $\boldsymbol{\beta}$ is the angle between the line, connected by front edge point of target ship and focus center, and image plane.

The estimated geometric parameters are closely related to the pixel width of ship $\boldsymbol{d_s}$. $\boldsymbol{err}_{\mathrm{ds}}$ is defined to evaluate the influence of rotation. Through simple derivations, relationship between $\boldsymbol{err}_{\mathrm{ds}}$ and $\boldsymbol{\theta}$ could be described as Eq. (7).

$$err_{d_s} = \frac{d_{s\theta}^1 - d_s^1}{d_s^1} = \frac{\sin\beta}{2}\left[\frac{1}{\sin(\beta+\theta)} + \frac{1}{\sin(\beta-\theta)}\right] - 1 = \frac{\sin^2\beta\cos\theta}{\sin^2\beta - \sin^2\theta} - 1 \quad (7)$$

Fig. 18 shows the relationship between $\boldsymbol{err}_{\mathrm{ds}}$ and $\boldsymbol{\theta}$, generated from Eq. (7), where $\boldsymbol{\beta}$ could be calculated with $\boldsymbol{R}$ and $\boldsymbol{W_s}$. In practical situations, $\boldsymbol{\beta}$ usually varies from 70 to 90 degrees at horizontal cases, and equivalent 60 to 80 degrees at vertical cases. It could be seen from Fig. 18, if error needs to be controlled less than 5%, 15-degree rotation is allowed at horizontal case and 10-degree at vertical case at most.

5. Conclusions

In this study, a method for real-time geometric parameter identification of ships based on a transfer-learned, state-of-the-art object detection algorithm (i.e., SSD) using computer vision techniques was proposed and employed for ship–bridge collision avoidance in a bridge area. The following are the conclusions obtained from this study:

- This work employed deep-learning-based algorithm for ship detection based on visible-light images in bridge area. A transfer-learned model based on the SSD network using ship images from Internet was trained with 12,000 iterations. The testing accuracy of ship detection reached 88.13%, higher than the directly trained model using the VOC 2007&2012 dataset.
- A segmentation method that uses saturation channel with morphological operation was employed to obtain accurate contours of the targeted ships. Relationships were established for the estimation of geometric parameters using pinhole cameras. And then, the width, length, height, localization, and velocity of the ships can be acquired.
- Video images, obtained from the installed cameras on the girder of the Wuhan Yangtze River Bridge above the shipping channel, were employed in this study. The effectiveness and accuracy were validated through in situ field testing results.

At the current stage, the proposed approach provides an effective and low-cost technique to analyze the risks of ship collision, which will shed light on the safety assessment and decision-making process for bridge authorities. It should be noted that the object segmentation method, used to obtain the contours of ships, were not sufficiently efficient. And the ship identification process does not combine the recognition results of the previous frame. The Mask R-CNN and optical flow for tracking will be considered to eliminate the dependence on human experience and improve the efficiency of ship tracking in the near future.

Acknowledgments

The research described in this paper was financially supported by National Natural Science Foundation of China (NSFC Grant No. 51678204, 51478149, 51638007) and Guangxi Science Base and Talent Program (Grand No. 710281886032).

References

Andrew, A.M. (2004), "Multiple view geometry in computer vision", *Kybernetes*, **30**(9-10), 1865-1872.

Bentes, C., Velotto, D. and Tings, B. (2017), "Ship classification in TerraSAR-X images with convolutional neural networks", *IEEE J. Oceanic Eng.*, **43**(1), 1-9.

Celik, O., Dong, C.Z. and Catbas, F.N. (2018), "A computer vision approach for the load time history estimation of lively individuals and crowds", *Comput. Struct.*, **200**, 32-52.

Celik, O., Dong, C.Z. and Catbas, F.N. (2019), *Measurement of Human Loads Using Computer Vision*, Springer.

Chen, Z., Li, H., Bao, Y., Li, N. and Jin, Y. (2016), "Identification of spatio-temporal distribution of vehicle loads on long‐span bridges using computer vision technology", *Struct. Control Health Monit.*, **23**(3), 517-534.

Chen, Z.Q. and Hutchinson, T.C. (2010), "Image-based framework for concrete surface crack monitoring and quantification", *Adv. Civil Eng.*, **2010**.

Chi, S. and Caldas, C.H. (2011), "Automated object identification using optical video cameras on construction sites", *Comput.‐Aided Civil Infrastruct. Eng.*, **26**(5), 368-380.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K. and Fei-Fei, L. (2009), "Imagenet: A large-scale hierarchical image database", *Computer Vision and Pattern Recognition, 2009*, Florida, USA, June.

