RESEARCH ON REGIONAL CLUSTERING AND TWO-STAGE SVM METHOD FOR CONTAINER TRUCK RECOGNITION

Chao Mi*, Jun Wang, Weijian Mi and Youfang Huang

Container Supply Chain Tech. Engineering Research Center
Shanghai Maritime University, China

Zhiwei Zhang

Shanghai Weiguo Port Equipment Company Ltd, China
Shanghai Abian Smart Technology Company Ltd, China
Shanghai Haizhu Smart Technology Company Ltd, China

Yongsheng Yang and Jun Jiang

Institute of Logistics Science & Engineering
Shanghai Maritime University, China

Postolache Octavian

ISCTE-IUL, Inst Telecomunicacoes, Lisbon, Portugal

Abstract. With large-scale, integrated, intelligence for ports, many ports begin to use intelligent detection systems to make their operations more efficient. The container truck recognition and positioning system is also beginning to apply into container quayside to assist the joint operations between quay cranes and container trucks. However, the traditional vehicle detection by using motion region detection cannot recognize the type of moving object, and the traditional pattern recognition method cannot meet the requirements in real-time operation. In order to solve these problems, an algorithm fused by regional clustering and two-stage SVM classifier is proposed in this paper. The method consists of two phases, which are independently executed in two camera systems on quay cranes. In the first stage, a fast motion regional clustering algorithm is used to detect moving image patches as the truck candidate sub-windows. In the second stage, the container trucks will be recognized in these sub-windows by an optimized two-stage SVM classifier. Compared with existing traditional algorithm, experimental results in container terminal show that the fusion algorithm with regional clustering and two-stage SVM has higher efficiency and better truck recognition performance.

1. **Introduction.** In container terminals, more and more intelligent systems are applied to make operations more efficient [9,17]. Usually, the container truck should park themselves to the suitable positions for quay cranes loading and unloading containers. As shown in Figure 1, according to the container handling design, the quay crane stop itself facing to the bay of the ship that will be loaded. During the whole period to load the same bay, the quay crane never move itself again. Therefore,

FIGURE 1. A quay crane was unloading container from a truck

the container trucks should park themselves at suitable positions underneath of the hoists for quay crane to load or unload containers.

In the past, the drivers parked the container trucks to the correct positions with the guiding by workers at quaysides. Those guiding operations may cost much time and have bad effect on handling performance. Therefore, detection and positioning of quayside container trucks has become one of the hot engineering problems in the port logistics. In recent years, the researchers and engineers try to seek and develop a suitable and real-time method to detect and position container trucks precisely and quickly [16].

The vehicle recognition and position is always a hot research area. An increasing number of researchers and engineers try to different ways depending upon different requirements to detect and position vehicles.

Qifan Wei and Hongmei Zhu et al used magnetic sensors to detect and track vehicles in the roads [22, 28]. This kind of methods show good stability and recall factor. However, only using magnetic sensors cannot differentiate vehicle types. In the quayside, not only container trucks but also other vehicles such as bus and pickup trucks pass through the quay cranes. Therefore, these methods may have some detection errors for that the other vehicles may be recognized falsely as container trucks. Radars are also a choice to apply to port engineering. For example, Mi C et al. applied the radar into ship identification for automatic ship Loader [11]. B Li et al. propose the vehicle detection method based on the 3D Lidar by convolutional network [10]. Fernando García and Xiao Wang et al set out methods that fused laser radars, millimeter-wave radars, GPS and vision systems to detect and position vehicles [3, 21]. This kind of methods show very effective on vehicle detection and positioning, in addition, they can also recognize a part of vehicle types. However, these methods are complex and cost much, and port end users cannot accept highly uneconomical and complex solutions.

Recently, with the development of the computing hardware, the vision systems begin to be a good selection for port engineering. For example, Mi C et al. designed a human detection system and a container corner casting recognition system for security in ports [12,13]. In traffic engineering, the vision systems are also a good selection for vehicle detection and positioning. Jemej Mrovlje developed a method with stereoscopy to cluster the large elements to be detected and remove the small objects for truck detection in the port [14]. But this method cannot recognize the concrete type of vehicle. Morteza Jalalat mixed several algorithms to detect, position and estimate displacements and speed of vehicle [5]. Mixed algorithms may be accuracy but sometimes very slow because several algorithms are combined in the same system. Xue Yuan, Lars Wilko Sommer, Shiva Kamkar and Yunsheng Zhang et al all proposed the methods through background detection and motion regions tracking to achieve the vehicle detection, recognition and positioning [4,7, 15,19,25,26]. These methods are very fast and can track any motion regions in the images. Nevertheless, the vehicle type recognition may be not accurate in these methods because of the very limited information of the vehicles. In order to be able to accurate classify vehicle types, pattern recognition methods were proposed by many researchers and engineers. Y Cai proposed a monocular and binocular vision method based on Deep Convolutional Neural Networks to search vehicle for the whole area of image [1,2,18,27]. Jie Yuan, Thanida Tangkocharoen, Zhaojin Zhang, Q Jiang and Hulin Kuang et al used multi-layer classifiers, deep learning classifiers or neural networks to recognize the vehicles [6,8,20,23,24]. These methods used sliding and multiscale windows to scan the whole image and find the vehicle targets. The way scanning the whole area of image leads to sufficient precision but also be slow in these methods. Therefore, the real-time computing performance cannot be satisfied.

