

INT 248

Lovely Professional University, Jalandhar

VEHICLE DETECTION AND COUNTING NUMBER OF IT



ADVANCED MACHINE LEARNING

DEPARTMENT OF INTELLIGENT SYSTEMS

SCHOOL OF COMPUTER SCIENCE ENGINEERING

Name: Praveen L

Reg. no: 11902378

Section: KM056

Roll no: A - 04

STUDENT DECLARATION

This is to declare that this report has been written by me/us. No part of the report is copied from other sources. All information included from other sources have been duly acknowledged. I/We aver that if any part of the report is found to be copied, I/we are shall take full responsibility for it.

Praveen L

A – 04

11902378

GitHub Link - <https://github.com/praveen9626/vehicle-detection>

TABLE OF CONTENTS		
S.NO	TITLE	PAGE NO.
1	Introduction	4
1.1	Related works of vehicle detection	4
1.2	Related works of vehicle tracking	6
2	Description of Project	7
2.1	Vehicle Dataset	7
2.2	System Structure	8
2.3	Methods	9
3	Results and Discussion	13
4	Conclusion	15

INTRODUCTION

Intelligent vehicle detection and counting are becoming increasingly important in the field of highway management. However, due to the different sizes of vehicles, their detection remains a challenge that directly affects the accuracy of vehicle counts. To address this issue, this paper proposes a vision-based vehicle detection and counting system. A new high definition highway vehicle dataset with a total of 57,290 annotated instances in 11,129 images is published in this study. Compared with the existing public datasets, the proposed dataset contains annotated tiny objects in the image, which provides the complete data foundation for vehicle detection based on deep learning. In the proposed vehicle detection and counting system, the highway road surface in the image is first extracted and divided into a remote area and a proximal area by a newly proposed segmentation method; the method is crucial for improving vehicle detection. Then, the above two areas are placed into the YOLOv3 network to detect the type and location of the vehicle. Finally, the vehicle trajectories are obtained by the ORB algorithm, which can be used to judge the driving direction of the vehicle and obtain the number of different vehicles. Several highway surveillance videos based on different scenes are used to verify the proposed methods. The experimental results verify that using the proposed segmentation method can provide higher detection accuracy, especially for the detection of small vehicle objects. Moreover, the novel strategy described in this article performs notably well in judging driving direction and counting vehicles. This paper has general practical significance for the management and control of highway scenes.

Vehicle detection and statistics in highway monitoring video scenes are of considerable significance to intelligent traffic management and control of the highway. With the popular installation of traffic surveillance cameras, a vast database of traffic video footage has been obtained for analysis. Generally, at a high viewing angle, a more-distant road surface can be considered. The object size of the vehicle changes greatly at this viewing angle, and the detection accuracy of a small object far away from the road is low. In the face of complex camera scenes, it is essential to effectively solve the above problems and further apply them. In this article, we focus on the above issues to propose a viable solution, and we apply the vehicle detection results to multi-object tracking and vehicle counting.

RELATED WORK ON VEHICLE DETECTION

At present, vision-based vehicle object detection is divided into traditional machine vision methods and complex deep learning methods. Traditional machine vision methods use the motion of a vehicle to separate it from a fixed background image. This method can be divided into three categories: the method of using background subtraction, the method of using continuous video frame difference, and the method of using optical flow. Using the video frame difference method, the variance is calculated according to the pixel values of two or three consecutive video frames. Moreover, the moving foreground region is separated by the threshold. By using this method and suppressing noise, the stopping of the vehicle can also be detected. When the background image in the video is fixed, the background information is used

to establish the background model. Then, each frame image is compared with the background model, and the moving object can also be segmented. The method of using optical flow can detect the motion region in the video. The generated optical flow field represents each pixel's direction of motion and pixel speed. Vehicle detection methods using vehicle features, such as the Scale Invariant Feature Transform (SIFT) and Speeded Up Robust Features (SURF) methods, have been widely used. For example, 3D models have been used to complete vehicle detection and classification tasks. Using the correlation curves of 3D ridges on the outer surface of the vehicle, the vehicles are divided into three categories: cars, SUVs, and minibuses.

