Metro Passenger Flow Statistics Based on YOLOv3

Weiming Liu1*, Xiaorui Du1, Qinghua Geng1, Jingning, Li1, Haiyu, Li2, Lan Liu2
1School of Civil Engineering & Transportation, South China University of Technology, Guangzhou, Guangdong, 510641, China
2Guangzhou Metro Group Co., Ltd, Guangzhou, Guangdong, 510641, China
*Corresponding author’s e-mail: 564245988@qq.com

Abstract. Passenger flow real-time detection in per train compartment is the basis for realizing metro entrance flow restriction, platform waiting guidance and optimizing train departure interval. In this paper, we collect RGB images as the input of the improved YOLOv3 network, then we can get the statistics of passenger flow by detecting and tracking passengers’ heads. For the head detection part, this paper re-selects the number of anchors by using the k-means++ clustering method and removes the large-scale object detection module in the original YOLOv3 network. For the head tracking part, this paper proposes the probability gradient map, IOU-Max maximum matching algorithm and voting strategy, which make the stability of target tracking greatly enhanced. Compared with the original YOLOv3 network, our research results show that the accuracy of head detecting based on the improved YOLOv3 network increases from 90.1% to 97.84%, the recall rate increases from 85.4% to 94.7%, and the accuracy of the head tracking algorithm is as high as 98.87%, and the accuracy of passenger flow density detecting is as high as 95%. The measured detection speed reaches 47FPS on the TITAN X server, enabling real-time fast detection and tracking of targets.

1. Introduction
Metro has become the most important means of urban transportation because of its large passenger capability, high velocity and punctuality. But in recent years, metro trains are becoming more and more overloaded[1]. As a result, probability of trampling, stampede and falling off the track has increased, when huge flows of passenger crowd in the limited platform space and the enclosed Metro trains. Giving the mentioned hidden dangers, it is urgent to come up with a method to monitor the passenger flow density and residual load capacity of each train carriage. Thus, metro staffs can conduct passengers in a rational way and adjust the frequency of train departure.

Traditionally, there are 4 common ways of detection for passenger flow density, including video surveillance detection, weighing detection, RFID reader detection[2]. However, all of those 4 detection methods suffer from problems of low detection accuracy, slow detection speed and high calculation cost. Besides, there are also some statistical methods based on gate card swiping information. However, they do not have the ability of real-time detection and visualization, because the data is usually daily periodic. Additionally, they cannot give the number of passengers who are waiting to board at the metro station. Finally, they lack the ability to calculate the number of passengers onboard. All of those weaknesses have hindered the ability of metro staffs to adjust the departure frequency and to guide the passenger flow[3].

Therefore, this paper proposes a model based on deep learning, which can calculate the number of passengers onboard and passengers waiting to board at the metro station. The model uses images taken
by an RGB camera, which is mounted between the train door and the shield door and taking pictures from the top. First, our model detects and tracks passengers’ head in real time, and provides statistics of boarding passengers and alighting passengers. Then, combining the number of passengers entering and leaving the metro station, which could be easily obtained from the metro department, we can accurately calculate the number of passengers onboard and the number of passengers waiting to board at the metro station. The innovative points of our model are as follows: it 1) provides accurate real-time passenger flow data, 2) calculates the number of onboard passengers, 3) calculates the number of passengers waiting to board at the metro station. It will improve the effectiveness of the passenger guidance and platform organisation remarkably.

2. Improved YOLOv3 Network
The anchors in YOLOv3[4] network is based on COCO data set. Joseph Redmon get these anchors by K-means clustering method. Considering the dimension span of the object is large in COCO data, Joseph Redmon eventually chooses nine anchors as priori anchors, which means Joseph Redmon needs to calculate \((13*13+26*26+52*52)*3\) bounding boxes in each picture.

