Hindawi
Journal of Advanced Transportation
Volume 2021, Article ID 9612480, 9 pages
https://doi.org/10.1155/2021/9612480

Research Article

Truck-Lifting Prevention System Based on Vision Tracking for Container-Lifting Operation

Qingfeng Huang,[1,2] **Yage Huang**,[3] **Zhiwei Zhang**,[4] **Yujie Zhang**,[3] **Weijian Mi**,[1] **and Chao Mi**[1]

[1]*Container Supply Chain Technology Engineering Research Center Ministry of Education, Shanghai Maritime University, Shanghai, China*
[2]*East China Headquarters of China Communications Construction Company Limited, Beijing, China*
[3]*Logistic Engineering School, Shanghai Maritime University, Shanghai, China*
[4]*Shanghai SMUVision Smart Technology Ltd., Shanghai, China*

Correspondence should be addressed to Chao Mi; chaomi@shmtu.edu.cn

Received 13 May 2021; Revised 6 July 2021; Accepted 14 November 2021; Published 1 December 2021

Academic Editor: Avinash Unnikrishnan

Truck-lifting accidents are common in container-lifting operations. Previously, the operation sites are needed to arrange workers for observation and guidance. However, with the development of automated equipment in container terminals, an automated accident detection method is required to replace manual workers. Considering the development of vision detection and tracking algorithms, this study designed a vision-based truck-lifting prevention system. This system uses a camera to detect and track the movement of the truck wheel hub during the operation to determine whether the truck chassis is being lifted. The hardware device of this system is easy to install and has good versatility for most container-lifting equipment. The accident detection algorithm combines convolutional neural network detection, traditional image processing, and a multitarget tracking algorithm to calculate the displacement and posture information of the truck during the operation. The experiments show that the measurement accuracy of this system reaches 52 mm, and it can effectively distinguish the trajectories of different wheel hubs, meeting the requirements for detecting lifting accidents.

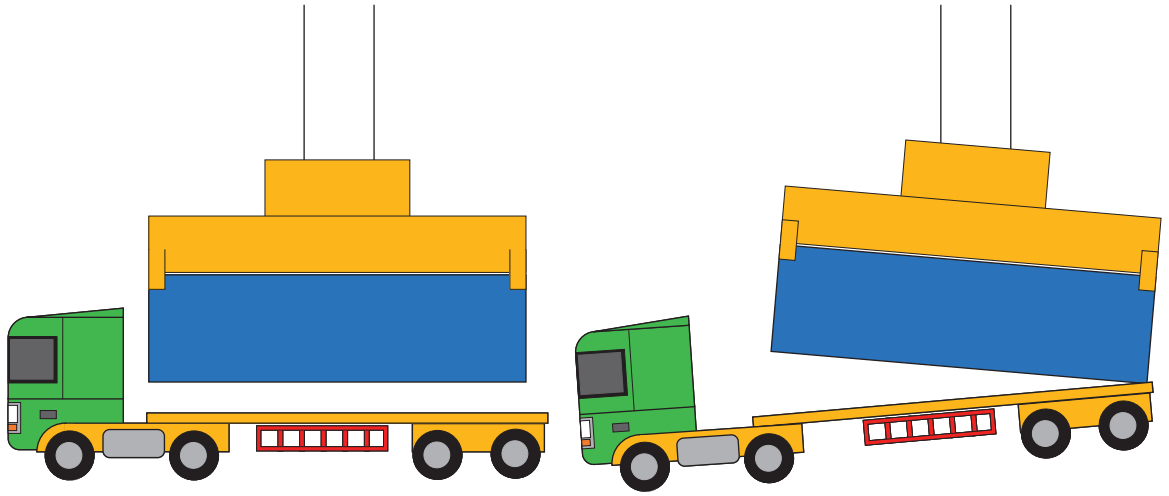
1. Introduction

Container terminals are facilities that provide storage and distribution services for container transportation. With the sustainable growth of global maritime trade, the development focus of container terminals has moved to automation and unmanned operations. Furthermore, terminals with a high level of automation are called automated container terminals (ACTs). Some of the advantages of ACTs are obvious, they use automated equipment to replace on-site workers, which improves operation efficiency and reduces operating costs; this also improves worker safety [1].

In the terminal operation process, containers need to be transferred between various storage areas to transfer equipment. These transfer operations are called container-lifting operations and are performed by container-lifting equipment (such as rail-mounted gantry cranes (RMG)) [2]. The truck-lifting accident is an accident that occurs in container-lifting operations, and an example is shown in Figure 1. When the container is lifted, the container lock pins are not released, and the truck is lifted with the container, which can negatively affect the container, trucks, and on-site workers.

In traditional container terminals, the container-lifting operation requires on-site workers to confirm whether the lock pin is fully released. However, the ACT requires a reduction in the number of on-site workers, and an automated accident detection method is required to prevent accidents.