The use of deep convolutional networks (CNNs) has achieved amazing success in the field of vehicle object detection. CNNs have a strong ability to learn image features and can perform multiple related tasks, such as classification and bounding box regression. The detection method can be generally divided into two categories. The two-stage method generates a candidate box of the object via various algorithms and then classifies the object by a convolutional neural network. The one-stage method does not generate a candidate box but directly converts the positioning problem of the object bounding box into a regression problem for processing. In the two-stage method, Region-CNN (R-CNN) uses selective region search in the image. The image input to the convolutional network must be fixed-size, and the deeper structure of the network requires a long training time and consumes a large amount of storage memory. Drawing on the idea of spatial pyramid matching, SPP NET allows the network to input images of various sizes and to have fixed outputs. R-FCN, FPN, and Mask RCNN have improved the feature extraction methods, feature selection, and classification capabilities of convolutional networks in different ways. Among the one-stage methods, the most important are the Single Shot Multibox Detector (SSD) and You Only Look Once (YOLO) frameworks. The MutiBox , Region Proposal Network (RPN) and multi-scale representation methods are used in SSD, which uses a default set of anchor boxes with different aspect ratios to more accurately position the object. Unlike SSD, the YOLO network divides the image into a fixed number of grids. Each grid is responsible for predicting objects whose centre points are within the grid. YOLOv2 added the BN (Batch Normalization) layer, which makes the network normalize the input of each layer and accelerate the network convergence speed. YOLOv2 uses a multi-scale training method to randomly select a new image size for every ten batches. Our vehicle object detection uses the YOLOv3 network. Based on YOLOv2, YOLOv3 uses logistic regression for the object category. The category loss method is two-class cross-entropy loss, which can handle multiple label problems for the same object. Moreover, logistic regression is used to regress the box confidence to determine if the IOU of the a priori box and the actual box is greater than 0.5. If more than one priority box satisfies the condition, only the largest prior box of the IOU is taken. In the final object prediction, YOLOv3 uses three different scales to predict the object in the image.

The traditional machine vision method has a faster speed when detecting the vehicle but does not produce a good result when the image changes in brightness, there is periodic motion in the background, and where there are slow moving vehicles or complex scenes. Advanced CNN has achieved good results in object detection; however, CNN is sensitive to scale changes in object detection. The one stage method uses grids to predict objects, and the grid's spatial constraints make it impossible to have higher precision with the two-stage approach, especially for small objects. The two stage method uses region of interest pooling to segment candidate regions into blocks according to given parameters, and if the candidate region is smaller than the size of the

given parameters, the candidate region is padded to the size of the given parameters. In this way, the characteristic structure of a small object is destroyed and its detection accuracy is low. The existing methods do not distinguish if large and small objects belong to the same category. The same method is used to deal with the same type of object, which will also lead to inaccurate detection. The use of image pyramids or multi-scale input images can solve the above problems, although the calculation requirements are large.

RELATED WORK ON VEHICLE TRACKING

Advanced vehicle object detection applications, such as multi-object tracking, are also a critical ITS task. Most multi-object tracking methods use Detection-Based Tracking (DBT) and Detection-Free Tracking (DFT) for object initialization. The DBT method uses background modeling to detect moving objects in video frames before tracking. The DFT method needs to initialize the object of the tracking but cannot handle the addition of new objects and the departure of old objects. The Multiple Object Tracking algorithm needs to consider the similarity of intra-frame objects and the associated problem of inter-frame objects. The similarity of intra-frame objects can use normalized cross-correlation (NCC). The Bhattacharyya distance is used to calculate the distance of the colour histogram between the objects, such as in. When inter-frame objects are associated, it is necessary to determine that an object can only appear on one track and that one track can only correspond to one object. Currently, detection-level exclusion or trajectory-level exclusion can solve this problem. To solve the problems caused by scale changes and illumination changes of moving objects, used SIFT feature points for object tracking, although this is slow. The ORB feature point detection algorithm is proposed for use in this work. ORB can obtain better extraction feature points at a significantly higher speed than SIFT.

In summary, it can be considered that the method of vehicle object detection has been transferred from research on traditional methods to that on deep convolutional network methods. Moreover, there are fewer public datasets for specific traffic scenes. The sensitivity of convolutional neural networks to scale changes makes small object detection inaccurate. It is challenging to conduct multi-object tracking and subsequent traffic analysis when highway surveillance cameras are used. In summary, our contributions include the following:

1. A large-scale high definition dataset of highway vehicles is established that can provide many different vehicle objects fully annotated under various scenes captured by highway surveillance cameras. The dataset can be used to evaluate the performance of many vehicle detection algorithms when dealing with vehicle scale changes.
2. A method for detecting small objects in highway scenes is used to improve vehicle detection accuracy. The highway road surface area is extracted and divided into the remote area and the proximal area, which are placed into the convolution network for vehicle detection.
3. A multi-object tracking and trajectory analysis method is proposed for highway scenes. The detection object feature points are extracted and matched by the ORB algorithm, and the road detection line is determined to count the vehicle movement direction and the traffic flow.