In conclusion, as the future of truck recognition in port, the vision system in vehicle recognition mainly includes two kinds of algorithms: motion region detection and pattern recognition. However, the traditional vehicle detection by using motion region detection cannot recognize the concrete type of moving object, and the traditional pattern recognition method cannot meet the requirements in real-time operation in port. Thus, a fast truck recognition method combined with regional clustering and two-stage classifier is proposed in this paper. The method consists of two phases, which are independently executed in two camera systems on quay cranes. In the pre-search stage, the image of the whole lane can be collected through the front camera on the quay crane. Then a fast motion regional clustering algorithm can be used to detect moving objects as truck candidate sub-windows. After that, the whole image can be divided into several sub-windows as the truck detection candidate windows. In the second stage, if there are any candidate windows found, another camera will begin to recognize container trucks in those corresponding window images through an optimized two-stage SVM classifier. Finally, the multi-scale recognition results can be fused to obtain the precise center position of container truck. Compared the other methods, the experimental results show that our method can detect, position and recognize the container trucks more efficient.

2. **Fast detection of truck position.**

2.1. **Camera installation and framework of algorithm.** Referring to camera installation above each lane, there are two cameras installed on the opposite beams of the quay crane. As shown in Figure 2, the pre-search camera and truck detection
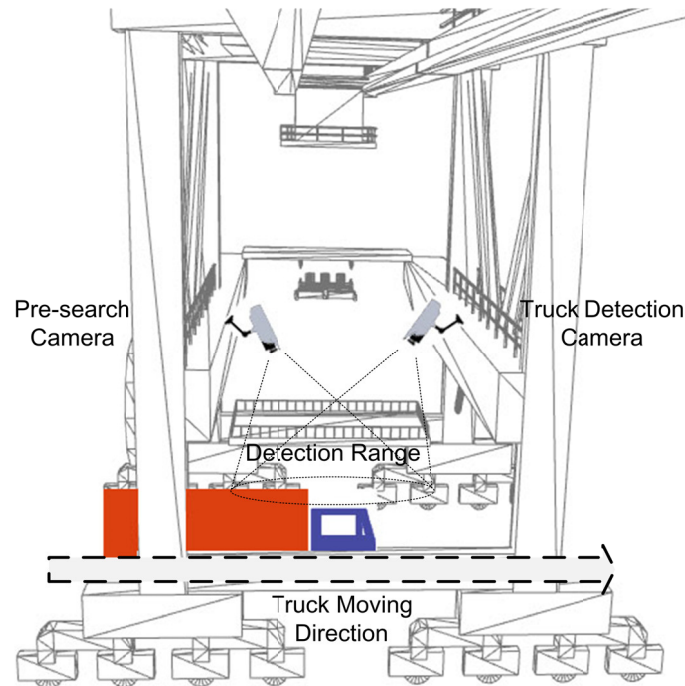
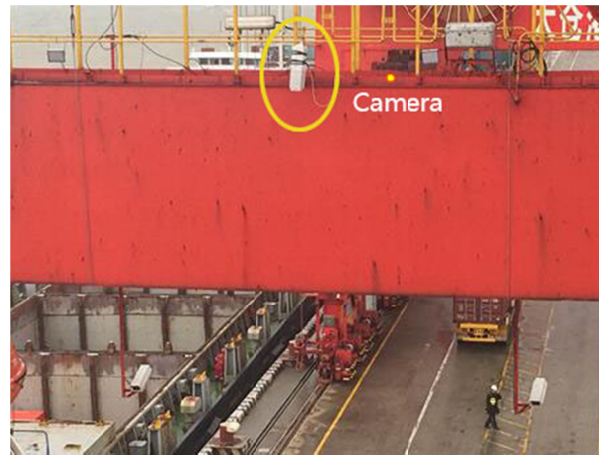FIGURE 2. Cameras' positions on the quay crane

FIGURE 3. The field installation of the camera

camera monitor the overlapping area. The detection of the two cameras can all range from entrance to exit of truck in the corresponding lane. Only the pre-search camera can keep working all day normally, while the truck detection one is always in a standby sleep mode to save computing resources until the pre-search camera send active messages to truck detection camera. Figure 3 is the field installation of the camera.
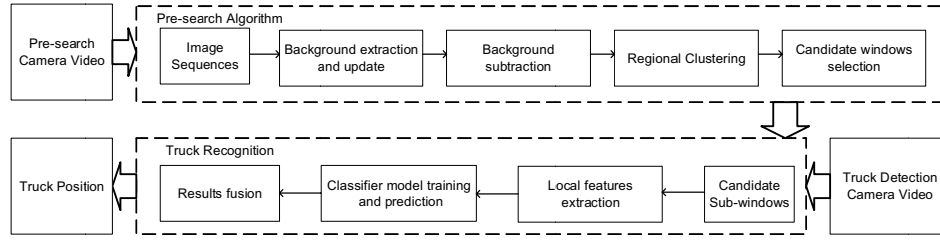
FIGURE 4. Truck detection algorithm flowchart

When the truck is moving across the lane, the pre-search camera will scan the image quickly to find candidate sub-windows. If the moving objects is detected in video sequences, the pre-search camera will send the information of candidate windows with the moving objects. After receiving the information, the truck detection camera will change the mode from sleep to active. Then the truck position will be detected accurately in the corresponding image sub-windows from the truck detection camera. The detailed process of truck detection is as follows:

Our method is expressed by two components: fast truck pre-search and detection of truck position, and the considered system includes two cameras. As shown in Figure 4, the purpose of pre-search camera is to select fewer candidate sub-windows. At first, the background image can be generated by Gaussian mixture modeling from the image sequences. Then the image patches of moving objects will be extracted through background subtraction method. Next, these image patches are clustered by k-means clustering and get the outer rectangular frames of clusters. Through filtering these rectangular frames, the candidate sub-windows can be selected based on some parameters conditions. After that, the process of truck detection camera will be activated. The next steps are all processed in the corresponding candidate windows. The HOG feature will be extracted to train the two-stage SVM classifiers model. Then the truck with two kinds of shapes can be detected in the candidate slide windows with the two-stage SVM classifiers. Finally, the prediction results of the two-stage SVM classifiers will be fused and the fusion result is the final coordinate of truck position.