Truck-lifting prevention can be considered as a target detection and tracking problem. It needs to detect and recognize the characteristics of the truck and then use it to calculate the displacement of the truck during the operation



(a) (b)

Figure 1: (a) Normal lifting. (b) Chassia lifting accident.

process. Existing solutions for truck-lifting prevention are based on laser scanners, such as the laser radar-based truck chassis positioning technology proposed by Chao-feng [3]. The laser scanner scans the contour information of target and restores it to a 3D model. By analyzing the geometry of the model, the size and position information of the targets is calculated by the system [4]. This technology has high detection accuracy and is not affected by weather or light conditions. However, this system relies on a high-precision laser scanner, which is expensive [5].

With the development of image sensors and computer vision algorithms, vision-based measurement (VBM) technology has become more widespread in recent years. This technology consists of only a camera and an image processing device, which makes its hardware cost much lower than the laser scanner solution. VBM technology has been widely used in industrial measurement and one of the typical applications of VBM is the automated inspection of product quality control [6]. In container terminals, vision-based detection technology is used in many applications [7], specifically in the recognition of complex features, such as container numbers [8] and container corner casting [9].

In addition to the lower equipment cost, vision-based detection technology has the following two advantages. One is that it can achieve higher measurement accuracy by noncontact measurement [10] because vision-based detection uses CMOS or CCD cameras to obtain image information; these devices have high image resolution. Another advantage is its ability to recognize complex features, which stems from convolutional neural network technology (CNN) [11].

CNNs can recognize and classify complex features from images, such as the classification of face features [12] and the classification of tumors [13]. Different recognition CNNs also have operability for training. Compared with the previous classification (such as SVM), CNN has achieved higher detection rate, detection accuracy, and calculation time [14, 15].

Nevertheless, the detection accuracy of CNNs is not perfect. The detection results of CNN are the area with the highest probability that contains the target. There is generally some deviation between the detected result and the target. However, traditional image processing has pixel-level accuracy and can achieve higher accuracy under the premise of successful detection.

Vision-based target tracking technology has been used in several applications, such as ship recognition and tracking based on video information [16, 17] and vehicle tracking based on aerial videos [18]. These technologies are usually based on detection-based tracking (DBT) [19], which is mainly because of the excellent target detection ability shown by CNN detection. The tracking principle of DBT is to use a CNN to detect the target from an image and then use the correlation algorithm to associate the same target in different frames [20]. This system has achieved a good tracking result, making the main problem of vision tracking change from detection to association.

This study proposes a truck-lifting prevention system based on a vision-based detection and tracking algorithm to provide a low-cost and easy-to-modify automated accident detection system for container-lifting operations. The system is based on a target detection method that combines CNN detection with traditional image processing algorithms and a DBT multitarget tracking algorithm. The system calculates the displacement of the truck wheel hub and determines whether an accident has occurred. This system supports real-time remote monitoring because it uses cameras to capture operation information. Moreover, the system can switch to manual monitoring when the accident detection algorithm fails, which is a function that the laser scanner solutions cannot achieve.

2. System Design and Control Principle

This system uses cameras as information capture devices, which makes it applicable for installation in most container-lifting equipment. Figure 2 shows the installation in the rail-

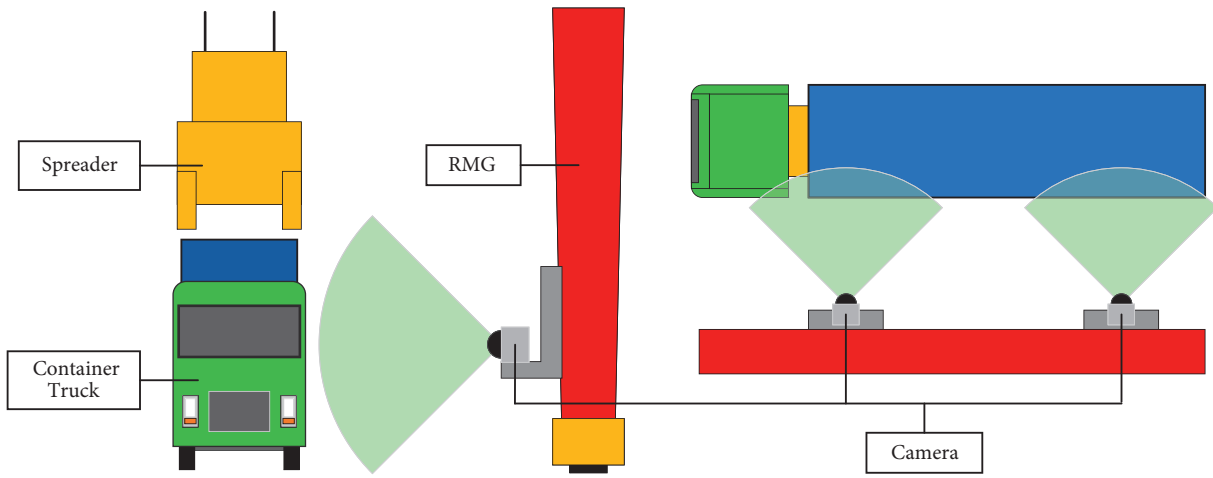


Figure 2: Installation of cameras.

mounted container gantry crane (RMG), which is a typical container-lifting equipment in a container terminal, and Figure 3 shows the actual installation of the cameras. At the operation site, container trucks only function on the truck road; therefore, the cameras were installed at the RMG leg to capture the image information of the trucks. Two sets of cameras were installed in the RMG leg to cover all the areas because the container truck has a long chassis.