Metro head detection system needs to satisfy better real-time and accuracy, so this paper makes full use of the characteristics of data sets under specific circumstances, re-adjusts the YOLOv3 network, and minimizes the amount of network computation under the condition of meeting the detection accuracy. According to the statistics, the pixel range of head in the 640*480 picture is 60*60~120*120, so the 52*52 detection layer in the original YOLOv3 network for small target detection is discarded, and the selection of anchor number is re-clustered by using k-means+ according to the size of the actual detection target. As is well-known, proper initialization of k-means is crucial for obtaining a good final solution. K-means++ initialization algorithm achieves this, obtaining an initial set of centers that is probably close to the optimum solution[5]. The results are shown in table 1. It can be seen from the table that with the number of clusters: With the increase of IOU, the average IOU tends to increase, but the increase is slow. Considering the amount of network computation, the clustering number is 4, and the pre-selected anchors' box size is \((71,63)(95,88)(110,92)(137,118)\). Therefore, the number of prediction boxes that need to be regressed for each forward propagation of the revised network is \((13*13+26*26)*2\), which is much less than the original network. The improved YOLOv3 network is shown in Figure 1.

| The number of clusters | 4   | 6   | 9   |
|------------------------|-----|-----|-----|
| The average value of IOU| 0.8577 | 0.8663 | 0.8712 |

2.1 Network Evaluation
In this paper, we collect 9000 original pictures form different metro stations in different times. In order to get more training data and increase the network's generalization ability, this paper flips a number of original pictures randomly, increases contrast and noise points, and finally obtains 20,000 pictures, randomly selects 18,000 pictures as a training data set and the remaining 2,000 pictures as network evaluation test set. The final evaluation results are shown in table 2.
3. Passenger Flow Statistics

3.1. Object Detection
Traditional inspection methods often adopt Haar features and Adaboost algorithm to implement the detection of the human's head through OpenCV[6]. However, those methods cannot carry out the complicated metro environment.

Passenger flow analysis based on the RGB camera can be divided into two steps:
- Step 1: Complete the detection of a human head;
- Step 2: Complete the tracking of the human head;

In most cases, the improved YOLOv3 can detect the human head well, but it cannot analyze the passengers' boarding and alighting behavior and lock the tracking target.

3.2. Object Tracking
There are two main modes of traditional target tracking: 1. Target tracking without prior knowledge, that is, extracting moving objects and background from ordered images; 2. Target tracking based on prior knowledge, that is, extracting moving objects by modeling first and then matching them best.

### Table 2. Network Evaluation

| Name            | Number of false detection | Accuracy   | Number of miss detection | Recall   |
|-----------------|----------------------------|------------|--------------------------|----------|
| YOLOv3          | 44                         | 90.10%     | 102                      | 85.40%   |
| Improved YOLOv3 | 10                         | 97.84%     | 38                       | 94.70%   |

![Figure 1. Improved YOLOv3 Network](image-url)
Common target tracking algorithms include Euclidean distance based similarity measurement algorithm, gradient-based Meanshift algorithm, particle filter based on Monte Carlo, and Bayesian theory. Table 3 lists four traditional target tracking algorithms and their shortcomings.

| Traditional Algorithm                          | Shortcomings                                                      |
|-----------------------------------------------|-------------------------------------------------------------------|
| The active contour[7]                         | Cannot track fast moving targets                                  |
| Feature-based tracking algorithm              | Unable to cope with a complex environment and weak robustness    |
| Region Tracking Algorithms[8]                 | Low accuracy                                                     |
| Model-based tracking[9]                       | The computation speed is slow, and the real-time performance is poor. |

### 3.3. IOU-Max Matching Tracking

In this paper, an IOU-Max matching tracking algorithm based on YOLOv3 is proposed. Its main function is to judge passengers’ boarding and disembarking behaviors and to realize real-time tracking of passengers on the following trains. In this paper, the top-mounted camera is used to collect the video stream. The installation schematic of the camera is presented in Figure 2, and Table 4 demonstrates the installation information.

