Journal of Coastal Research	SI	103	822–829	Coconut Creek, Florida	2020

A Port Container Code Recognition Algorithm under Natural Conditions

Chao Mi[†*], Lingen Cao[†], Zhiwei Zhang[‡], Yufei Feng[‡], Lei Yao[‡], and Yunbao Wu[‡]

[†]Logistics Engineering School
Shanghai Maritime University
Shanghai 201306, China

[‡]Shanghai SMU Vision Smart Technology Ltd.
Shanghai 201306, China

www.cerf-jcr.org

www.JCRonline.org

ABSTRACT

Mi, C.; Cao, L.; Zhang, Z.; Feng, Y.; Yao, L., and Wu, Y., 2020. A port container code recognition algorithm under natural conditions. *In*: Yang, Y.; Mi, C.; Zhao, L., and Lam, S. (eds.), *Global Topics and New Trends in Coastal Research: Port, Coastal and Ocean Engineering*. *Journal of Coastal Research*, Special Issue No. 103, pp. 822–829. Coconut Creek (Florida), ISSN 0749-0208.

Automatic container code recognition is very important for modern container intelligent management system. Under natural conditions, aiming at the problems of uneven illumination, tilt and deflection of container number in port container code recognition. A new differential edge detection algorithm is used to realize binary segmentation of uneven illumination container number image, and then the problem of accurate location of container number deflection is solved effectively by the improved least square method, then use gradient descent projection based character correction and segmentation algorithm to correct and segment tilt container number; BP neural network to recognize the segmented characters. Finally, experiments are carried out on the images taken under different conditions. The comprehensive recognition rate is 96.8%, the localization rate is 2.4% higher than the traditional method, and the comprehensive recognition rate is 6.5% higher than yolov3 algorithm, which meets the real-time requirements.

ADDITIONAL INDEX WORDS: *Container code recognition, differential edge binarization, least square method, tilt correction, BP neural network.*

INTRODUCTION

Nowadays, with the development of global economic integration, more and more international trade and cargo transportation depend on containers. The throughput of port containers is increasing rapidly (Tang, Xu, and Gao, 2019). Accurate and fast extraction of container number is very important for intelligent container management and transportation. However, computer vision-based automatic container code recognition (ACCR) has gradually become the mainstream way of modern port container code recognition because it does not need additional accessories and has low cost.

The container number adopts the ISO 6346 international standard, which consists of 3 parts (Frost, 2004), 4 capital case main English letters, 6-bit classification number and 1 check code. These 11 ISO characters are the unique identification codes for containers. and the typical container number image is shown in Figure 1. Generally, the key of technologies of automatic container code recognition (ACCR) system includes three modules: (1) location of container code area and tilt correction; (2) container code character segmentation; (3) container code character recognition (Panahi and Gholampour, 2017). But unlike the license plate, there are a series of problems in the container number image, such as uneven illumination, tilt and deflection, so, some researchers and engineers have done a lot of research in this field. Wu *et al*. (2012) uses horizontal high-pass filter and scan line analysis to detect container number area, and uses projection and two-step segmentation to segment container number characters, which effectively solves the problem of uneven illumination, but has poor robustness for container number samples with noise and tilt. Wu *et al*. (2015) divides container number area into single character block and multi-character block by connecting domain analysis and LLT method under the condition of inhibiting uneven illumination, and uses support vector machine (SVM) (Mi *et al*., 2016) and hidden Markov model (HMM) to realize two types of character recognition.

In recent years, because deep learning can directly rely on training character data to realize character recognition (Liang *et al*., 2019), it has been studied by a large number of scholars at present. Cao *et al*. (2017) used maximum stable extremum region (MSER) and spatial structure template to locate container number characters, and recognized characters by various CNN classifiers. To achieve end-to-end recognition of container code. Wang, Wang, and Xing (2019) uses Faster-RCNN, a target detection algorithm based on deep learning, to detect and identify container number characters. Wang and Wang (2020) uses improved Faster-RCNN and region production network (RPN) to detect container code, and finally uses improved CNN to recognize characters of interest. In order to simultaneously detect the container code on all sides of the container and output the optimal container number results. Yoon *et al*. (2016) uses decision-level integration method. The character recognition method based on deep learning can recognize different size strings through reasonable network design or improvement. Nevertheless, if the container number

DOI: 10.2112/SI103-170.1 received 21 November 2019; accepted in revision 4 February 2020.
*Corresponding author: chaomi@shmtu.edu.cn

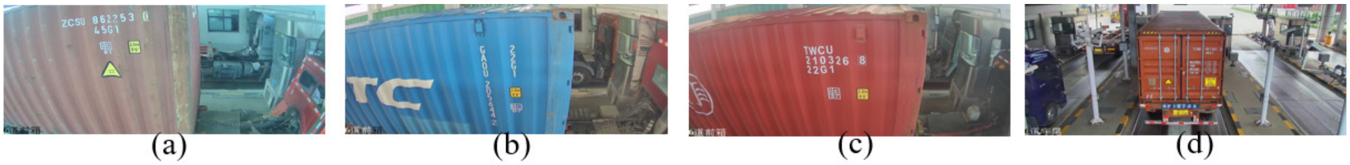

Figure 1. Example of container-code image.