The lifting prevention process is shown in Figure 4. When the operation starts, the cameras capture the image on the side of the truck and sends it to the image processing unit (IPU) for calculation. In the IPU, the wheel hubs of the truck are detected first and then the movement trajectory of each wheel hub during the operation is tracked to determine whether the truck has been lifted. When an accident is detected, the IPU sends the accident information to the automated crane control system (ACCS), which stops lifting the container spreader by controlling the programmable logic controller (PLC).

The reliability of the equipment was also considered. In the traditional operation process, the container-lifting operation needs to be guided by on-site workers. We did not install a backup system because it would add additional costs and complicate the communication systems. When the system fails, the traditional method is considered acceptable. As the camera is installed at a low position that can capture the image of the wheel, it can be easily wiped off when the lens of cameras is stained.

3. Truck-Lifting Detection Algorithm

There are several types of container trucks, and therefore, it is difficult to directly recognize truck chassis and measure their displacement. However, truck tires have standard specifications, and considering that the tires deform under normal loading conditions, we calculated the displacement information of the truck chassis by detecting the coordinates of the wheel hubs on the image. Because the work site was an open-air environment, the light conditions were unstable, and the color and contamination conditions of different

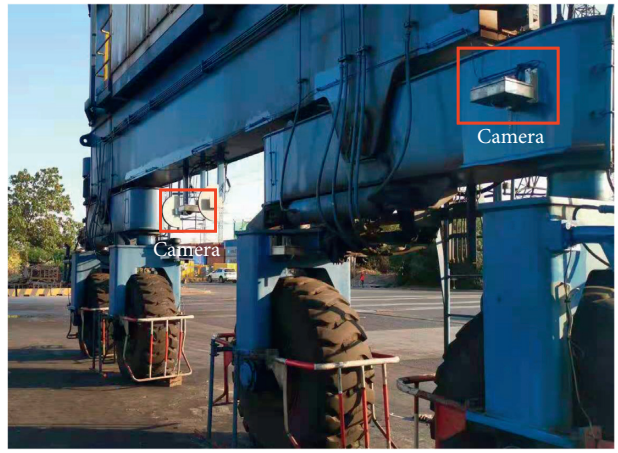


Figure 3: Actual installation of cameras.

trucks were also different. We first used neural network detection to obtain a higher target recognition rate and then used traditional image algorithms to improve the detection accuracy. Finally, a deep sort-based tracking algorithm was used to distinguish and track different wheel hubs.

3.1. First Wheel Hubs Detection Based on the Modified SSD. SSD (Single Shot MultiBox Detector) [21] is a feedforward convolutional network. It uses anchor boxes with different aspect ratios and sizes to sample the image, and several feature layers with different receptive fields are used to extract and classify features. Owing to this design, SSD has a higher detection speed than two-stage methods, such as Fast region-based convolutional network (Fast R-CNN) [22], making it suitable for real-time detection.

To achieve the best detection performance, we made some modifications to the SSD network. The original SSD has VGG-16 [23] as the convolutional layer, which was replaced with ResNet [24], a newer CNN model that uses deeper neural networks to extract more feature information. The structure of the modified SSD model is illustrated in Figure 5.

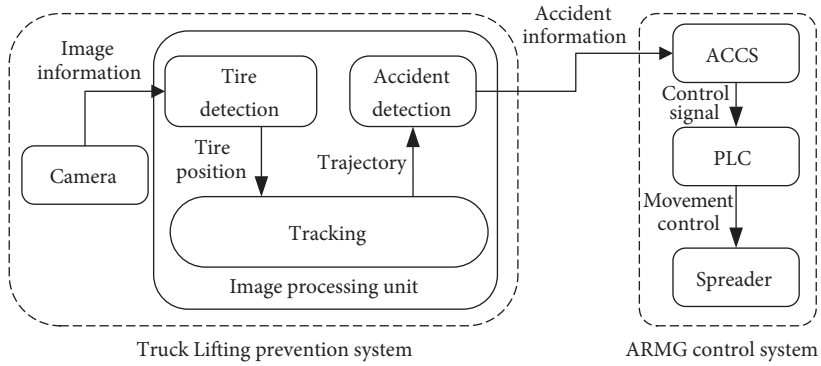


Figure 4: Principle of truck-lifting prevention system.