2.2. **Fast truck pre-search algorithm.** Generally, the shape of truck head is uniformed in the same terminal, but the shape of truck trailer has great difference between empty or equipped with a container. So the truck detection also means the truck head detection.

In the actual situation of positioning the truck, the traditional method of truck detection is to scan the whole image to search the target truck, which will cost too much time and hardware resources. Thus, the fast truck pre-search algorithm is proposed in this paper to find out the possible position regions of truck in the original image, which can save a lot of time for two reasons:

- The pre-search algorithm can find out a truck position range roughly to avoid scan the whole image.
- The pre-search algorithm can pick out the adaptive classifier according to the per-search results, which can avoid the resources waste of the simultaneous detection for the two-stage classifiers.
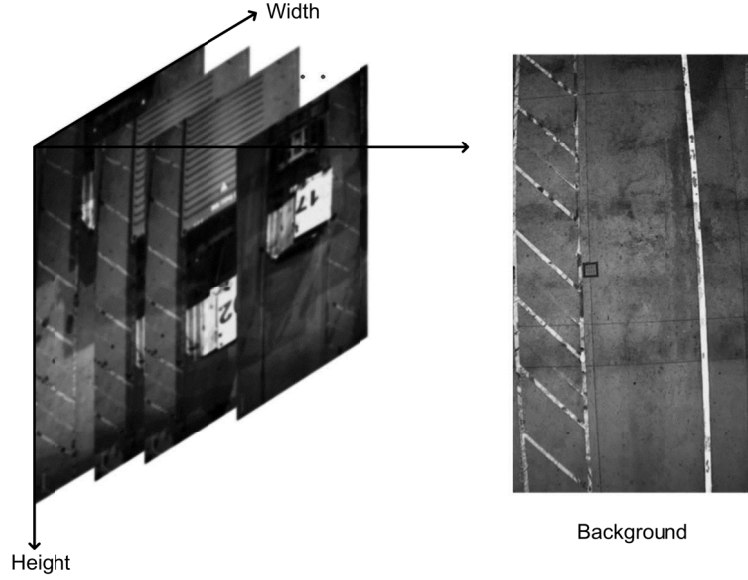
FIGURE 5. Schematic illustration of background extraction and update

Therefore, the background subtraction method is applied as the fast truck pre-search method in this paper. This method compares the real-time monitoring and background to find out the possible truck region. The detail process of fast truck pre-search algorithm is as follows:

1. Background extraction and update

First of all, the original RGB image should be converted to grayscale images. Then the background can be extracted by Gaussian mixture modeling method as shown in Figure 5. In order to adapt the multimodal background scene, the Gaussian mixture model is built by using the weighted sum of K Gaussian distributions. Thus, it can deal with the complex background scenes, which can be described by Eq. 1.

$$P\left(X_t\right) = \sum_{i=1}^{K} w_{i,t} \cdot \eta\left(X_t, \mu_{i,t}, \delta_{i,t}^2\right), \tag{1}$$

Where, means the weight of Gaussian distribution at time t and K is the number of distributions. $\eta\left(X_t, \mu_{i,t}, \delta_{i,t}^2\right), \mu_{i,t}, \delta_{i,t}^2$ is the probability density function, the mean value and variance, and $\eta\left(X_t, \mu_{i,t}, \delta_{i,t}^2\right)$ can be calculated by Eq. 2.

$$\eta\left(X_t, \mu_{i,t}, \delta_{i,t}^2\right) = \frac{1}{\sqrt{2\pi}\delta_{i,t}} e^{-\frac{1}{2}(X_t-\mu_{i,t})^T \delta_{i,t}^{-2}(X_t-\mu_{i,t})}, \tag{2}$$

2. Background subtraction

Through the subtraction of background frame and the current frame, if the pixel value alteration is very small, there is no moving object. Otherwise, there is a truck possibly in the corresponding region. The method can be

FIGURE 6. The result of background subtraction

described by Eq. 3.

$$D\left(x,y\right) = \begin{cases} 1, & \text{if } |f_b(x,y) - f_k(x,y)| > T \\ 0, & \text{others} \end{cases}, \tag{3}$$

Where, $D(x,y)$ is the pixel gray value of target frame, $f_k(x,y)$ and $f_b(x,y)$ means the pixel values of point (x,y) at time k frame and background frame, T represents the threshold value.

As shown in Figure 6, the result of background subtraction contains many small image patches, even not belong to the target truck. Therefore, the result of this step need to be processed further.

3. Regional clustering

For large image data, the traditional clustering algorithm cannot be effectively dealt with, which is due to the traversal operators at all pixel points of the image. Different from the traditional clustering, the regional clustering algorithm is proposed in this paper. First, the white points of the same connected region are considered as one element. Each element contains a series parameters of attribute such as center point coordinate, the connected region area and the radius of outer circle. Then, the large amount points of image data can be mapped to low-dimensional space. In the lower-dimensional space, these image patches can be clustered into several categories based on k-means algorithm. Utilized the method of Euclidean space weighted distance, the regional clustering is evaluated by criterion function of minimizing the error sum of the squares of results, which can be calculated by Eq. 4.

$$J = \sum_{i=1}^{k} \sum_{x \in X_i} \|x - m_i\|^2, \tag{4}$$

Where, k is the number of categories, X_i is the collection of image patches nearby the cluster center m_i of cluster i.