Eum, H., Yoon, C., Lee, H. and Park, M. (2015), "Continuous human action recognition using depth-MHI-HOG and a spotter model", *Sensors*, **15**(3), 5197-5227.

German, S., Jeon, J.S., Zhu, Z., Bearman, C., Brilakis, I., Desroches, R. and Lowes, L. (2013), "Machine Vision-Enhanced Postearthquake Inspection", *J. Comput. Civil Eng.*, **27**(6), 622-634.

Girshick, R. (2015), "Fast R-CNN", *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, Boston, Massachusetts, USA, June.

Girshick, R., Donahue, J., Darrell, T. and Malik, J. (2014), "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, Ohio, USA, June.

Goerlandt, F., Montewka, J., Kuzmin, V. and Kujala, P. (2015), "A risk-informed ship collision alert system: Framework and application", *Safety Sci.*, **77**(1), 182-204.

Gu, C., Lim, J.J., Arbeláez, P. and Malik, J. (2009), "Recognition using regions", *Computer Vision and Pattern Recognition 2009*, Florida, USA, June.

He, K., Gkioxari, G., Dollár, P. and Girshick, R. (2017). "Mask r-cnn", *Computer Vision (ICCV), 2017 IEEE International Conference on Computer Vision*, Venice, Italy, October.

Hoskere, V., Narazaki, Y., Hoang, T. and Spencer Jr., B.F. (2018), "Vision-based structural inspection using multiscale deep convolutional neural networks", *arXiv preprint arXiv:1805.01055*.

Hoskere, V., Narazaki, Y., Hoang, T.A. and Spencer Jr, B.F. (2018), "Towards automated post-earthquake inspections with deep learning-based condition-aware models", *arXiv preprint arXiv:1809.09195*.

Huang, C.L. and Ma, H.N. (2012), "A moving object detection algorithm for vehicle localization", *Proceedings of the 2012 Sixth International Conference on Genetic and Evolutionary Computing*.

Hyukmin, E., Jaeyun, B., Changyong, Y. and Euntai, K. (2015), "Ship detection using edge-based segmentation and histogram of oriented gradient with ship size ratio", *Int. J. Fuzzy Log. Intell. Syst.*, **15**(4), 251-259.

Jahanshahi, M., Kelly, J., Masri, S. and Sukhatme, G. (2009), "A survey and evaluation of promising approaches for automatic image-based defect detection of bridge structures", *Struct. Infrastruct. Eng.*, **5**(6), 455-486.

Kong, X. and Li, J. (2018), "Automated fatigue crack identification through motion tracking in a video stream", *Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems 2018.*

Kulchandani, J.S. and Dangarwala, K.J. (2015), "Moving object detection: Review of recent research trends", *Proceedings of the International Conference on Pervasive Computing*, St. Louis, Missouri, USA, March.

Lecun, Y., Bengio, Y. and Hinton, G. (2015), "Deep learning", *Nature*, **521**(7553), 436.

Lee, B.J., Shin, D.H., Seo, J.W., Jung, J.D. and Lee, J.Y. (2011). "Intelligent bridge inspection using remote controlled robot and image processing technique", *Isarc Proceedings*, Seoul, Korea, June.

Li, S., Zhu, S., Xu, Y.L., Chen, Z.W. and Li, H. (2012), "Long-term condition assessment of suspenders under traffic loads based on structural monitoring system: Application to the Tsing Ma Bridge", *Struct. Control Health Monit.*, **19**(1), 82-101.

Lin, C.W., Hsu, W.K., Chiou, D.J., Chen, C.W. and Chiang, W.L. (2015), "Smart monitoring system with multi-criteria decision using a feature based computer vision technique", *Smart Struct. Syst.*, **15**(6), 1583-1600.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y. and Berg, A.C. (2016), "SSD: Single shot multibox detector", *European Conference on Computer Vision*, Amsterdam, The Netherlands, October.

Liu, Y., Cho, S., Spencer, B.F.J. and Fan, J. (2014), "Automated assessment of cracks on concrete surfaces using adaptive digital image processing", *Smart Struct. Syst.*, **14**(4), 719-741.

Liu, Z., Zhou, F., Bai, X. and Yu, X. (2013), "Automatic detection of ship target and motion direction in visual images", *Int. J. Electronics*, **100**(1), 94-111.

Makantasis, K., Protopapadakis, E., Doulamis, A., Doulamis, N. and Loupos, C. (2015), "Deep convolutional neural networks for efficient vision based tunnel inspection", *Proceedings of the IEEE International Conference on Intelligent Computer Communication and Processing*, Huston, TX, USA, April.

Narazaki, Y., Hoskere, V., Hoang, T.A. and Spencer Jr., B.F. (2018a), "Automated vision-based bridge component extraction using multiscale convolutional neural networks", *arXiv preprint arXiv:1805.06042*.