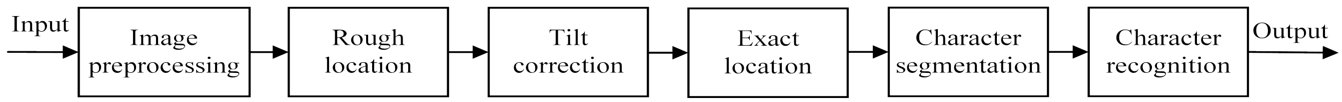

Figure 2. Flow chart of container code recognition.

is broken or inclined greatly, the accuracy of feature extraction will be affected, and the recognition accuracy of characters will be lower when a small number of samples are trained. At the same time, as the key technology of the automatic container code recognition system, the traditional tilt correction method has poor accuracy, which makes the container code recognition rate lower than the deep learning.

Therefore, a new differential edge detection algorithm and a coarse container number locating method based on neighboring clustering are proposed, which combines the existing research results with the collecting environment and the characteristics of the container number itself. Then, an improved least squares method and a character correction and segmentation algorithm based on gradient descent projection are used to realize the accurate locating and segmentation of the tilted container number. Finally, BP neural network is used for container code character recognition and compared with traditional algorithm and deep learning algorithm. The experimental results show that the method in this paper can improve the speed of container code recognition and realize high detection rate and low error rate of container code recognition.

CONTAINER CODE RECOGNITION ALGORITHM

The main flow of the proposed algorithm is shown in Figure 2. Firstly, image preprocessing and tilt correction are carried out to realize the accurate location of container code, The character segmentation and recognition.

A New Image Binarization Algorithm for Differential Edge Detection

(1) Traditional Differential Edge Detection Image Binarization

The derivative of the image function is obtained according to the first-order difference, and the difference edge detection is defined as the two-dimensional discrete image function:

$$f(x+1,y)-f(x,y)\quad f(x,y+1)-f(x,y) \tag{1}$$

The algorithm sets threshold value to extract edge information of image and binarizes it according to discriminant criterion (2):

$$out(x,y)=\begin{cases}255 & f(x,y)>\theta \\ 0 & other\end{cases} \tag{2}$$

where θ is threshold.

Characters segmented by traditional differential edge binarization algorithm with fixed parameters are prone to strokes breaking or sticking under uneven illumination. As shown in Figure 4.

(2) New Differential Edge Detection Image Binarization

Differential edge extraction operation is carried out on container gray image according to Equation (3):

$$G(x,y)=\left|f(x,y)-f(x,y+1)\right|+\left|f(x,y)-f(x+1,y)\right| \tag{3}$$

where $G(x,y)$ is the absolute value of the gray difference of the adjacent lower right pixel point.

Binarization of edge image according to Equation (4):

$$out(x,y)=\begin{cases}255 & G(x,y)>Threshold \\ 0 & G(x,y)\le Threshold\end{cases} \tag{4}$$

where $threshold$ is the dynamic threshold.

The process of the new differential edge detection image binarization algorithm is as follows:

1) Weighted averaging converts color container image into gray image (Panahi and Gholampour, 2017);

2) Set the size of container image is $M\times N$, which has N rows and M columns. Initialization threshold and edge detection number: $Threshold=1$, $Numcount=M\times N$;

3) Edge detection number is $Numcount=0$ and initial conditions is $Threshold=1$, if $Numcount/M/N>b$, proceed to step (3). If not, stop through the image;

4) Traverse the container image, extract the image edge according to Equation (3), if $G(x,y)>Threshold$ in Equation (5), then binarization of container edge image according to Equation (4), and the number of edge detections $Numcount+1$;

5) Dynamic threshold $Threshold+a$, proceed with steps (3) and (4), and detect the image edges that meets the conditions;

6) The obtained edge binary image is filled with region to form container binary image.

Through a large number of container number image sample experiments, a=0.07-0.08, b=3, ideal edge binarization image can be obtained. The flow chart of the algorithm implemented by the above steps is shown in Figure 3. The essence of the algorithm is to remove false edges and determine the real and potential edges, and then to achieve binarization (Liu *et al*., 2019). In the process of image binarization, a simple difference algorithm, Bernsen algorithm, OTSU algorithm and this algorithm are used to carry