As shown in Figure 7, these image patches are clustered into two categories: truck and non-truck objects. For each clustering result, the minimal outer
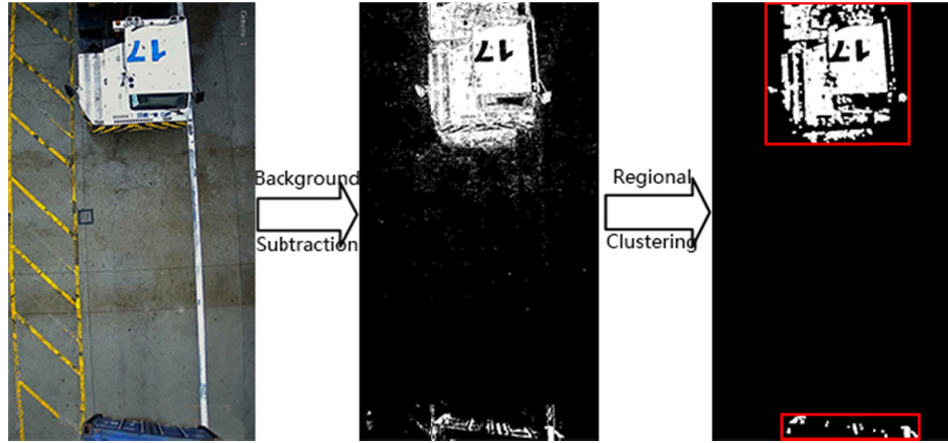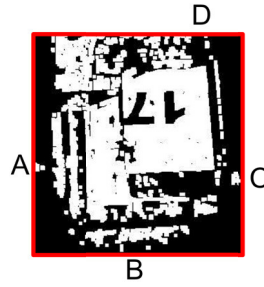
FIGURE 7. Regional clustering of image patches

FIGURE 8. Outer rectangle of the cluster

rectangle can be drawn by Eq. 5.

$$\begin{cases} O(x,y) = \left(\frac{x_C + x_A}{2}, \frac{y_B + y_D}{2} \right) \\ \qquad W_O = x_C - x_A \\ \qquad H_O = y_B - y_D \end{cases}, \qquad (5)$$

In the Figure 8, the point A, B, C, D are the leftmost, lowest, right most and top-most points, which can determine the boundary of outer rectangle. $O(x,y), W_o, H_o$ represent the center coordinates, width and height of the rectangle. x, y means the coordinate of each point.

4. Candidate windows selection

After image patches clustering, some non-truck objects might also appear in the truck images, such as operators, cars or other moving objects. Whether these rectangular frames are reserved as candidate sub-windows with the selection criteria including the size of area, width, height and angle. However, these selection criteria only can find the candidate sub-windows approximately without a sufficient accuracy. Therefore, the next stage will detect truck position more accurately in these candidate sub-windows.

2.3. **Detection algorithm of truck position.** After the process of pre-search camera, the detection of truck position will be completed by the truck detection

camera, as shown in Fig. 1. According to the candidate sub-windows in the previous step, the corresponding regions in the truck detection camera will be processed by sliding windows. However, the two cameras are installed at opposite positions what means monitor the overlapping area. The images from different cameras are symmetrical with each other. Thus, the image coordinates of truck detection camera need to be converted to keep consistent with the other one before the next steps.

Finding the corresponding candidate sub-windows approximately, the truck detection camera will detect the truck position by three steps: Local feature extraction, Classifier model training and results fusion. Firstly, the HOG features will be extracted in this paper, and then the SVM classifier model will be trained and the truck position can be predicted. Finally, the results from each slide windows will be fusion as the coordinate of truck position.

2.3.1. *Local features extraction.* As shown in Figure 6, feature extraction can be divided into five steps: gamma correction operation, the gradient calculation, HOG (Histogram of Oriented Gradient) descriptor calculation, feature normalization and feature vector combination.

1. Image noise is easy to appear in truck image because of the poor environment in container terminal, which can be suppressed by a square root compression gamma correction operation. The standardized pixel values of three channels, $r'_{x,y}, g'_{x,y}$, and $b'_{x,y}$ can be calculated using Eq. 6:

$$\begin{aligned} r'_{x,y} &= \sqrt{r_{x,y}} \\ g'_{x,y} &= \sqrt{g_{x,y}} \\ b'_{x,y} &= \sqrt{b_{x,y}} \end{aligned} \tag{6}$$

   Where, $r'_{x,y}, g'_{x,y}$, and $b'_{x,y}$ are the original pixel values of the red, green, and blue channels, respectively, and the subscript x, y represents the pixel coordinates.

   The gradient magnitude and gradient direction can be calculated by a one dimensional discrete differential template according to Eq. 7 and Eq. 8

$$\nabla f(x, y) = \sqrt{G_x^2(x, y) + G_y^2(x, y)}, \tag{7}$$

$$\theta(x, y) = \tan^{-1}(G_y(x, y) / G_x(x, y)) \tag{8}$$

   Where, $\nabla f(x, y)$ and $\theta f(x, y)$ are the gradient magnitude and gradient direction of points (x,y), respectively. $G_x(x, y), G_y(x, y)$ indicate the component of the gradient magnitudes of pixel points in the x and y axis.

2. According to Figure 9, the whole image is scanned by a pixels sliding block, which is divided into four pixels cells. The gradient direction of each point in a cell is discretized into 9 bins in order to extract the cell HOG feature vector using a tri-linear interpolation method. A 36-dimensional HOG feature vector can be extracted in each block.

3. The HOG feature of each block needs to be normalized by L2-Norm function to suppress the influence of the background illumination and edge

4. Finally, all the entire 63 blocks of the whole image are connected to form a 2268-dimensional feature vector. After the steps above are complete, the truck HOG feature is obtained as shown in Figure 10.

However, in actual image, the shape feature will be very different when the truck is located in the upper or lower half of the image. Thus, two sets of sample data

are required to be trained for different shapes of trucks as it is presented in Figure 10.