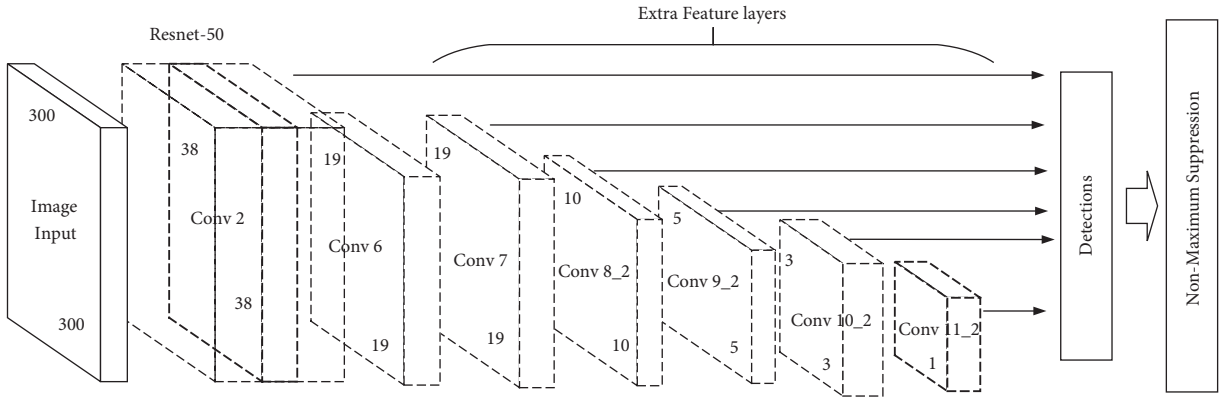


Figure 5: The structure of modified SSD.

3.2. The Second Detection Stage Based on Traditional Image Processing. The result of SSD detection is not the target itself; however, it is an area that contains the target with the greatest probability. The detection results usually have a positioning error from the actual target position. However, traditional image processing algorithms have pixel accuracy, but for an entire image, it takes longer time to calculate; hence, requirements for real-time detection are not met. Therefore, after SSD detection, we perform a second-wheel hub detection based on traditional image processing to improve the detection accuracy.

A flowchart of the second detection is shown in Figure 6. The input data are the wheel hub image that was detected by the SSD. The results detected by SSD are defined in (1), where x_0 and y_0 are the center coordinates of the detection result and s_0 and r_0 are the size and aspect ratio, respectively:

$$D_0 = [x_0, y_0, s_0, r_0]. \tag{1}$$

The first part is the preprocessing operation. We used the Single-scale Retinex (SSR) algorithm to enhance the information in the dark area of the image because the operation site is open air and the light conditions are unstable. SSR was proposed by Jobson et al. [25], and it is based on Land's Retinex theory [26]. This enhancement algorithm uses a Gaussian wrap function to convolve the image, and its expression is as follows:

$$R_i(x, y) = \log I_i(x, y) - \log\left(F(x, y) * I_i(x, y)\right), \tag{2}$$

where $I_i(x, y)$ is the original color value of the point (x, y) on the color channel I, $R_i(x, y)$ is the enhanced color value, And $F(x, y)$ is the Gaussian wrap function and its calculation is shown in (3). C represents the scale value of Gaussian wrap; it means the neighborhood size of (x,y) during convolution operation. λ is a scale parameter; it must make (4) hold. The enhanced image is the merged result of each color channels:

$$F(x, y) = \lambda e^{\left[-x^2+y^2/c\right]}, \tag{3}$$

$$\int\int F(x, y)\mathrm{dxdy} = 1. \tag{4}$$

Next, we used an adaptive HSV threshold to filter out the wheel hub area in the image. The HSV color space divides different colors by the hue H, saturation S, and brightness value V. Since to the wheel hub area usually is the higher brightness part in the image, it can be extracted from the image by filtering the lower brightness part of the image. The calculation of HSV thresholding is shown in (5); $T_V(x, y)$ is the pixel value of pixel (x, y) in the HSV V space and $\mathrm{Th}(x, y)$ is the new pixel value. Thresh_V is the threshold value that is calculated from the average pixel value of the image ($\overline{T_V(x, y)}$), and the adjustment value is ε; its calculation is shown in (6). The preprocessed image is shown in

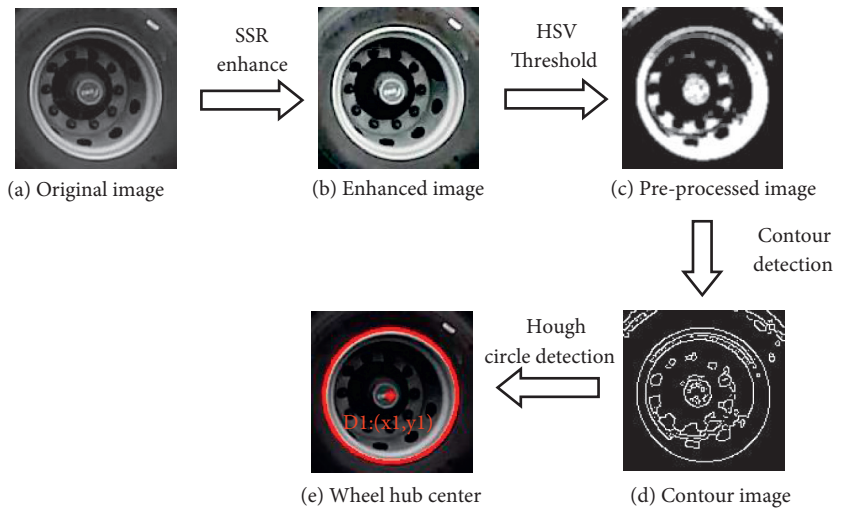


Figure 6: Image processing process.