This research will be described in the following sections. “Vehicle dataset” section introduces the vehicle dataset used in this paper. “The system structure” section introduces the overall process of the proposed system. “Methods” section describes our strategy in detail. “Results and discussion” section presents the experiments and related analysis. “Conclusions” section summarizes the entire article.

VEHICLE DATASET

Surveillance cameras in roads have been widely installed worldwide but traffic images are rarely released publicly due to copyright, privacy, and security issues. From the image acquisition point of view, the traffic image dataset can be divided into three categories: images taken by the car camera, images taken by the surveillance camera, and images taken by non-monitoring cameras. The KITTI benchmark dataset contains images of highway scenes and ordinary road scenes used for automatic vehicle driving and can solve problems such as 3D object detection and tracking. The Tsinghua-Tencent Traffic-Sign Dataset has 100,000 images from car cameras covering various lighting conditions and weather conditions, although no vehicles are marked. The Stanford Car Dataset is a vehicle dataset taken by non-monitoring cameras with a bright vehicle appearance. This dataset includes 19,618 categories of vehicles covering the brands, models, and production years of the vehicles. The Comprehensive Cars Dataset is similar to the Stanford Car Dataset but contains many pictures. The 27,618 images include the vehicle’s maximum speed, number of doors, number of seats, displacement, and car type. The 136,727 images include the overall appearance of the vehicle. The datasets are taken by surveillance cameras; an example is the BIT-Vehicle Dataset, which contains 9,850 images. This dataset divides the vehicle into six types: SUV, sedan, minivan, truck, bus, and micro-bus; however, the shooting angle is positive, and the vehicle object is too small for each image, which is difficult to generalize for CNN training. The Traffic and Congestions (TRANCOS) dataset contains pictures of vehicles on highways captured by surveillance cameras and contains a total of 1,244 images. Most of the images have some occlusion. This dataset has a small number of pictures, and no vehicle type is provided, which makes it less applicable. Therefore, few datasets have useful annotations, and few images are available in traffic scenes.

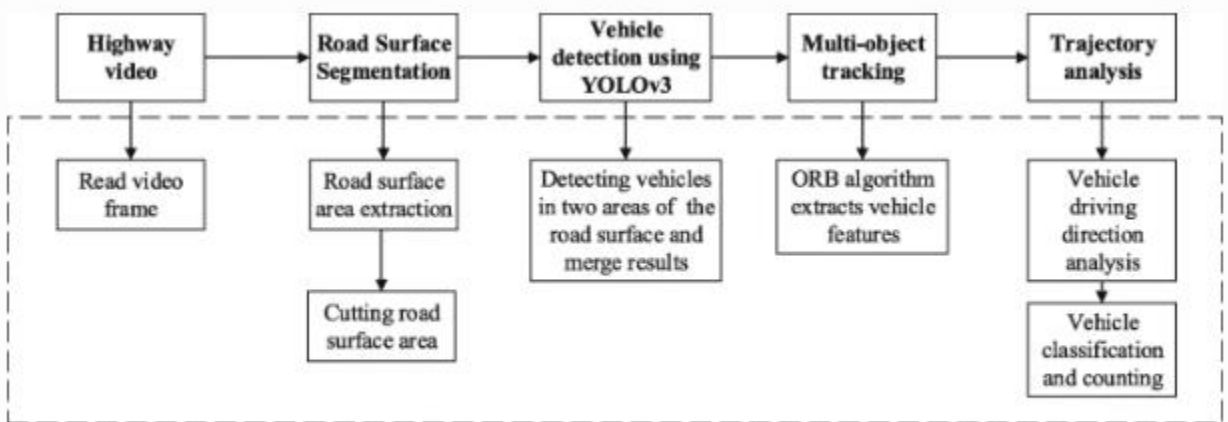
This section introduces the vehicle dataset from the perspective of the highway surveillance video we produced. The dataset has been published in: http://drive.google.com/open?id=1li858elZvUgss8rC_yDsb5bDfiRyhdx. The dataset picture is from the highway monitoring video of Hangzhou, China (Fig. 1). The highway monitoring camera was installed on the roadside and erected at 12 meters; it had an adjustable field of view and no present position. The images from this perspective cover the far distance of the highway and contains vehicles with dramatic changes in scale. The dataset images were captured from 23 surveillance cameras for different scenes, different times, and different lighting conditions. This dataset divides the vehicles into three categories: cars, buses, and trucks. The label file is stored in a text document that contains the numeric code of the object category and the normalized coordinate value of the bounding box. As shown in Table 1, this dataset has a total of 11,129 images and 57,290 annotation boxes.. The images have an RGB format and 1920*1080 resolution. Note that we annotated the smaller objects in the proximal road area, and the dataset thus contains vehicle objects with massive scale changes. An annotated instance near the camera