2.3.2. *Classifier model: Training and prediction.* In this work, we use the SVM classifier to deal with truck position. For truck position detection, the size of sample image is 80×64 pixels with multi-dimensional feature space. Comparing to the traditional single SVM classifier, the different position of truck needs two-stage SVM classifiers proposed for truck detection. As shown in Figure 11, the two-stage SVM classifiers need to find two hyper-planes H and H′) to classify positive and negative samples correctly. The distance between plane $H_1(H_1')$ and plane $H_2(H_2')$ is called the 1maximum margin, and the features of the sample points on planes $H_1(H_1')$ and $H_2(H_2')$ are support vectors. The equation of the hyper-plane can be formulated with Eq. 9.

$$w^T x + b = 0, \tag{9}$$

During the preprocess stage, the two-stage SVM classification algorithm firstly divides the truck samples into three classes: the truck in the lower half of the image, the truck in the upper half of the image, non-truck as shown in Figure 12. Firstly, each sample selected from the truck images need to be made a binary decision: the upper or lower half of the image, as shown in **Figure 13**. Then the two-stage classifiers can make the samples divided into truck or non-truck respectively in two cases. Finally, the possible truck area leads the appropriate classifier to detect the truck position with corresponding parameters.

As a supervised algorithm, SVM need to make the sample sets marked with corresponding labels. The training steps of classifiers are following presented:

1. Preprocessing

    According to the images captured in the truck lane, the truck is kept by a window size of 80x64 pixels, as positive samples. Of course, the upper half area truck and the lower truck will be the positive samples of the two-stage
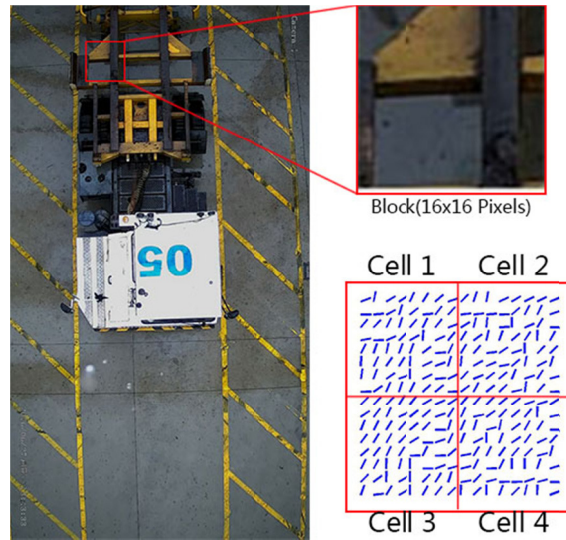

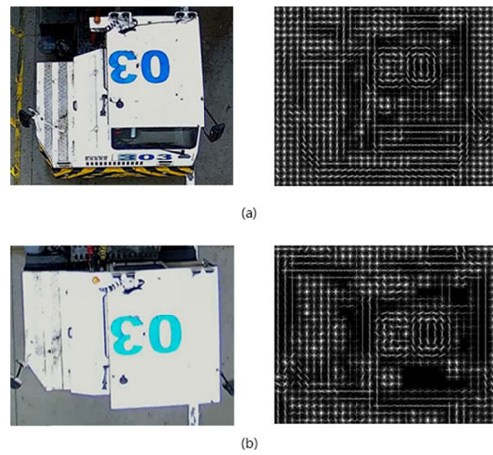FIGURE 9. Schematic illustration of the HOG descriptor

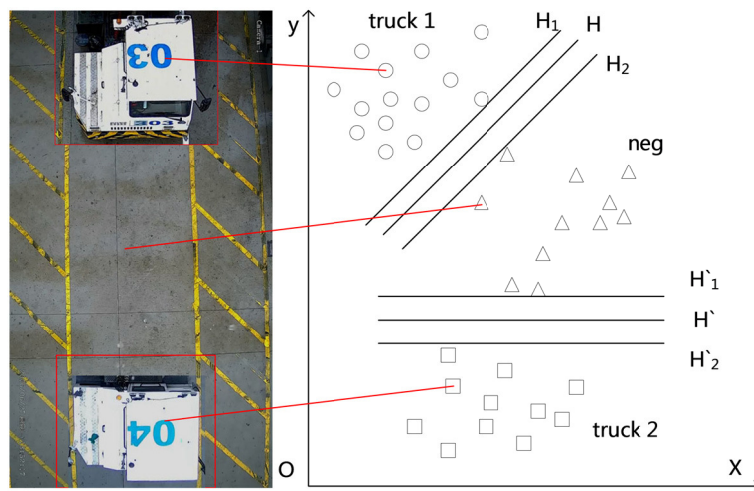FIGURE 10. The HOG features of trucks in different position

FIGURE 11. Schematic illustration of two-stage SVM classifier
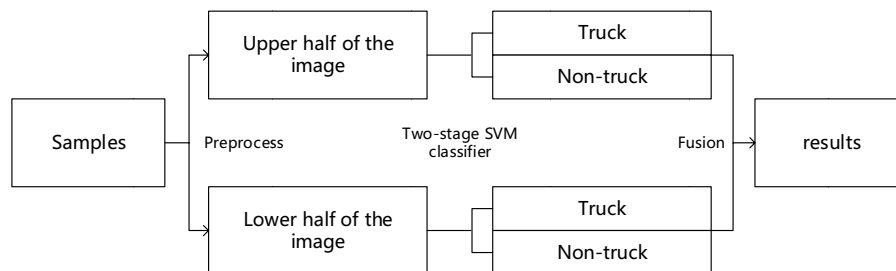

FIGURE 12. Two-stage SVM algorithm flowchart

classifiers respectively. The positive samples are shown in **Figure 13**. Besides

FIGURE 13. The sample sets of two kinds of truck shape features

the truck, the other part of the images can be cut as negative samples of two-stage classifiers. Then the HOG feature extracted from positive samples will be labeled as, and the negative one will be labeled as.