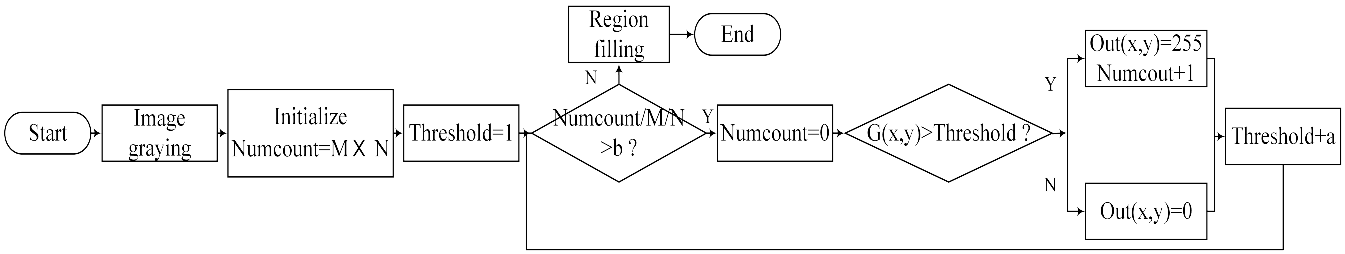

Figure 3. Flow chart of a new differential edge detection binary algorithm.

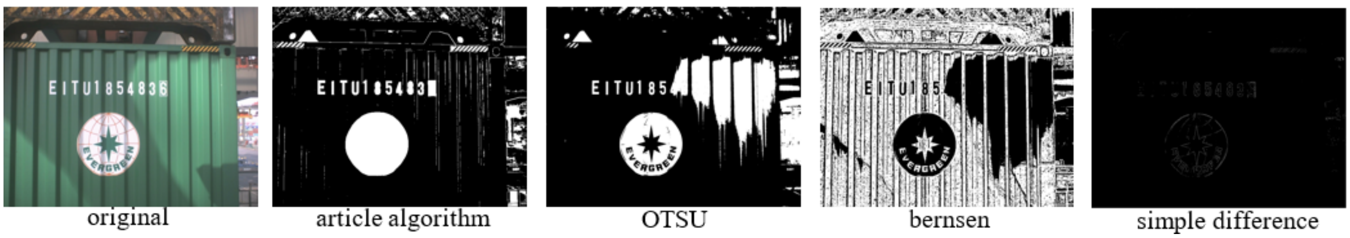

Figure 4. Comparison of various segmentation algorithms.

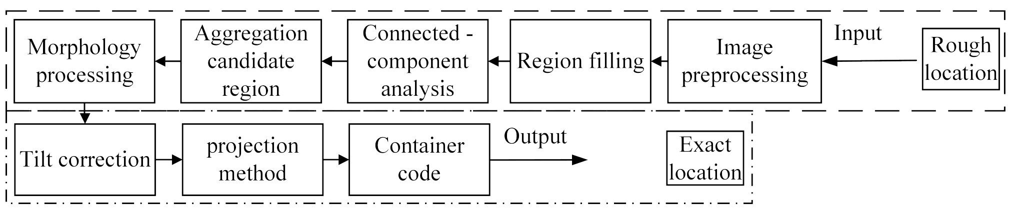

Figure 5. Container number location process.

Figure 6. Example of container number preprocessing.

out the experiment. The experimental results are shown in Figure 4. Experimental results show that the algorithm in this paper is effective and can segment characters accurately and clearly in the case of illumination unevenness. The container number pretreatment process is shown in Figure 6.

Location of Container Number

Container number localization is divided into rough localization and precise localization. Rough localization is mainly divided into two parts: connected component analysis and clustering candidate regions, the precise localization is mainly divided into two parts: horizontal tilt correction and projection method. The flow chart is shown in Figure 5.

(1) Rough Location Algorithm Based on Neighborhood Clustering

1. Connected component analysis

For the operation of contour extraction for several connected areas in the image, according to the connection area contour, the ellipse long axis (L), the short axis (S), the maximum value between the long axis and the short axis $\max(L,S)$, the minimum value between the long axis and the short axis $\min(L,S)$, the contour area (A) and other parameters, the non-container number area is removed, and the interested container number area is reserved, which provides the basis for subsequent accurate container number detection and recognition (Liu *et al*., 2020). A container number character area detection experiment was performed on 300 image of 500 experimental images. The following parameters were obtained by using strategy that can be mis-checked but not missed during the detection phase:

$$\begin{aligned} &\max(L,S) \le 100 \,\&\&\, \min(L,S) \le 25 \\ &A \le 700 \,\&\&\, \min(L,S) \le 80 \,\|\, A \ge 100 \\ &\max(L,S) \le 200 \,\&\&\, \min(L,S) \ge 13 \\ &A \le 1000 \,\&\&\, \min(L,S) \le 42 \end{aligned} \tag{5}$$

2. Adjacent clustering candidate regions

The connected domain analysis results in a single container number character candidate area, and the connected components are clustered. In the clustering algorithm, the measure of category similarity is the key to reflect the effectiveness of the clustering algorithm. Scan all the container number character candidate areas in the image, gets the area $(Area)$, width (w), center coordinate (x,y) of the minimum enclosing rectangle and the coordinates of the adjacent nearest rectangle vertex (X,Y), and cluster (Mi *et al*., 2019) by Equation (6):

$$\begin{aligned} &\sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2} \le 3.3\min\left(w_i, w_j\right) \\ &\sqrt{\left(X_a - X_b\right)^2 + \left(Y_a - Y_b\right)^2} \le 53 \\ &Area_i < 1050 \,\&\&\, Area_j < 1050 \end{aligned} \tag{6}$$

where (w_i, w_j) is the width of the minimum enclosing rectangle; $(x_i y_i)$, $(x_j y_j)$ and $(X_a X_a)$, $(X_b Y_b)$ are the central coordinates of the minimum enclosing rectangle of the (i,j) connected domain and the vertex coordinates of the adjacent nearest rectangle, respectively.