has more features, and an instance far from the camera has few features. Annotated instances of different sizes are beneficial to the improvement of the detection accuracy of small vehicle objects. This dataset is divided into two parts: a training set and a test set. In our dataset, cars accounted for 42.17%, buses accounted for 7.74%, and trucks accounted for 50.09%. There are 5.15 annotated instances in each image on average. Figure 3 compares the difference between the number of annotated instances in our dataset and the PASCAL VOC, ImageNet, and COCO datasets. Our dataset is a universal dataset for vehicle targets that can be used in a variety of areas, such as Europe. Compared with the existing vehicle datasets, our dataset has a large number of high definition images, sufficient lighting conditions, and complete annotations.

SYSTEM STRUCTURE

This section describes the main structure of the vehicle detection and counting system. First, the video data of the traffic scene are entered. Then, the road surface area is extracted and divided. The YOLOv3 deep learning object detection method is used to detect the vehicle object in the highway traffic scene. Finally, ORB feature extraction is performed on the detected vehicle box to complete multi-object tracking and obtain vehicle traffic information.

The road surface segmentation method is used to extract the road area of the highway. The road area is divided into two parts based on the position where the camera is erected, a remote area and a proximal area. Then, the vehicles in the two road areas are detected using the YOLOv3 object detection algorithm. This algorithm can improve the small object detection effect and solve the problem that the object is difficult to detect due to the sharp change of the object scale. The ORB algorithm is then used for multi-object tracking. The ORB algorithm extracts the detected box's features and matches them to achieve correlation between the same object and different video frames. Finally, traffic statistics are calculated. The trajectory generated by the object tracking is generated, the vehicle driving direction is determined, and traffic information such as the number of vehicles in each category is collected. This system improves the accuracy of object detection from the highway surveillance video perspective and constructs a detection tracking and traffic information acquisition plan within the full field of the camera view.

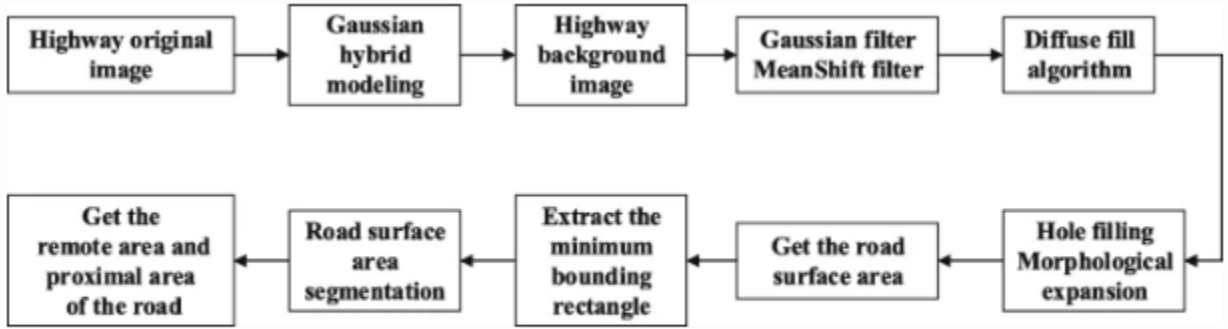


Overall flow of the method

METHODS

ROAD SURFACE SEGMENTATION

This section describes the method of highway road surface extraction and segmentation. We implemented surface extraction and segmentation using image processing methods, such as Gaussian mixture modelling, which enables better vehicle detection results when using the deep learning object detection method. The highway surveillance video image has a large field of view. The vehicle is the focus of attention in this study, and the region of interest in the image is thus the highway road surface area. At the same time, according to the camera's shooting angle, the road surface area is concentrated in a specific range of the image. With this feature, we could extract the highway road surface areas in the video.