2. Initial SVM classifier training

After the HOG feature are extracted, the 2268-dimensional features will be mapped to a higher-dimensional feature space by using Radial Basic Function (RBF), as shown in Eq. 10.

$$K(x, x_c) = e^{\left(-||x-x_c||^2/\left(2\sigma^2\right)\right)} \tag{10}$$

Here, x is a feature vector and x_c is the position of the center of kernel function. The width parameter σ can control the radial scope of impact from the kernel function.

After that, the HOG feature space of samples can be linearly separated. Through objective function and constraint function, the two hyper-planes H and H′ of initial SVM classifier can be obtained by Eq. 11.

$$\begin{cases} \min \frac{1}{2}||w||^2 + c\sum_{i=1}^{l} \zeta_i, \min \frac{1}{2}||w'||^2 + c\sum_{i=1}^{l'} \zeta_i \\ \text{s.t?} y_i \cdot \left(w^T x_i + b\right) - 1 \geq 0 \, (i = 1, 2, \cdots, l) \\ y'_i \cdot \left(w'^T x'_i + b'\right) - 1 \geq 0 \, (i = 1, 2, \cdots, l') \end{cases}, \tag{11}$$

Where, c and ζ_i represent the punishment factor and slack variables, respectively, which can increase the fault tolerance of the SVM classifier. l(l') is the number of samples, and $y_i(y'_i)$ indicates the label value of $x_i(x'_i)$. w(w') represents a combination of feature vectors, which can be calculated by Eq. 12:

$$w = \alpha_1 x_1 y_1 + \alpha_2 x_2 y_2 + \cdots + \alpha_n x_n y_n \tag{12}$$

Where, α and n is Lagrange multipliers and the size of sample sets.

3. Final SVM classifier training.

Sometimes, it is hard to get correct results when testing some negative samples by the initial classifier. Thus, these examples are generally called hard examples. And the HOG features of hard examples will be extracted to combine with the initial HOG features as new negative samples. Repeating the step 3, the final two-stage SVM classifiers with appropriate detection accuracy will be trained.

2.3.3. *Fusion results.* The final purpose of truck detection is to find the position of the target truck in the whole image. Thus, all of the candidate windows need to be scanned by a slide window in a multi-scale way. During this processing, the same truck will be detected several times at different scale and position. Fusion results

FIGURE 14. Results fusion

means restoring the size and accuracy positions under the slide window into the original image and clustering these positions. According to all the position results, the mean-shift clustering method will be used to fusion the coordinates of truck position.

In n-dimensional space R^n, there are n sample points x_i, mean-shift vector $M_h(x)$ can be calculated by Eq. 13.

$$M_h\left(x\right) = \frac{1}{k} \sum_{x_i \in S_h} \left(x_i - x\right) \tag{13}$$

Where, the S_h is a high dimensional ball area with radius h, which can be defined by Eq. 14.

$$S_h\left(x\right) = \left(y \middle| \left(y - x\right)\left(y - x\right)^T \le h^2\right) \tag{14}$$

The green rectangular frames in **Figure 14** is the discrete results of the same truck position. After averaging these positions, the coordinate of the target truck will be calculated as an accurate truck position.

In conclusion, the fast pre-search method operates with fewer of the candidate windows and reduce the search region to reduce the detection times. On the other hand, the two-stage SVM classifiers are trained for two different truck shapes with a higher accuracy. Combination of fast pre-search method and the two-stage SVM classifiers will meet the demand of real-time and detection accuracy.

3. **Results and discussion.** The fast truck recognition approach has been integrated into a system, which has been tested in Taicang Port. The screenshot of experimental results are as shown in **Figure 15**.

We evaluate our detection approach on the truck dataset collected from Taicang Port. This dataset includes 500 pictures of two types positive samples and 3000 negative samples in training sets, 500 daytime pictures and 500 nighttime pictures (540*960 pixels) in testing set, which contains 981 container trucks. All of these sample pictures are from the complicated background during the handling process in container terminal.
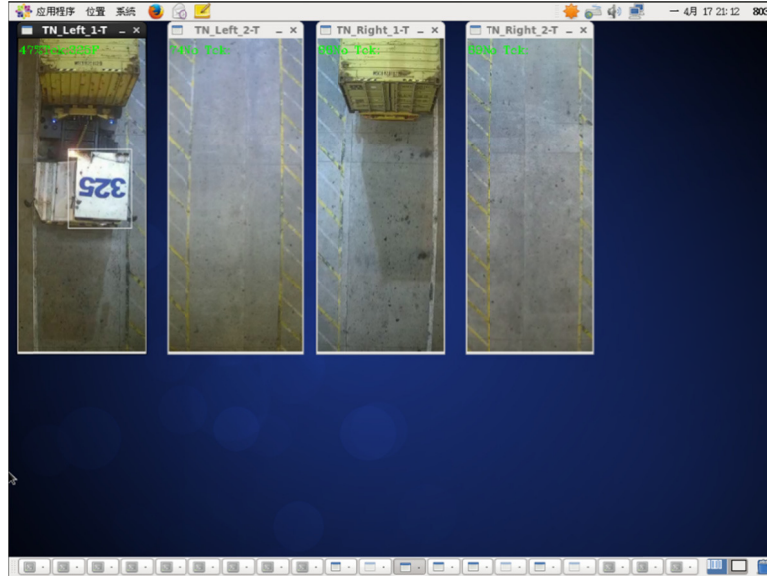
FIGURE 15. The experimental results of truck detection