Figure 6(c); most of the noise pixels have been removed by HSV thresholding:

$$Th(x,y) = \begin{cases} T_V(x,y), & if\ T_V(x,y) > \text{Thresh}_V, \\ 0, & \text{otherwise}, \end{cases} \quad (5)$$

$$\text{Thresh}_V = \varepsilon \times \overline{T_V(x,y)}. \quad (6)$$

The second part is the detection of the contours on the preprocessed image, and the calculation of the largest circle in the contours is by Hough circle detection. Owing to the light condition and lens distortion, the wheel hub image at the edge of the image screen exhibit some deformation and defects. We used the adaptive method shown in (7) to adjust the threshold of the Hough circle detection accumulator so that the Hough circle detection can detect the largest circles with no perfect shapes. X in (7) is the horizontal resolution of the image, and A_0 and A_1, respectively, represent the original threshold and adjusted threshold of the accumulator, and c is the adjustment ratio:

$$A_1 = A_0 - c\left|\frac{X}{2} - x_0\right|. \quad (7)$$

The second detection result is defined as D_1, as shown in (8). Because the second detection is unstable, when the second detection result D_1 and the first detection result D_0 have a large deviation, it should be considered as a failure of the second detection. Therefore, the final detection result needs a re-evaluation, and we used (9) to estimate whether the result of the second detection is suitable as the final detection result. $N_{0.95}$ is the maximum error with 95% confidence of the first detection, which is calculated by normal fitting:

$$D_1 = [x_1, y_1, s_0, r_0], \quad (8)$$

$$(x_2, y_2) = \begin{cases} (x_0, y_0) & if\ \sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2} > N_{0.95} \\ (x_1, y_1) & if\ \sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2} < N_{0.95} \end{cases} \quad (9)$$

3.3. Trajectory Tracking Based on the Modified Deep Sort. Deep Sort [27] is an online multiobject tracking algorithm proposed by Wojke et al. in 2017. As a DBT algorithm, Deep Sort's tracking is based on the detection result data, making it suitable for combination with CNN detection or traditional image processing detection. We modified the tracking process of Deep Sort to improve the tracking speed, and the new tracking process is shown in Figure 7.

The Deep Sort uses the state vector shown in (10) as the description model of the targets. u and v are the center coordinates of the target detection result, and γ and h represent the aspect ratio and height of the target detection result, respectively. $\dot{u}, \dot{v}, \dot{\gamma}$, and $\dot{h}$ are the predicted target positions in the next frame, which are predicted by Kalman fitting, an algorithm that uses a series of measurements observed over time to produce estimates of unknown variables:

$$X = [u, v, \gamma, h, \dot{u}, \dot{v}, \dot{\gamma}, \dot{h}]^T. \quad (10)$$

The predicted result is used to match the detection results in the next frame. The matching algorithm is based on the Kuhn–Munkres algorithm, which uses the IOU value of the prediction result and detection result as the weight to classify different tracking targets. The calculation of IOU is shown in (11), Dete is the detection result, and Pred is the prediction result. The detection result closest to the prediction result was classified as the same target:

$$IOU = \frac{\text{Area}(\text{Dete}) \cap \text{Area}(\text{Pred})}{\text{Area}(\text{Dete}) \cup \text{Area}(\text{Pred})}. \quad (11)$$

To solve the problem of target loss when the target passes through obstacles, the original Deep Sort uses the Mahalanobis distance and the descriptor of the target after the convolution operation to match the detection result and existing trajectories. However, the calculation of the convolution descriptor requires longer time, which makes the calculation time of Deep Sort much longer than Sort [28]. Therefore, we only used the Mahalanobis distance as the standard for trajectory matching.

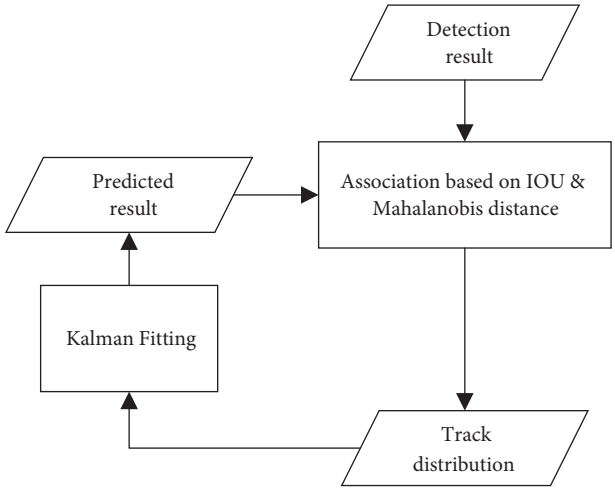


Figure 7: The tracking principle of modified Deep Sort.