Overall flow of road surface extraction

To eliminate the influence of vehicles on the road area segmentation, we used the Gaussian mixture modeling method to extract the background in the first 500 frames of the video. The value of the pixel in the image is Gaussian around a certain central value in a certain time range, and each pixel in each frame of the image is counted. If the pixel is far from the centre, the pixel belongs to the foreground. If the value of the pixel point deviates from the centre value within a certain variance, the pixel point is considered to belong to the background. The mixed Gaussian model is especially useful in images where background pixels have multi-peak characteristics such as the highway surveillance images used in this study.

After extraction, the background image is smoothed by a Gaussian filter with a 3*3 kernel. The MeanShift algorithm is used to smooth the colour of the input image, neutralize the colour with a similar colour distribution, and erode the colour area with a smaller area. On this basis, the flooding filling algorithm is used to separate the road surface area. The flooding filling algorithm selects a point in the road surface area as a seed point and fills the adjacent continuous road surface areas with the pixel value of the seed point. The pixel value of the adjacent continuous road surface areas is close to the seed point pixel value. Finally, the hole filling and morphological expansion operations are performed to more completely extract the road surface. We extracted the road surfaces of different highway scenes, and the results.

We segmented the road surface area to provide accurate input for subsequent vehicle detection. For the extracted road surface image, a minimum circumscribed rectangle is generated for the image without rotation. The processed image is divided into five equal parts, the 1/5 area adjacent to the origin of the coordinate axis is defined as the near remote area of the road surface, and the remaining 4/5 area is defined as the near proximal area of the road surface. The near proximal area and the near remote area overlap by 100 pixels to address the problem that the vehicle in the image may be divided into two parts by the above procedure. The pixel values of the near proximal area and the near remote area are searched column by column. If the pixel values in the column are all zero, the image of the column is all black and is not the road surface area; it is then deleted. After the not-road-surface areas are excluded, the reserved areas are called remote areas and proximal areas of the road surface.

VEHICLE DETECTION USING YOLOv3

This section describes the object detection methods used in this study. The implementation of the highway vehicle detection framework used the YOLOv3 network. The YOLOv3 algorithm continues the basic idea of the first two generations of YOLO algorithms. The convolutional neural network is used to extract the features of the input image. According to the size of the feature map, such as 13*13, the input image is divided into 13*13 grids. The centre of the object label box is in a grid unit, and the grid unit is responsible for predicting the object. The network structure adopted by YOLOv3 is called Darknet-53. This structure adopts the full convolution method and replaces the previous version of the direct-connected convolutional neural network with the residual structure. The branch is used to directly connect the input to the deep layer of the network. Direct learning of residuals ensures the integrity of image feature information, simplifies the complexity of training, and improves the overall detection accuracy of the network. In YOLOv3, each grid unit will have three bounding boxes of different scales for one object. The candidate box that has the largest overlapping area with the annotated box will be the final prediction result. Additionally, the YOLOv3 network has three output scales, and the three scale branches are eventually merged. Shallow features are used to detect small objects, and deep features are used to detect large objects; the network can thus detect objects with scale changes. The detection speed is fast, and the detection accuracy is high. Traffic scenes taken by highway surveillance video have good adaptability to the YOLOv3 network. The network will finally output the coordinates, confidence, and category of the object.