Narazaki, Y., Hoskere, V., Hoang, T.A. and Spencer Jr,. B.F. (2018b), "Automated bridge component recognition using video data", *arXiv preprint arXiv:1806.06820*.

Oh, J.K., Jang, G., Oh, S., Lee, J.H., Yi, B.J., Moon, Y.S., Lee, J.S. and Choi, Y. (2009), "Bridge inspection robot system with machine vision", *Automat Constr.*, **18**(7), 929-941.

Otsu, N. (1979), "A threshold selection method from gray-level histograms", *IEEE T Syst. Man Cy*, **9**(1), 62-66.

Ou, J. and Li, H. (2010), "Structural health monitoring in mainland China: Review and future trends", *Struct. Health Monit.*, **9**(3), 219-231.

Pan, S.J. and Yang, Q. (2010), "A survey on transfer learning", *IEEE T. Knowledge Data Eng.*, **22**(10), 1345-1359.

Redmon, J., Divvala, S., Girshick, R. and Farhadi, A. (2016), "You only look once: Unified, real-time object detection", *Proceedings of the IEEE conference on computer vision and pattern recognition*, Las Vegas, USA, June.

Ren, S., Girshick, R., Girshick, R. and Sun, J. (2015), "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", *IEEE T. Pattern Anal. Machine Intell.*, **39**(6), 1137-1149.

Rowley, H.A., Baluja, S. and Kanade, T. (1998), "Rotation invariant neural network-based face detection", *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Santa Barbara, CA, USA, June.

Sermanet, P., Kavukcuoglu, K., Chintala, S. and Lecun, Y. (2013), "Pedestrian detection with unsupervised multi-stage feature learning", *Computer Vision and Pattern Recognition 2013*, Portland, Oregon, USA, June.

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

Sinha, S.K. and Fieguth, P.W. (2006), "Automated detection of cracks in buried concrete pipe images", *Automat Constr.*, **15**(1), 58-72.

Steger, C., Ulrich, M. and Wiedemann, C. (2018), *Machine vision algorithms and applications*, John Wiley & Sons.

Stockman, G. and Shapiro, L.G. (2001), *Computer Vision*, Prentice Hall, Upper Saddle River, New Jersey, USA.

Szeliski, R. (2010), *Computer vision: algorithms and applications*, Springer Science & Business Media, Berlin, Germany.

Vaillant, R., Monrocq, C. and Cun, Y.L. (1994), "Original approach for the localisation of objects in images", *Vision, Image and Signal Processing, IEE Proceedings*, **141**(4), 245-250.

Wang, X. (2011), "Ship target detection and tracking in cluttered infrared imagery", *Opt. Eng.*, **50**(5), 057207-057207-057212.

Xu, Y., Bao, Y., Chen, J., Zuo, W. and Li, H. (2018), "Surface fatigue crack identification in steel box girder of bridges by a

deep fusion convolutional neural network based on consumer-grade camera images", *Struct. Health Monit.*, 1475921718764873.

Xu, Y., Li, S., Zhang, D., Jin, Y., Zhang, F., Li, N. and Li, H. (2017), "Identification framework for cracks on a steel structure surface by a restricted Boltzmann machines algorithm based on consumer-grade camera images", *Struct. Control Health Monit.*, **25**(2), e2075.

Yang, Y., Dorn, C., Mancini, T., Talken, Z., Kenyon, G., Farrar, C. and Mascareñas, D. (2017), "Blind identification of full-field vibration modes from video measurements with phase-based video motion magnification", *Mech. Syst. Signal Pr.*, **85**, 567-590.

Yao, Y., Jiang, Z. and Zhao, D. (2017), "Ship detection in optical remote sensing images based on deep convolutional neural networks", *J. Appl. Remote Sens.*. **11**(4), 1.

Ye, X.W., Dong, C.Z. and Liu, T. (2016), "Image-based structural dynamic displacement measurement using different multi-object tracking algorithms", *Smart Struct. Syst*., **17**(6), 935-956.

Ye, X.W., Ni, Y.Q., Wai, T.T., Wong, K.Y., Zhang, X.M. and Xu, F. (2013), "A vision-based system for dynamic displacement measurement of long-span bridges: Algorithm and verification", *Smart Struct. Syst*., **12**(3-4), 363-379.

Yeum, C.M., Dyke, S.J. and Ramirez, J. (2018), "Visual data classification in post-event building reconnaissance", *Eng. Struct*., **155**, 16-24.

Zhu, D., Feng, Y., Chen, Q. and Cai, J. (2010), "Image recognition technology in rotating machinery fault diagnosis based on artificial immune", *Smart Struct. Syst*., **6**(4), 389-403.

BS