The calculation of the Mahalanobis distance is shown in (12), $d_{i,j}$ is the motion matching value between the trajectory i and the detection result j, S_i is the covariance matrix of the observation space in this frame, which is obtained by the Kalman filter.

$$d_{i,j} = \left(d_j - y_j\right)^T S_i^{-1} \left(d_j - y_j\right). \tag{12}$$

Because the motion of the target is continuous, the Mahalanobis distance can be used to screen the detection results. The calculation method of the screening is shown in (13). $t^{(1)}$ is the threshold that is defined by the chi-square distribution with 0.95 degrees. When $d_{i,j}$ is lower than the threshold, it means that the trajectory I is associated with the detection result j.

Because the motion of the target is continuous, the Mahalanobis distance can be used to screen the detection results. The screening calculation method is given in (13). $t^{(1)}$ is the threshold defined by the chi-square distribution with 0.95 degrees. When $d_{i,j}$ is lower than the threshold, it means that trajectory I is associated with the detection result j:

$$b_{i,j} = \prod \left[d_{i,j} \le t^{(1)}\right]. \tag{13}$$

4. Experiment

The core of this lifting prevention system is the wheel hub detection and tracking method. We used a typical industrial computer configuration to verify our method, the specifications of which are as follows:

CPU : Intel i7-6700

GPU : Nvidia GeForce GTX970-4 GB.

The first detection algorithm was implemented by Pytorch [29], and the second detection and tracking algorithm was implemented using OpenCV [30] in a Python environment. The images used in the experiment were captured by a camera, as shown in Figure 2. The resolution of the image was 1920 × 1080, and fps was 24.

Figure 8: First detection result.

4.1. Evaluation of Wheel Hub Detection. The modified SSD training used 3000 images from the side of the container truck. The trucks in these images were driven on the truck road next to the RMG, and the distance between the camera and the trucks was approximately 4–6 m. The first detection result is presented in Figure 8.

The performance evaluation of the first detection used 500 images for testing, and the evaluation of the second detection used 500 images that only contained the tire part. The test results are listed in Table 1. The horizontal error is the distance between detection result and the center of wheel hub in the direction of truck road, and the vertical error is the distance between detection result and the center of wheel hub in the vertical direction; both error values are the 95% confidence value after normal fitting. The actual distance was estimated with reference to the pixel size of the wheel hub.

4.2. Evaluation of Wheel Hub Tracking. The tracking algorithm evaluation used several videos of trucks passing through the camera area at normal speed and several videos of trucks under the container-lifting operations. The former was used to test the track of the horizontal displacement of the truck, and the latter was used to test the vertical displacement. These videos were under normal light conditions during the daytime and night-time, and the tracking results are shown in Figures 9 and 10.

Table 2 lists the performance of the target tracking algorithm. The tracking error is defined as the distance between the detection result and the prediction result, and it is the maximum error at the 95% confidence level after normal fitting.

4.3. Discussion. The experimental results showed a detection error of 6.31 pixels (in the experimental environment, it was approximately 52 mm), and the total tracking rate (including the detection time) reached 10 fps (average of 2.5 tires per image). Because the maximum vertical displacement in the container-lifting operation was approximately 100 mm, the detection accuracy of this system met the requirements of truck-lifting prevention. However, the experimental results also showed some issues.

In the detection experiment, certain detection failures were observed, and these failure detection samples were focused on the second detection. We observed that these failures were caused by tires with defaced and low light

Table 1: Performance of detection.

Item	Detection success rate (%)	Horizontal error		Vertical error		Detection time
		Pixel value	Actual value (mm)	Pixel value	Actual value (mm)	
SSD detection	99.4	10.72 pixels	90	8.23 pixels	68	20.3 ms
Image detection	95.1	6.31 pixels	52	4.23 pixels	35	34.8 ms

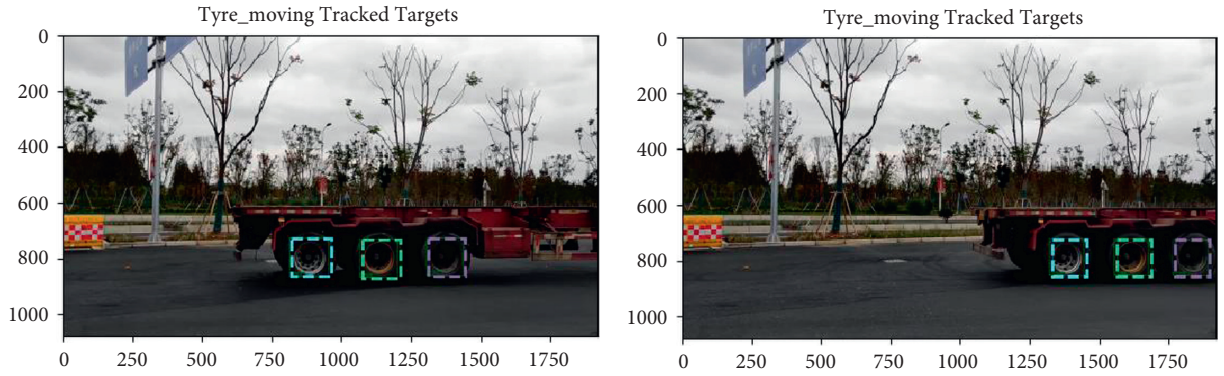


Figure 9: Wheel hub tracking in truck movement.