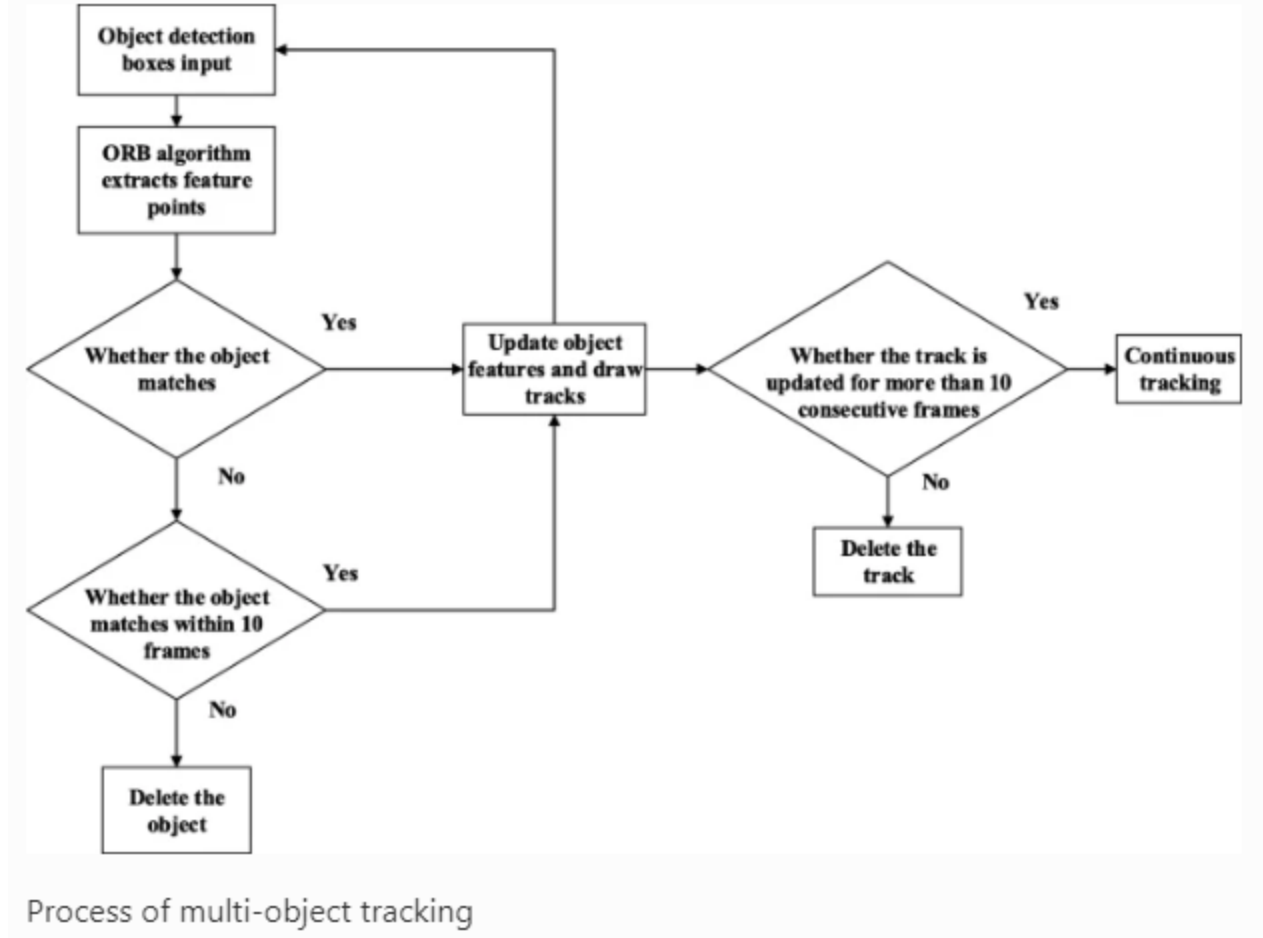
When using YOLO detection, images are resized to the same size, such as 416*416, when they are sent to the network. Since the image is segmented, the size of the remote road surface becomes deformed and larger. Therefore, more feature points of a small vehicle object can be acquired to avoid the loss of some object features due to the vehicle object being too small. The dataset presented in “[Vehicle dataset](#)” section is placed into the YOLOv3 network for training, and the vehicle object detection model is obtained. The vehicle object detection model can detect three types of vehicles: cars, buses, and trucks. Because there are few motorcycles on the highway, they were not included in our detection. The remote area and proximal area of the road surface are sent to the network for detection. The detected vehicle box positions of the two areas are mapped back to the original image, and the correct object position is obtained in the original image. Using the vehicle object detection method for obtaining the category and location of the

vehicle can provide necessary data for object tracking. The above information is sufficient for vehicle counting, and the vehicle detection method thus does not detect the specific characteristics of the vehicle or the condition of the vehicle.