From the table 4, it is clear that the frame rate of the video is 30 frames per second (i.e., 0.033 s per frame), and the average moving distance of adults is about 1m/s. So the moving distance of the target between two consecutive frames is 3.3 cm, reflecting that the moving pixel range of two consecutive frames in the 640*480 image is 9.48-7.44, of which 9.48 corresponds to passengers of 1.8 m height, and 7.44 corresponds to passengers of 1.2 m height, so the moving distance between two consecutive frames is 9.48-7.44. There must be overlapping regions in the target detection frame. Figure3 to Figure5 shows three consecutive frame detection pictures. Figure6 shows the overlap of Figure3 and Figure 4. Figure7 shows the overlap between Figure4 and Figure5. Figure8 shows the overlap of Figure3, Figure4 and Figure5. It is easy to see the position correlation of the target in the continuous multi-frame image from the following figure.
In order to improve the stability and real-time performance of the target tracking algorithm, this paper abandons the traditional methods of moving object detection, such as k-means, optical flow and other algorithms which need a lot of computational resources\[10\]. Further, it makes full use of the context information of video stream, and achieves the goal of continuous tracking and locking target by one target frame calibration. The final target frame will have three attributes: 1. up and down attributes; 2. the probability that the target will disappear in the next frame; 3. the unique ID number. Firstly, the algorithm uses prior knowledge to delineate the upstream and downstream regions, as shown in Fig. 9. If the target is detected in the up area for the first time, the target box will get the up markers and vice versa.

According to the prior knowledge, this paper proposed the probability gradient map of target disappearance. The probability of target disappearance is 10%, 30%, 50%, 70%, and 90% inside out, as shown in Figure 10. The higher the probability of target located in the next frame, the higher the probability of target disappearing in the next frame, and vice versa, the smaller the probability of target disappearing in the next frame. Usually, the target box will span two or more probabilistic vanishing regions. At this time, the vanishing probability of the target box is its weighted average value. Its value determines whether the target box assigns the unique ID number and whether the missing head operation is carried out. The calculation is as follows:

\[
P = \frac{\sum Area_i \cdot P_i}{\sum Area_i}
\]

\[
Area_i : \text{Area of target box in different probability intervals}
\]

\[
P_i : \text{Probability of target box falling in different probability intervals}
\]

Step2: Define old and new target boxes.

When the probability of disappearance of the next frame is less than 85%, the new target box and the old target box will be defined in this paper. As shown in Figure 11, the decision rules are as follows:

New target box: If the maximum IOU of a target box in the current frame (e.g., box 5) is less than 0.3 compared with any other boxes in the previous frame, then this target box is defined as a new target box.
Old target box: If the maximum IOU of the target box in the current frame (e.g., box 3, 4) is greater than 0.3 compared with some box (e.g., box 1, 2, respectively) in the previous frame, then it is recognized as an old target box, inheriting the whole attributes from that corresponding box. The definition of IOU is as follows:

\[
IoU = \frac{O_A}{U_A} = \frac{\text{Overlap area of target box}}{\text{Union area of target box}}
\]

In this algorithm, if it is the new target box, the algorithm will assign a new follow ID to it, and it will obtain the attributes of getting on the train or off the train, if it is the old target box, it will inherit all the attributes of the corresponding box in the previous frame.

The whole algorithm includes three stages: the emergence of new targets, the follow-up of old targets, and the disappearance of old targets. It shows the whole process from the opening of train shield doors to the closing of train shield doors, and accurately records passengers’ boarding and disembarking. Among them, the number of ID numbers allocated represents the total passenger flow of the train, and the number of target boxes with up and down markers represents the number of up and down passengers, respectively. The diagram is shown in Figure 12 to Figure 17.

To sum up, the improved YOLOV3 algorithm has three advantages: 1. fast computing speed; 2. stable target tracking; 3. Target detection using gradient probability distribution map can effectively filter false detection.

3.4. Special case processing

In the research process, the ideal passenger boarding and disembarking rules cannot fully meet the actual passenger boarding and disembarking situation. For example, because of the complexity of the metro environment, YOLOV3 cannot detect 100% of the target box. It occurs when a certain target
Given the causes of misdetection, this paper uses prior knowledge and gradient probability map to complete the location information of missing detection target frame according to the incomplete information provided by YOLOv3. As can be seen from the previous paper, the moving position of the target between two consecutive frames is limited (< 20 pixels), and the probability of disappearance of the target frame at different positions in the gradient probability map is different. Therefore, the target lost position will be searched by enlarging the region, and the lost target can be found by template matching. The improved head detection and tracking flow chart is shown in Figure 18.