At the same time, for the non-aggregated container number area, morphological closed operation is used to aggregate the number area. The size of rectangular structural elements is 21x13, which realizes the rough locating of the container number. The example of rough location of container number is shown in Figure 7, and the rough positioning process of the container number is shown in Figure 8.

(2) Accurate Location Algorithm Based on Improved Least Squares Method

The container number characters after rough locating are tilted, and besides the container number characters, there are also non-container number characters which are very close (Quan *et al*., 2020). The container number cannot be precisely located directly by projection method, and will affect the subsequent container number character segmentation. In this paper, the center point of container number character column is used as feature point, and the tilt correction is carried out by improved least square method to achieve accurate locating.

The principle of the least square method to fit the straight line is as follows:

Let the linear equation be $y = ax + b$ where b is the intercept and a is the slope. According to the nature of the least squares method, the sum of squares of deviations between the fitted line and the input data should be the smallest, that is, the objective function $F = \sum_{i=0}^{n}(ax_i + b - y_i)^2$ is the smallest. In this way, partial derivatives are obtained by a and b in F:

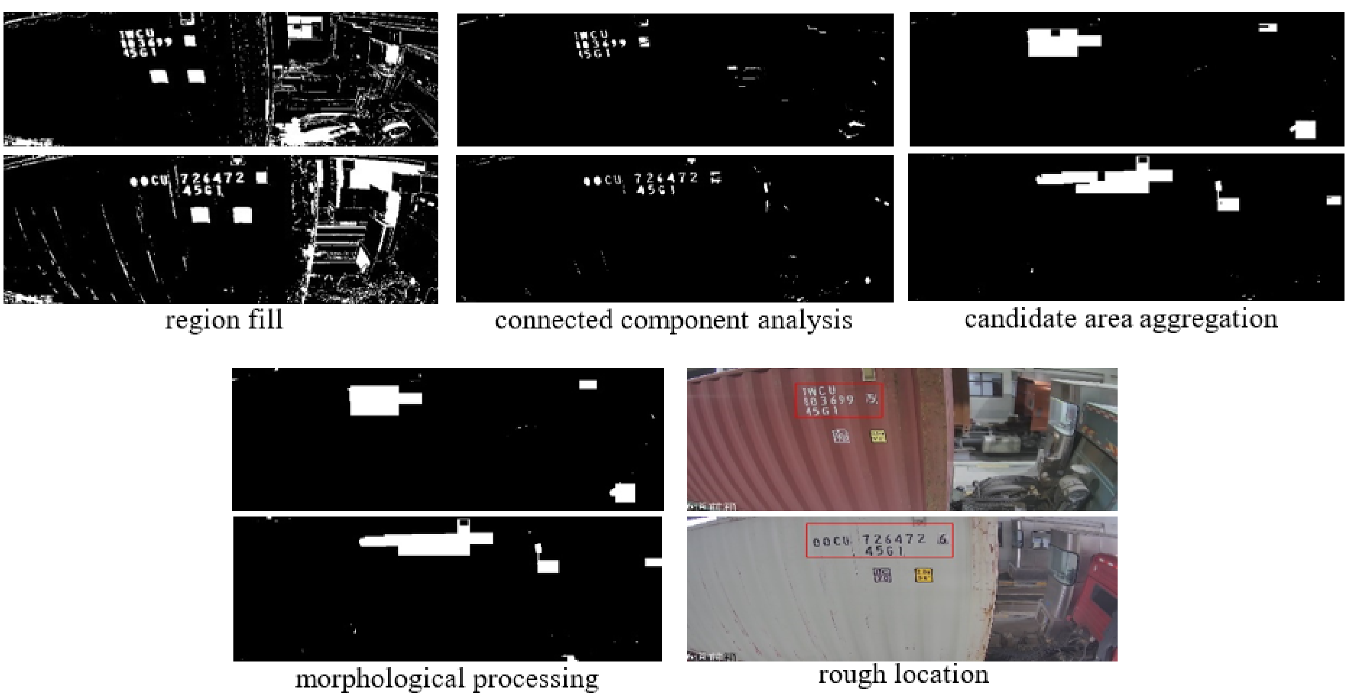

Figure 7. Example of rough location of container number.

$$\frac{\delta F}{\delta a}=0 \qquad \frac{\delta F}{\delta b}=0$$

$$a=\frac{\sum_{i=1}^{n} y_i - a\sum_{i=1}^{n} x_i}{n} \quad b=\frac{\sum_{i=1}^{n} y_i \sum_{i=1}^{n} x^2 - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} x_i y_i}{n\sum_{i=1}^{n} x_i^{\,2} - \left(\sum_{i=1}^{n} x_i\right)^2} \tag{7}$$

Find the values of a and b, then get the fitting straight line

The principle and implementation process of the improved least squares method are as shown in Figure 8:

(1) Diagram as shown in Figure 8, d_t is the t th defined as the threshold for removing interference points, $d_t = h \times d_{t\_\max}$, where $d_{t\_\max}$ is the distance from the farthest data point to the straight line after fitting the t th straight line and h is the proportional coefficient.