MULTI-OBJECT TRACKING

This section describes how multiple objects are tracked based on the object box detected in “Vehicle detection using YOLOv3” section. In this study, the ORB algorithm was used to extract the features of the detected vehicles, and good results were obtained. The ORB algorithm shows superior performance in terms of computational performance and matching costs. This algorithm is an excellent alternative to the SIFT and SURF image description algorithms. The ORB algorithm uses the Features From Accelerated Segment Test (FAST) to detect feature points and then uses the Harris operator to perform corner detection. After obtaining the feature points, the descriptor is calculated using the BRIEF algorithm. The coordinate system is established by taking the feature point as the centre of the circle and using the centroid of the point region as the x-axis of the coordinate system. Therefore, when the image is rotated, the coordinate system can be rotated according to the rotation of the image, and the feature point descriptor thus has rotation consistency. When the picture angle is changed, a consistent point can also be proposed. After obtaining the binary feature point descriptor, the XOR operation is used to match the feature points, which improves the matching efficiency.

When the number of matching points obtained is greater than the set threshold, the point is considered to be successfully matched and the matching box of the object is drawn. The source of the prediction box is as follows: feature point purification is performed using the RANSAC algorithm, which can exclude the incorrect noise points of the matching errors, and the homography matrix is estimated. According to the estimated homography matrix and the position of the original object detection box, a perspective transformation is performed to obtain a corresponding prediction box.



We used the ORB algorithm to extract feature points in the object detection box obtained by the vehicle detection algorithm. The object feature extraction is not performed from the entire road surface area, which dramatically reduces the amount of calculation. In object tracking, the prediction box of the object in the next frame is drawn since the change of the vehicle object in the continuous frame of the video is subtle according to the ORB feature extracted in the object box. If the prediction box and the detection box of the next frame meet the shortest distance requirement of the centre point, the same object successfully matches between the two frames. We define a threshold T that refers to the maximum pixel distance of the detected centre point of the vehicle object box, which moves between two adjacent video frames. The positional movement of the same vehicle in the adjacent two frames is less than the threshold T . Therefore, when the centre point of the vehicle object box moves over T in the two adjacent frames, the cars in the two frames are not the same, and the data association fails. Considering the scale change during the movement of the vehicle, the value of the threshold T is related to the size of the vehicle object box. Different vehicle object boxes have different thresholds. This definition can meet the needs of vehicle movement and different input video sizes. T is calculated, in which *box height* is the height of the vehicle object box.

We delete the trajectory that is not updated for ten consecutive frames, which is suitable for the camera scene with a wide-angle of image collection on the highway under study. In this type of scene, the road surface captured by the camera is distant. In ten consecutive video frames, the vehicle will move farther away. Therefore, when the trajectory is not updated for ten frames, the trajectory is deleted. At the same time, the vehicle trajectory and the detection line will only intersect once, and the threshold setting thus does not affect the final counting result. If the prediction box fails to match in consecutive frames, the object is considered to be absent from the video scene, and the prediction box is deleted. From the above process, the global object detection results and tracking trajectories from the complete highway monitoring video perspective are obtained.

TRAJECTORY ANALYSIS

This section describes the analysis of the trajectories of moving objects and the counting of multiple-object traffic information. Most of the highways are driven in two directions, and the roads are separated by isolation barriers. According to the direction of the vehicle tracking trajectory, we distinguish the direction of the vehicle in the world coordinate system and mark it as going to the camera (direction A) and driving away from the camera (direction B). A straight line is placed in the traffic scene image as a detection line for vehicle classification statistics. The detection line is placed at the 1/2 position on the high side of the traffic image. The road traffic flow in both directions is simultaneously counted. When the trajectory of the object intersects the detection line, the information of the object is recorded. Finally, the number of objects in different directions and different categories in a certain period can be obtained.

RESULT AND DISCUSSION

In this section, we describe the performance testing of the methods presented in “Methods” section. We experimented with the vehicle object dataset established in “Vehicle dataset” section. Our experiment used high definition highway videos for three different scenes.