TABLE 1. Experimental results

Algorithm time		Number	The fast truck detection algorithm				Traditional SVM			
			TRUE	FALSE	Accuracy	APT (ms)	TRUE	FALSE	Accuracy	APT (ms)
Test Set 1	day	50	50	0	100%	185.4	50	0	100%	423.6
Test Set 2	night	50	48	2	96%	135.4	46	2	92%	392.3
Test Set 3	day	50	50	0	100%	133.1	48	5	96%	404.8
Test Set 4	night	50	47	3	94%	107.4	47	1	94%	384.3
Test Set 5	day	50	49	1	98%	236.5	48	6	96%	419.9
Test Set 6	night	50	50	0	100%	201.0	46	2	92%	424.9
Test Set 7	day	50	50	0	100%	125.4	49	6	98%	381.8
Test Set 8	night	50	49	1	98%	103.1	45	4	90%	406.8
Test Set 9	day	50	50	0	100%	124.2	50	1	100%	359.3
Test Set 10	night	50	50	0	100%	105.4	48	3	96%	380.6
Test Set 11	day	50	50	0	100%	113.2	48	30	96%	365.8
Test Set 12	night	50	47	3	94%	245.1	46	4	92%	401.3
Test Set 13	day	50	49	1	98%	135.2	50	0	100%	502.8
Test Set 14	night	50	48	2	96%	116.3	45	5	90%	396.7
Test Set 15	day	50	50	0	100%	127.8	50	0	100%	456.5
Test Set 16	night	50	50	0	100%	156.5	46	4	92%	379.8
Test Set 17	day	50	50	0	100%	195.2	49	1	98%	427.8
Test Set 18	night	50	48	2	96%	188.3	42	8	84%	511.3
Test Set 19	day	50	50	0	100%	152.3	47	3	94%	386.5
Test Set 20	night	50	46	4	92%	164.5	44	6	88%	462.4
sum		1000	981	19	98%	152.565	944	91	94%	413.46

After building the testing set, 1000 images were randomly selected and distributed to 20 sample test sets equally. The test computer with CentOS operation system presents the following characteristics 8GB memory and Intel(R) Core i7-3770 @3.40GHz, CPU, and the experimental results are shown in Table 1.

According to **Table 1**, the performance of our proposed fast truck detection approach and the traditional SVM in the daytime is both better than that in the nighttime. The reason is that the lighting condition at night is more complex and
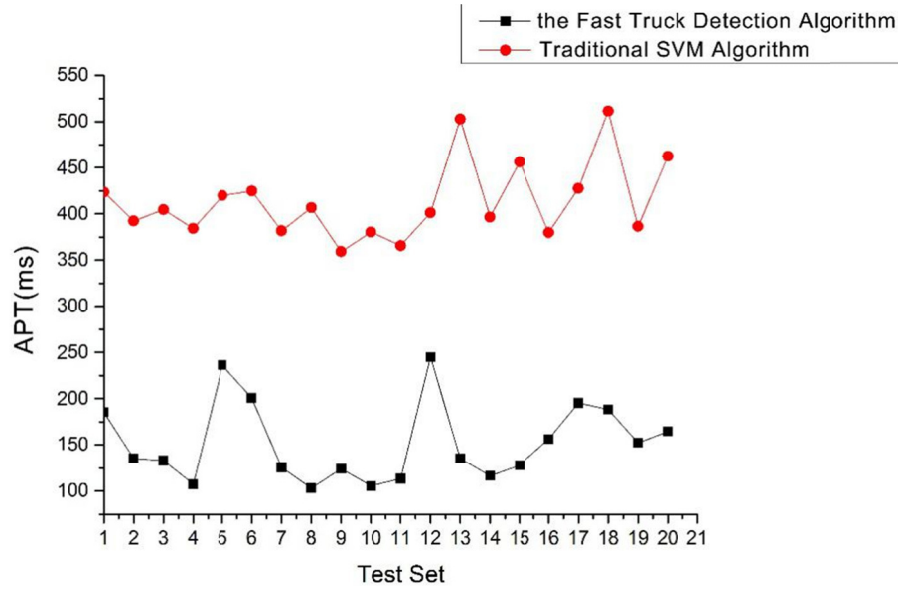
FIGURE 16. Comparison of two algorithms about average processing time

worse than the day. In fact, the overall detection rate of our fast detection approach is 98% in **Table 1**, which can satisfy the handling request, while the traditional approach reached only 94%. It means that the training way of the two-stage SVM classifier separately can achieve a better classifier, and the traditional way with the mixed types samples is difficult to obtain a high-performance universal classifier.

As shown in **Figure 16**, the traditional algorithm always has more average processing time (APT) than our fast detection algorithm: the traditional one takes about 400ms per image while the fast detection one need only 152ms. Because the fast truck pre-search algorithm can reduce a lot scanning area for the two-stage SVM classifier and the traditional algorithm always needs to scan the whole image. However, the variation of light at night brings more search area. As a result, the truck detection will take more time at night than the day.

In conclusion, this paper proposed a novel container truck detection approach which is made up of a fast truck pre-search algorithm and a two-stage SVM classifier for different types of truck head. It can improve the detection accuracy to 98%, which has a great progress compared to traditional SVM classifier and can also satisfy the real-time request.

4. **Conclusions.** According to the existing problems of traditional vision algorithm, a novel fast truck recognition method base on fast regional clustering algorithm and the two-stage classifier is proposed in this paper. Firstly, the truck region will be pre-searched quickly by regional clustering as the candidate sub-windows. Then, the truck position can be recognized by the two-stage SVM accurately in the corresponding sub-windows. The experimental tests in container terminal of Taicang Port show that not only the detection accuracy of the fast truck recognition approach can reach 98%, but also costs less time than the traditional SVM and satisfy the real-time request. In conclusion, the improved classification approach

proposed in this paper has higher efficiency and better truck recognition performance.

Conflicts of Interest. The authors declare no conflict of interest. The founding sponsors Mi Chao and Yang Yongsheng decided to publish this results.

REFERENCES

[1] Y. Cai, X. Chen and H. Wang, et al., Deep representation and stereo vision based vehicle detection, *IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems IEEE*, (2015), 305–310.

[2] Y. Cao, G. I. Kirilova and M. L. Grunis, Cooperative research projects of master's students (education programs) in the open informational educational environment, *Eurasia Journal of Mathematics Science and Technology Education*, **13** (2017), 2859.