(2) The relationship between the total data point n and the reserved effective data point n_s is $n_s = n \times p$, where p is the proportional coefficient, if the distance from the set data point to the fitting line d_n is greater than d_t, the data will be deleted. The data will be deleted y and the remaining data $g = n - y$. The fitting line will be L_t by least square method every time.

Specific implementation process: Firstly, set the ratio parameter h and p, and use the least square method to fit the line L_t for the data n. Then, $d_t = h \times d_{t\_\max}$, if $d_n \geq d_t$, interference data point removal, if $d_n \leq d_t$, result retention. Secondly, if $g \leq n_s$, stop fitting, the fitted line L_t is the best, if $g > n_s$, repeat the previous step and continue to fit line L_{t+1}; finally, perform line fitting several times to get the best fitting line L_{t+1}.

The implementation process of the improved least square method for precise positioning:

(1) For the binarized graph of mixed tilted container number, the character-connected component is marked by seed filling algorithm to determine the upper left vertex coordinate (x_{n_0}, y_{n_0}) and the lower right vertex coordinate (x_{n_2}, y_{n_2}) of the rectangle outside each connected component. (n=1,2,3…k, k is the number of connected component);

(2) Let the size of the rectangle outside each character connected component be $M * N$, let X and Y represent the abscissa and ordinate coordinates of the center of each column of characters, and the central coordinates of each column can be (X_m, Y_m), where (m =1,2… M);

(3) Find the maximum y_{max} and minimum y_{min} of the ordinate of each column of pixels of a character. If $y_{max} = y_{\min}$, the ordinate of the column center point is $Y_m = y_{\min}$, otherwise the ordinate of the column center point is $Y_m = y_{min} + (y_{max} - y_{min})/2$.

(4) The coordinates of the center point of the column are moved into the rectangular coordinate system, and the straight line is fitted by the improved least square method.

(5) After obtaining the tilt angle α of the best fitting straight line, the horizontal tilt correction is realized by affine transformation. Accurate location of container by projection method.

The above method is experimented with 40 images. The object scale parameter h=0.80-0.82, p =0.8, which is the best fit for straight line, is studied in this paper. The precise positioning of container number is shown in Figure 9.

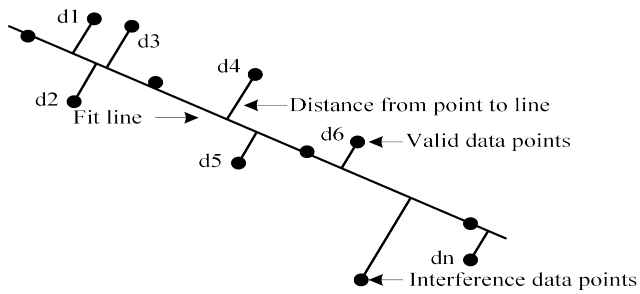

Figure 8. Schematic diagram of data point fitting straight line.