The traditional Euclidean distance matching algorithm cannot effectively use pixel information and has low stability. Therefore, this paper will use the improved Euclidean distance for template matching, which is calculated as follows:

\[
\begin{align*}
    r &= \frac{c_{1R} + c_{2R}}{2} \\
    \Delta R &= c_{1R} - c_{2R} \\
    \Delta G &= c_{1G} - c_{2G} \\
    \Delta B &= c_{1B} - c_{2B} \\
    \Delta C &= \sqrt{(2 + \frac{r}{256}) \cdot \Delta R^2 + 4 \cdot \Delta G^2 + (2 + \frac{255-r}{256}) \cdot \Delta B^2}
\end{align*}
\]

- \( r \): Mean value of the red component of the two graphs
- \( \Delta R \): The difference of red components between two graphs
- \( \Delta G \): The difference of green components between two graphs
- \( \Delta B \): The difference of blue components between two graphs
- \( \Delta C \): Improvement of Euclidean distance value of the two graphs
This group of equations is a combination of weighted Euclidean distances, similar to LUV[12], but with higher stability, it will not become extremely sensitive due to changes in the environment. The execution effect of the activation target missed detection processing flow algorithm is shown in Fig. 19, Fig. 20, and Fig. 21 as follows:
Figure 19. The first and third frames are normally detected, the second frame is missed, and the missed detection algorithm is activated.

Figure 20. Implement missing target search algorithm to reconstruct the failed target.

Figure 21. Target normal display at the terminal.

By counting the detection information of one door of 1000 trains, the test results of the target missed detection processing flow and the non-target missed detection flow are compared in Table 5:

| Misdetection process | The number of tracking | False tracking | Tracking accuracy | Processing speed |
|----------------------|------------------------|----------------|-------------------|-----------------|
| with                 | 9847                   | 111            | 98.87%            | 47FPS           |
| without              | 9847                   | 612            | 93.78%            | 48FPS           |

Through the table, it can be seen that the accuracy of target tracking can be improved from 93.78% to 98.87% with the target missed detection process while ensuring that the processing speed is not reduced, and the target missed detection process also can improve the stability and reliability of the algorithm.

3.5. Passenger flow statistics

After target detection, target tracking, and special case processing, the passenger flow of Metro can be analyzed statistically. The passenger carrying the status of trains and passenger detention in metro stations can be explained by the formula in table 6 below:

Table 6. Passenger flow statistics

| Statistic               | Calculation formula |
|-------------------------|---------------------|
| Total Passenger Flow of a Train $P_{\text{total}}$ | $P_{\text{total}} = P_{\text{up}} + P_{\text{down}} + P_{\text{i\_gate}} + P_{\text{o\_gate}}$ |
Number of passengers boarding from the first train leaving the station to the second train leaving the station $P_{up}$ 

The number of passengers getting off the train from the first train leaving the station to the second train leaving the station $P_{down}$.

Number of passengers entering the gate from the first train departure to the second train departure $P_{i\_gate}$

Number of passengers departing from the first train to the second train during departure $P_{o\_gate}$

Number of passengers detained at a metro station between the departure of the first train and the departure of the second train $P_{remain}$

Number of passengers on board $P_{passenger}$

\[ P_{up} = \sum_{i_{up\_new}=F_{end}}^{i_{up\_new}=F_{start}} T_{i_{up\_new}} \]
\[ P_{down} = \sum_{i_{down}=F_{start}}^{i_{down}=F_{end}} T_{i_{down}} \]
\[ P_{i\_gate} = \sum_{i_{gate}=F_{start}}^{i_{gate}=F_{end}} T_{i_{gate}} \]
\[ P_{o\_gate} = \sum_{i_{gate}=F_{start}}^{i_{gate}=F_{end}} T_{o_{gate}} \]
\[ P_{remain} = P_{i\_gate} + P_{down} - P_{up} - P_{o\_gate} \]
\[ P_{passenger} = P_{pre\_passenger} + P_{up} - P_{down} \]

$T_{i_{up\_new}}$: The number of new upstream ID boxes is obtained in each frame from the first train departure to the second train departure.

$T_{i_{down}}$: The number of new downstream ID boxes is obtained in each frame from the first train departure to the second train departure.

$T_{i_{gate}}$: The number of passengers entering per second during the departure of the first train from the station to the second train

$T_{o_{gate}}$: The number of passengers departures per second during the departure of the first train from the station to the second train

$P_{pre\_passenger}$: Number of passengers on this train at the last stop.