[3] F. Garcia, D. Martin and A. D. L. Escalera, et al., Sensor Fusion Methodology for Vehicle Detection. *IEEE Intelligent Transportation Systems Magazine*, **9** (2017), 123–133.

[4] I. Iszaidy, A. Alias and R. Ngadiran, et al., Video size comparison for embedded vehicle speed detection & travel time estimation system by using Raspberry Pi, *Robotics, Automation and Sciences (ICORAS),* (2016), 1–4.

[5] M. Jalalat, M. Nejati and A. Majidi, Vehicle detection and speed estimation using cascade classifier and sub-pixel stereo matching, *Signal Processing and Intelligent Systems (ICSPIS),* (2016), 1–5.

[6] Q. Jiang, L. Cao and M. Cheng, et al., Deep neural networks-based vehicle detection in satellite images, *International Symposium on Bioelectronics and Bioinformatics*, (2015), 184–187.

[7] S. Kamkar and R. Safabakhsh, Vehicle detection, counting and classification in various conditions, *IET Intelligent Transport Systems,* **10** (2016), 406–413.

[8] H. Kuang, X. Zhang and Y. J. Li, et al., Nighttime vehicle detection based on bio-inspired image enhancement and weighted score-level feature fusion, *IEEE Transactions on Intelligent Transportation Systems*, **PP** (2017), 1–10.

[9] W. Li, W. Liu and X. Xu, et al., The port service ecosystem research based on the lotka-volterra model[J], *Polish Maritime Research,* **24** (2017), 86–94.

[10] B. Li, T. Zhang and T. Xia, Vehicle Detection From 3D Lidar Using Fully Convolutional Network, arXiv preprint, 2016, arXiv:1608.07916.

[11] C. Mi, Y. Shen and W. Mi, et al., Ship identification algorithm based on 3d point cloud for automated ship loaders, *Journal of Coastal Research*, **73** (2015), 28–34.

[12] C. Mi, X. He and H. Liu, et al., Research on a fast human-detection algorithm for unmanned surveillance area in bulk ports, *Mathematical Problems in Engineering*, **2014** (2014), 1–17.

[13] C. Mi, Z. Zhang and Y. Huang, et al., a fast automated vision system for container corner casting recognition, *Journal of Marine Science & Technology*, **24** (2016), 54–60.

[14] J. Mrovlje and D. Vran, Automatic detection of the truck position using stereoscopy, *IEEE International Conference on Industrial Technology*, (2012), 755–759.

[15] S A. Nur, M. M. Ibrahim and N. M. Ali, et al., Vehicle detection based on underneath vehicle shadow using edge features, *Control System, Computing and Engineering (ICCSCE),* (2016), 407–412.

[16] Y. Shen, W. Mi and Z. Zhang, A positioning lockholes of container corner castings method based on image recognition, *Polish Maritime Research,* **24** (2017), 95–101.

[17] Y. Shen, N. Zhao and M. Xia, et al., A deep q-learning network for ship stowage planning problem, *Polish Maritime Research*, **24** (2017), 102–109.

[18] L. Si and H. Qiao, Performance of Financial Expenditure in China's basic science and math education: Panel data analysis based on CCR model and BBC model, *EURASIA Journal of Mathematics Science and Technology Education*, **13** (2017), 5217–5224.

[19] L. W. Sommer, T. Schuchert and J. Beyerer, A comprehensive study on object proposals methods for vehicle detection in aerial images, *Pattern Recogniton in Remote Sensing (PRRS),* (2016), 1–6.

[20] T. Tangkocharoen and A. Srisuphab, Vehicle detection on a pint-sized computer, *Knowledge and Smart Technology (KST)*, (2017), 40–44.

[21] X. Wang, L. Xu and H. Sun, et al., On-road vehicle detection and tracking using mmw radar and monovision fusion, *IEEE Transactions on Intelligent Transportation Systems,* **17** (2016), 1–10.

[22] Q. Wei and B. Yang, Adaptable vehicle detection and speed estimation for changeable urban traffic with anisotropic magnetoresistive sensors, *IEEE Sensors Journal*, **17** (2017), 2021–2028.

[23] J. Yuan, W. Cheng and L. Sun, et al., A constructing vehicle intrusion detection algorithm based on BOW presentation model, *Signal and Image Processing (ICSIP)*, (2016), 183–188.

[24] Z. Zhang, C. Xu and W. Feng, Road vehicle detection and classification based on Deep Neural Network, *Software Engineering and Service Science (ICSESS)*, (2016), 675–678.

[25] Y. Zhang, C. Zhao and J. He, et al., Vehicles detection in complex urban traffic scenes using Gaussian mixture model with confidence measurement, *IET Intelligent Transport Systems,* **10** (2016), 445–452.

[26] Z. Zhang, X. Yu and F. You, et al., A front vehicle detection algorithm for intelligent vehicle based on improved gabor filter and SVM, *Recent Patents on Computer Science*, **8** (2015), 32–40.

[27] Y. Zhang, W. Chen and H. Liu, Design for the competitiveness assessment index system of aviation aerospace manufacture industry on basis of GEMPearsonVC combination method, *Journal of Interdisciplinary Mathematics,* **20** (2017), 255–268.

[28] H. Zhu and F. Yu, A cross-correlation technique for vehicle detections in wireless magnetic sensor network, *IEEE Sensors Journal,* **16** (2016), 1–1.

E-mail address: chaomi@shmtu.edu.cn
E-mail address: smu_wangjun@126.com
E-mail address: mwj@shmtu.edu.cn
E-mail address: yhuang@shmtu.edu.cn
E-mail address: zhangzhiwei@abiantech.com
E-mail address: yangys@shmtu.edu.cn
E-mail address: jiangjun@abiantech.com
E-mail address: octavian.postolache@gmail.com