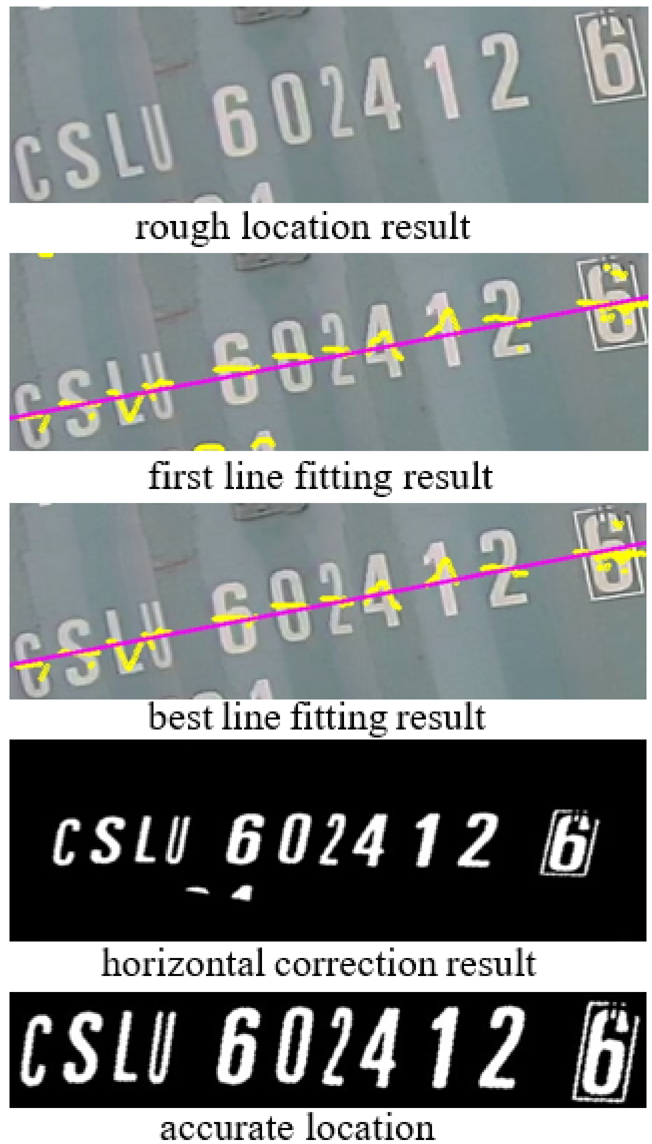

Figure 9. Example of container number accurate location process.

Container Number Segmentation and Recognition

(1) Character Correction and Segmentation Algorithm Based on Gradient Descent Projection

Precisely locational container numbers can also tilt vertically. If the character is too tilted, it will cause the character to stick in a certain direction. For later character recognition, using neural network is easy to cause misrecognition, such as 1 and 7. Therefore, this paper proposes a character correction and segmentation algorithm based on gradient descent projection, which rotates and projects the accurately positioned container number edge image through different step sizes, and finds out the rotation angle corresponding to the minimum projection length by using the projection negative gradient direction corresponding to the objective function, Then tilt correction is realized. The results of vertical correction are shown in Equation (8):

$$\min \quad f(L)=(\sum_{j=1}^{n} L_j)_\theta$$
$$L^{(i+1)} = L^{(i)} - \eta_t \nabla f(L^{(i)}) \tag{8}$$
$$s.t.\begin{cases} -30^\circ \le \theta \le 30^\circ \\ L_j > 0 \qquad j=1,2\ldots n \end{cases}$$

In order to avoid the gradient algorithm falling into local optimum, an appropriate angle range is selected so that only one trough exists in the range of character correction. The detailed realization process of the character correction and segmentation algorithm for gradient descent projection is as follows:

(1) Judging the direction of vertical tilt of container number. Set the step size to $\eta_t=\alpha^\circ$, the number of characters to n, the projection length of the j_{th} character to L_j, and the projection length under η_t to L_{η_t}, then $L_{\eta_t}=\sum_{j=1}^{n} L_j$, calculate the projection length of the original container number character to L_0, rotate the container number image to $\pm\eta_t$, calculate L_{α° and $L_{-\alpha^\circ}$, respectively, if $L_{\alpha^\circ} < L$, the direction is clockwise, the angle range is $\left[0^\circ, 30^\circ\right]$, if $L_{-\alpha^\circ} < L$, the direction is counterclockwise, the angle range is $\left[30^\circ, 0^\circ\right]$.

(2) Set the initial total projection length L_0, step $\eta_t=\beta^\circ$, convergence condition σ, angle θ. Projection gradient calculation $\nabla f(L)$, update L, use the updated $L^{(i+1)}$ to calculate $f(L^{(i+1)})$, compare the size relationship between $f(L^{(i+1)})$ and $f(L^{(i)})$, if $f\left(L^{(i+1)}\right)-f\left(L^{(i)}\right)>\sigma$, repeat the above steps, otherwise stop, and record the angle label.

(3) Set the step size to $\eta_t=\kappa^\circ$, repeat step 2 in range $\left[\text{label-}5^\circ, \text{label}+5^\circ\right]$ to find the label corresponding to the smallest $f(L)$.

(4) Set the step size to $\eta_t=\lambda^\circ$, repeat step 2 in range $\left[\text{label-}1.5^\circ, \text{label}+1.5^\circ\right]$ to find the tilt angle φ corresponding to the smallest $f(L)$. Vertical Tilt Correction by shear Transformation, projection for character segmentation.

The gradient descent projection character correction algorithm is shown in the algorithm below. Figure 10 shows the result of vertical correction, and Figure 11 shows the container number character image segmented by projection.

Gradient descending projection character correction algorithm

---

Require: Step-size η_t, Angle. θ, Convergence conditions σ
Repeat
Select container number X n character projection length sample $\{L^{(1)}, L^{(2)}, \cdots, L^{(n)}\}$, where $L^{(j)}$ corresponds to $y^{(j)}$;
Projection gradient calculation: $g \leftarrow \nabla_\theta \sum_j I(f(L^{(j)};\theta), y^{(j)}) / n$;
Parameter update: $L \leftarrow L - \eta_t g$
Until achieve convergence conditions

Figure 10. Vertical correction result image.

Figure 11. Container number character segmentation image.