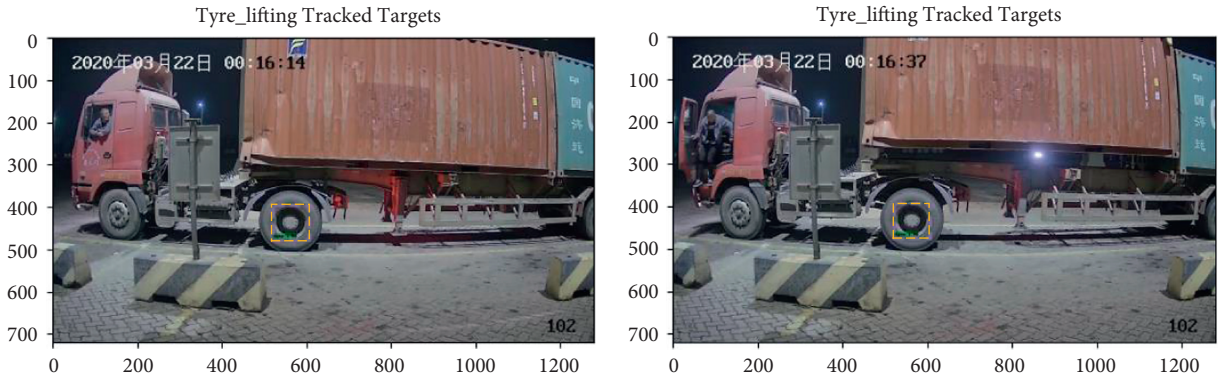


Figure 10: Wheel hub tracking in container lifting.

Table 2: Performance of tracking.

Item	Tracking success rate (%)	Horizontal error		Vertical l error		Tracking time (ms)
		Pixel value	Actual value (mm)	Pixel value	Actual value (mm)	
Result	99.6	2.54 pixels	13	4.52 pixels	37	5.6

environments, and such factors obscured the details of the wheel hub. In this study, we used the processing of pixel values in the HSV space to solve this problem, but the experimental results showed that it is not sufficient.

When the tire appeared on the edge of the image, the detection error of the second detection increased. This is because the cameras have some lens distortion, which is caused by lens distortion and coordination. This causes distortion of the edge part of the image.

5. Conclusion

To solve the problem of automated accident prevention in container-lifting operations, this study designed a vision-based truck-lifting prevention system that calculates the displacement of the truck wheel hubs to determine whether the truck is lifted. The experiment showed that the detection accuracy of this system reaches 6.31 pixels and the average fps is 10 frames, which is sufficient to detect the truck-lifting accident in time.

However, certain limitations were also observed. We believe that an algorithm to extract the contour characteristics from the tire images with defaced and low light environment should be explored. Considering that the convolutional neural network is insensitive to different defaced and light conditions, it may be possible to use the convolution operation to extract detailed information in the picture to avoid the interference of light and defacement. However, complex calculations will increase the calculation time and reduce the efficiency of the system; therefore, this problem needs to be resolved.

Data Availability

The experiment data used to support the findings of this study have been deposited in the Google Drive repository (https://drive.google.com/file/d/1mqZrmlnOMwxeLsM9pBxItZ_jMj4qsRrV/view?usp=sharing).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported by the Science and Technology Commission of Shanghai Municipality (no. 202H1101900) and China (Shanghai) Pilot Free Trade Zone Lin-gang Special Area Administration (no. SH-LG-GK-2020-21).

References

[1] W. Yan, Y. Zhu, and J. He, "Performance analysis of a new type of automated container terminal," *International Journal of Hospitality Information Technology*, vol. 7, no. 2, pp. 237–248, 2014.

[2] B. Brinkmann, *Operations Systems of Container Terminals: A Compendious Overview Handbook of Terminal Planning*, Springer, New York, NY, USA, 2011.

[3] L. Chao-Feng, D. U. Zheng-Chun, and B. Brinkmann, *Operations Systems of Container Terminals: A Compendious Overview Handbook of Terminal Planning*, pp. 25–39, Springer, New York, NY, USA, 2011.

[4] W. Zhen, S. Zeng, and S. Soberer, "Robust localization and localizability estimation with a rotating laser scanner," in *Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6240–6245, IEEE, Singapore, June 2017.

[5] C. Torresan, A. Berton, F. Carotenuto et al., "Development and performance assessment of a low-cost UAV laser scanner system (LasUAV)," *Remote Sensing*, vol. 10, no. 7, p. 1094, 2018.

[6] M. Chao, C. Kai, and Z. Zhiwei, "Research on tobacco foreign body detection device based on machine vision," *Transactions of the Institute of Measurement and Control*, vol. 42, no. 15, pp. 2857–2871, 2020.

[7] C. Mi, Y. Huang, C. Fu, Z. Zhang, and O. Postolache, "Vision-based measurement: actualities and developing trends in automated container terminals," *IEEE Instrumentation and Measurement Magazine*, vol. 24, no. 4, pp. 65–76, 2021.