$F_{start}$: Frame of the start of the first train leaving the station.

$F_{end}$: Frame of the end of the second train leaving the station.

$T_{start}$: Time of the start of the first train leaving the station (seconds).

$T_{end}$: Time of the end of the second train leaving

Through the formulas in the above tables, we can accurately count the number of passengers on trains and the number of passengers detained in metro stations in real time, and feedback them to the Metro staff in time. It is beneficial to the metro staff to effectively channel the passenger flow and dispatch the train during the peak period, which improves the operating efficiency of the metro.

### 4. Conclusion

In this paper, we detected passengers’ head by the improved YOLOv3 network. Then, we tracked passengers’ head by proposed IOU-Max matching algorithm. Through the detection and trace of passengers’ head, we realized the goal of calculating the passenger flow density and residual load capacity of each train carriage. Additionally, we proposed probability gradient map and voting strategy. Those techniques made our algorithm more adaptable and much more robust to different complex metro situations.
Compared with the original YOLOv3 network, the accuracy of target detection based on the improved YOLOv3 network increased from 90.1% to 97.84%, and the recall rate increased from 85.4% to 94.7%. As for the object tracking, the accuracy of the IOU-Max matching algorithm was as high as 98.87%, and the accuracy of passenger flow density detection reached 95%. All in all, the measured detection speed reached 47FPS on the TITAN X server, enabling real-time fast detecting and tracking of targets. Given the high accuracy and real-time performance of detecting and tracking, our model can lay a foundation of passenger flow guidance. Therefore, it has a very high use and promotion value in the field of rail transit.

Funding projects: Supported by the 13th Five-Year National Key Research and Development Program (2016YFB1200402), and the compilation task of Guangdong High-end Equipment Manufacturing Industry Standard Project "Foreign Object Detection System between Metro Platform Screen Door and Train" in 2015.

References
[1] Jiang C S, Deng Y F, Hu C, et al. Crowding in platform staircases of a subway station in China during rush hours[J]. Safety science, 2009, 47(7): 931-938.
[2] Smith J R, Fishkin K P, Jiang B, et al. RFID-based techniques for human-activity detection[J]. Communications of the ACM, 2005, 48(9): 39-44.
[3] Smith J R, Fishkin K P, Jiang B, et al. RFID-based techniques for human-activity detection[J]. Communications of the ACM, 2005, 48(9): 39-44.
[4] Redmon J, Farhadi A. Yolov3: An incremental improvement[J]. arXiv preprint arXiv:1804.02767, 2018.
[5] Bahmani B, Moseley B, Vattani A, et al. Scalable k-means++[J]. Proceedings of the VLDB Endowment, 2012, 5(7): 622-633.
[6] Chen Y, Zhang L, Wang J. Passenger detection for subway transportation based on video[C]//2014 10th International Conference on Natural Computation (ICNC). IEEE, 2014: 720-724.
[7] Bresson X, Esedoḡlu S, Vandergheynst P, et al. Fast global minimization of the active contour/snake model[J]. Journal of Mathematical Imaging and vision, 2007, 28(2): 151-167.
[8] Basele B, Deriche R. Region tracking through image sequences[C]//Proceedings of IEEE International Conference on Computer Vision. IEEE, 1995: 302-307.
[9] Koller D, Daniilidis K, Nagel H H. Model-based object tracking in monocular image sequences of road traffic scenes[J]. International Journal of Computer 11263on, 1993, 10(3): 257-281.
[10] Guan W, Chen X, Huang M, et al. High-speed robust dynamic positioning and tracking method based on visual visible light communication using optical flow detection and bayesian forecast[J]. IEEE Photonics Journal, 2018, 10(3): 1-22.
[11] Gulati P, Yadav M. A Novel Approach for Extracting Pertinent Keywords for Web Image Annotation Using Semantic Distance and Euclidean Distance[M]//Software Engineering. Springer, Singapore, 2019: 173-183.
[12] Kekre H B, Thepade S D. Improving color to gray and back'using kekre's luv color space[C]//2009 IEEE International Advance Computing Conference. IEEE, 2009: 1218-1223.