(2) Character Recognition Algorithm Based on BP Network

Character statistical features extract and classify the most relevant information from the original data Single container number character is normalized and refined, and 13-point feature method is used to extract refined character features. Vertical correction has a good effect on character segmentation, so BP network algorithm is used. BP algorithm is also called error back-propagation algorithm. Its essence is to transform a group of sample input and output problems into a non-linear optimization problem, and to solve the weight problem by iterative operation through gradient descent algorithm (Zhang, 2017). The working process of neural network is mainly divided into two stages: learning and execution: 1) In the learning stage, given the training set, adjust the weight coefficient according to certain learning rules so as to minimize certain cost function, i.e. converge the weight coefficient to the optimum value. 2) In the execution stage, the input information is processed and the corresponding output is generated by using the connection weight coefficient obtained in the learning stage.

In the three-layer neural network designed in this paper, the number of neurons in the input layer is 13; the number of recognition nodes in the output layer is 5, and the number of recognition nodes is 4; the number of neurons in the hidden layer that distinguish numbers and letters is 8, the number of neurons in the hidden layer that recognize letters is 40, and the number of neurons in the hidden layer that recognize numbers is 32. In the BP network, the learning rate is 0.01, the momentum constant is 0.9, the display interval is 50, the maximum number of cycles is 5000, the learning rate decline multiplier factor is 0.8, and the error target is 0.0001.

EXPERIMENTAL RESULTS AND ANALYSIS

The experimental samples in this paper were collected at the entrance and exit of the Gate Lane in Taicang Port, Suzhou. The algorithm is implemented by using MATLAB R2017b programming on a computer with Intel Core i5-8300 H processor and 2.30 GHz CPU and 8 G memory. To verify the validity of this method, 1050 experimental samples were collected, including different environments, such as container type, lighting, tilt angle, and so on, with a resolution of 1920×1080. The number of successful cases (NS), number of failures (NF), success rate

Table 1. *Statistics of container codes recognition experiment result.*

Item	Algorithm	Time	Number	NS	NF	SR (%)	FR (%)	AT (ms)
Location	MSERDA	Day	650	615	35	94.7	5.3	463
		Night	400	368	32	92.2	7.8	
	THGBP	Day	650	566	84	87.2	12.8	513
		Night	400	330	70	82.5	17.5	
	Ours	Day	650	629	21	96.9	3.1	634
		Night	400	380	20	95.0	5.0	
Recognition	Yolov3	Day	650	597	53	91.8	8.2	20321
		Night	400	355	45	88.7	11.3	
	Literature (Tseng and Lee, 2011)	Day	650	590	60	90.9	9.1	1630
		Night	400	357	43	89.3	10.7	
	Ours	Day	650	630	20	97.0	3.0	1591
		Night	400	387	16	96.7	3.3	

Figure 12. Example of rough location and segmentation of container number.

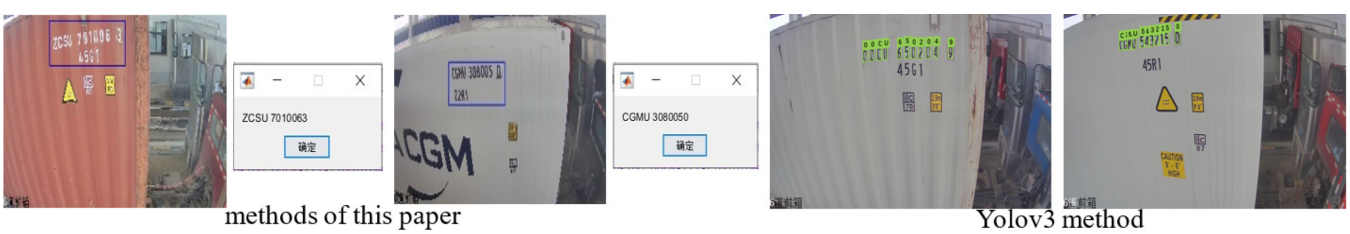

Figure 13. Example of Yolov3 and the recognition results of this method.

(SR), failure rate (FR), and average recognition time (AT) were counted. As shown in Table 1.

According to the analysis in Table 1, compared with MSERDA algorithm (Weng, Liu, and Guo, 2017) and THGBP algorithm (Wang and Xie, 2010), the algorithm in this paper shows higher accuracy and lower number of false alarms in container number positioning. Overall, the method in this paper has higher accuracy and better anti-interference. The time results of character processing by different algorithms show that the method in this paper meets the real-time requirements in practical application. The correct rate of character segmentation by projection method is 98%. Figure 12 is an example of rough positioning and segmentation of container number

In order to better illustrate the effectiveness of the overall recognition of this method, it is compared with Yolov3 algorithm in literature (Liu, Wang, and Xing, 2019) and the method in literature (Tseng and Lee, 2011). The experimental results show that the overall detection accuracy of this method is 96.8%, that of literature (Tseng and Lee, 2011) is 90.1%, and that of literature (Liu, Wang, and Xing, 2019) is 90.6%. This method improves greatly compared with Yolov3 and literature (Tseng and Lee, 2011), mainly in the stage of locating and segmentation in container number recognition. The tilted and noisy characters can be accurately located and segmented. Figure 13 shows an example of Yolov3 and the results of this method.