[8] C. Mi, L. Cao, Z. Zhang, Y. Feng, L. Yao, and Y. Wu, "A port container code recognition algorithm under natural conditions," *Journal of Coastal Research*, vol. 103, no. SI, pp. 822–829, 2020.

[9] C. Mi, Z. W. Zhang, Y. F. Huang, and Y. Shan, "A fast automated vision system for container corner casting recognition[J]," *Journal of Marine Science and Technology*, vol. 24, no. 1, pp. 54–60, 2016.

[10] N. V. Ngo, Q. C. Hsu, W. L. Hsiao, and C. J. Yang, "Development of a simple three-dimensional machine-vision measurement system for in-process mechanical parts," *Advances in Mechanical Engineering*, vol. 9, no. 10, Article ID 1687814017717183, 2017.

[11] Y. D. Li, Z. B. Hao, and H. Lei, "Survey of convolutional neural network," *Journal of Computer Applications*, vol. 36, no. 9, pp. 2508–2515, 2016.

[12] W. Wu, Y. Yin, X. Wang, and D. Xu, "Face detection with different scales based on faster R-CNN," *IEEE Transactions on Cybernetics*, vol. 49, no. 11, pp. 4017–4028, 2019.

[13] J. Y. Chiao, K. Y. Chen, K. Y. K. Liao, P. H. Hsieh, G. Zhang, and T. C. Huang, "Detection and classification the breast tumors using mask R-CNN on sonograms," *Medicine*, vol. 98, no. 19, 2019.

[14] K. T. Islam, R. G. Raj, and A. Al-Murad, "Performance of SVM, CNN, and ANN with BoW, HOG, and image pixels in face recognition," in *Proceedings of the 2017 2nd International Conference on Electrical & Electronic Engineering (ICEEE)*, pp. 1–4, IEEE, Rajshahi, Bangladesh, December 2017.

[15] R. L. Galvez, A. A. Bandala, E. P. Dadios, R. R. P. Vicerra, and J. M. Z. Maningo, "Object detection using convolutional neural networks," in *Proceedings of the TENCON 2018 - 2018 IEEE Region 10 Conference*, pp. 2023–2027, Jeju, Korea (South), 2018.

[16] X. Chen, S. Wang, C. Shi, H. Wu, J. Zhao, and J. Fu, "Robust ship tracking via multi-view learning and sparse representation," *Journal of Navigation*, vol. 72, no. 1, pp. 176–192, 2019.

[17] X. Chen, L. Qi, Y. Yang et al., "Video-based detection infrastructure enhancement for automated ship recognition and behavior analysis," *Journal of Advanced Transportation*, vol. 2020, Article ID 7194342, 2020.

[18] X. Chen, Z. Li, Y. Yang et al., "High-resolution vehicle trajectory extraction and denoising from aerial videos," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 5, pp. 3190–3202, 2020.

[19] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, and K. Schindler, "MOT16: a benchmark for multi-object tracking," 2016, https://arxiv.org/abs/1603.00831.

[20] W. Luo, J. Xing, A. Milan et al., "Multiple object tracking: a literature review," *Artificial Intelligence*, vol. 293, no. 2, p. 103448, Article ID 103448, 2021.

[21] W. Liu, D. Anguelov, D. Erhan et al., "SSD: s detector," *Computer Vision-ECCV 2016*, pp. 21–37, 2016.

[22] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time object detection with region proposal networks," 2015, https://arxiv.org/abs/1506.01497.

[23] Q. Guan, Y. Wang, B. Ping et al., "Deep convolutional neural network VGG-16 model for differential diagnosing of papillary thyroid carcinomas in cytological images: a pilot study," *Journal of Cancer*, vol. 10, no. 20, pp. 4876–4882, 2019.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, Las Vegas, NV, USA, June 2016.

[25] D. J. Jobson, Z. Rahman, and G. A. Woodell, "Properties and performance of a center/surround retinex," *IEEE Transactions on Image Processing*, vol. 6, no. 3, pp. 451–462, March 1997.

[26] E. H. Land, "An alternative technique for the computation of the designator in the retinex theory of color vision," *Proceedings of the National Academy of Sciences*, vol. 83, no. 10, pp. 3078–3080, 1986.

[27] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *Proceedings of the 2017 IEEE international conference on image processing (ICIP)*, pp. 3645–3649, IEEE, Beijing, China, September 2017.

[28] A. Bewley, Z. Ge, and L. Ott, "Simple online and realtime tracking," in *Proceedings of the 2016 IEEE international conference on image processing (ICIP)*, pp. 3464–3468, IEEE, Beijing, China, September 2016.

[29] A. Paszke, S. Gross, F. Massa et al., "Pytorch: an imperative style, high-performance deep learning library," 2019, https://arxiv.org/abs/1912.01703.

[30] A. Kaehler and G. Bradski, *Learning OpenCV 3: Computer Vision in C++ with the OpenCV library*, O'Reilly Media, Inc., Newton, Massachusetts, United States, 2016.