CONCLUSION

Aiming at the problems of uneven illumination, inclination and deflection of container number image, a new differential edge detection algorithm is used to realize binary segmentation of uneven illumination container number image, and then the problem of accurate location of container number deflection is solved effectively by the improved least square method, then use gradient descent projection based character correction and segmentation algorithm to correct and segment tilt container number. Finally, BP neural network is used to recognize container number characters. The experimental results show that the accuracy of this method is high, the comprehensive recognition rate is 96.8%, among which the location method is 2.4% higher than the traditional method, and the comprehensive recognition rate is 6.5% higher than Yolov3 algorithm, which meets the real-time requirements of the box number recognition system, and has a certain practical value.

ACKNOWLEDGEMENTS

The authors would like to thank editor and anonymous referees for their helpful and very delicate comments. This research was supported by the Science and Technology Commission of Shanghai Municipality (No.19595810700 and No.18295801100).

LITERATURE CITED

Cao, L.; Gai, Z.; Liu, E.; Gao, H.; Li, H.; Yang, L., and Li, H., 2017. Automatic container code recognition system based on geometrical clustering and spatial structure template matching. *International Conference on Communications*, 2198-2204.

Frost, R., 2004. International organization for standardization (ISO). *Quality Assurance Journal*, 8(3), 198-206.

Liang, Q.; Zhu, W.; Sun, W.; Yu, Z.; Wang, Y., and Zhang, D., 2019. In-line inspection solution for codes on complex backgrounds for the plastic container industry. *Measurement*, 148, 106965.

Liu, Y.; Yang, C.; Sun, Q., and Chen, Y., 2020. (k,n) scalable secret image sharing with multiple decoding options. *Journal of Intelligent & Fuzzy Systems*, 38(1), 219-228.

Liu, Y.; Yang, C.; Wu, C.; Sun, Q., and Bi, W., 2019. Threshold changeable secret image sharing scheme based on interpolation polynomial. *Multimedia Tools and Applications*, 78(13), 18653-18667.

Liu, Z.; Wang, Z., and Xing, Y., 2019. Wagon number recognition based on the YOLOv3 detector. *International Conference on Computer and Communication Engineering*, 159-163.

Mi, C.; Wang, J.; Mi, W.J.; Huang, Y.F.; Zhang, Z.W.; Yang, Y.S.; Jiang, J., and Octavian, P., 2019. Research on regional clustering and two-stage SVM method for container truck recognition. *Discrete & Continuous Dynamical Systems - S*, 12(4&5), 1117-1133.

Mi, C.; Zhang, Z.; Huang, Y., and Shen, Y., 2016. A fast automated vision system for container corner casting recognition. *Journal of Marine Science and Technology*, 24(1), 54-60.

Panahi, R. and Gholampour, I., 2017. Accurate detection and recognition of dirty vehicle plate numbers for high-speed applications. *IEEE Transactions on Intelligent Transportation Systems*, 18(4), 767-779.

Quan, Q.; Zou, H.; Huang, X., and Lei, J., 2020. Research on water temperature prediction based on improved support vector regression. *Neural Computing & Applications*.

Tang, S.; Xu, S., and Gao, J., 2019. An optimal model based on multifactors for container throughput forecasting. *Ksce Journal of Civil Engineering*, 23(9), 4124-4131.

Tseng, C. and Lee, S., 2011. Automatic container code recognition using compressed sensing method. *Conference on Multimedia Modeling*, 196-207.

Wang, W. and Xie, M., 2010. A novel location method of characters in container codes. *International Conference on Education Technology and Computer*, 3, V3-383-V3-385.

Wang, Z. and Wang, Y., 2020. FRCA: High-efficiency container number detection and recognition algorithm with enhanced attention. *International Conference Graphics and Image Processing*, 11373, 1137306.

Wang, Z.M.; Wang, W.X., and Xing, Y.X., 2019. Automatic container code recognition via faster-RCNN. *International Conference on Control and Automation*, 870-874.

Weng, M.; Liu, Q., and Guo, J., 2017. MSER and connected domain analysis based algorithm for container code locating process. *International Conference on Industrial Informatics*, 83-86.

Wu, W.; Liu, Z.; Chen, M.; Liu, Z.; Wu, X., and He, X., 2015. A new framework for container code recognition by using segmentation-based and hmm-based approaches. *International Journal of Pattern Recognition and Artificial Intelligence*, 29(01).

Wu, W.; Liu, Z.; Chen, M.; Yang, X., and He, X., 2012. An automated vision system for container-code recognition. *Expert Systems with Applications*, 39(3), 2842-2855.

Yoon, Y.; Ban, K.; Yoon, H., and Kim, J., 2016. Automatic container code recognition from multiple views. *Etri Journal*, 38(4), 767-775.

Zhang, M., 2017. Application of BP neural network in acoustic wave measurement system. *Modern Physics Letters B*, 31(19-21), 1740052.