NETWORK TRAINING AND VEHICLE DETECTION

We used the YOLOv3 network for vehicle object detection and our established dataset for network training. In network training, there is no perfect solution for the dataset division. Our dataset dividing method follows the usual usage. We split the dataset into an 80% training set and a 20% test set. Our dataset has 11,129 images, the training set images, and the test set images are randomly selected from the dataset. Due to a large number of dataset pictures, the rate of the test set and training set is sufficient to obtain the model. To obtain an accurate model, the rate of the training set should be high. The training set has 8,904 images, and numerous vehicle samples can be trained to obtain accurate models for detecting cars, buses, and truck targets. The test set has 2225 images with vehicle targets that are completely different from the training set, which is sufficient to test the accuracy of the model that has been trained. We used a batch size of 32 and set the weight attenuation to 0.0005 and the momentum value to 0.9 for the maximum number of training iterations of 50,200. We used a learning rate of 0.01 for the first 20,000 iterations, which

changed to 0.001 after 20,000 iterations. This approach made the gradient fall reasonably and made the loss value lower. To make the default anchor box more suitable for the dataset annotation box to be annotated, we used the k-means++ method to make changes. The training set of our dataset calculated the default anchor box size at the network resolution of 832×832 , and we obtained nine sets of values: [13.2597, 21.4638], [24.1990, 40.4070], [39.4995, 63.8636], [61.4175, 96.3153], [86.6880, 137.2218], [99.3636, 189.9996], [125.6843, 260.8647], [179.7127, 198.8155], [189.3695, 342.4765], with an average IOU of 71.20%. To improve the detection effect of small objects, we did not discard samples with less than 1-pixel value during training but put them into the network for training. We output the result of splicing the feature map of the previous layer of the routing layer before the last yolo layer of Darknet-53 and the 11th layer of Darknet-53. We set the step size to 4 in the upsampling layer before the last yolo layer. When we set the image input to the network, the network resolution was 832×832 instead of the default 416×416 resolution. After the input resolution is increased, when the network is output in the yolo layer, it can have a correspondingly larger resolution and can thus improve the accuracy of the object detection.

A continuous 3000 frames of images were used for vehicle detection in a variety of highway scenes by using our trained model. We extracted and divided the road surface area and put it into the network for vehicle detection. Then, we compared our method with the detection of images with 1920×1080 resolution into the network (without dividing the road surface). We compared the number of object detections under different methods with the actual number of vehicles.

TRACKING AND COUNTING

After obtaining the object box, we performed vehicle tracking based on the ORB feature point matching method and performed trajectory analysis. In the experiment, when the matching point of each object was greater than ten, the corresponding ORB prediction position was generated. Based on the direction in which the tracking trajectory was generated, we used the detection line to judge the direction of motion of the vehicle and classify it for counting. We conducted experiments on the other three videos that are the same as the scene in “Network training and vehicle detection” section but with a different number of frames. We used the *real time rate* to evaluate the speed of the system proposed in this paper, which is defined as the ratio of the time required for the system to process a video to that of the original video played. In Eq. 4, the *system running time* is the time required for the system to process a video, and the *video running time* is the time required for the original video played. The smaller the *real time rate* value is, the faster the system performs the calculations. When the value of the *real time rate* is less than or equal to 1, the input video can be processed in real time.

The results show that the average accuracies of vehicle driving direction and vehicle counting are 92.3% and 93.2%, respectively. In the highway monitoring video, the car class has a small object and is easily blocked by large vehicles. At the same time, there will be multiple cars in parallel, which will affect the accuracy of the track counting. Our original video runs at 30 frames per second. From the calculation of the speed, it can be found that the vehicle tracking algorithm based on the ORB feature is fast. The system processing speed is related to the number of

vehicles in the scene. The greater the number of vehicles, the more features need to be extracted, and the system processing time will thus become longer. In general, the vehicle counting system proposed in this manuscript is very close to real-time processing.

CONCLUSION

This study established a high-definition vehicle object dataset from the perspective of surveillance cameras and proposed an object detection and tracking method for highway surveillance video scenes. A more effective ROI area was obtained by the extraction of the road surface area of the highway. The YOLOv3 object detection algorithm obtained the end-to-end highway vehicle detection model based on the annotated highway vehicle object dataset. To address the problem of the small object detection and the multi-scale variation of the object, the road surface area was defined as a remote area and a proximal area. The two road areas of each frame were sequentially detected to obtain good vehicle detection results in the monitoring field. The position of the object in the image was predicted by the ORB feature extraction algorithm based on the object detection result. Then, the vehicle trajectory could be obtained by tracking the ORB features of multiple objects. Finally, the vehicle trajectories were analyzed to collect the data under the current highway traffic scene, such as driving direction, vehicle type, and vehicle number. The experimental results verified that the proposed vehicle detection and tracking method for highway surveillance video scenes has good performance and practicability. Compared with the traditional method of monitoring vehicle traffic by hardware, the method of this paper is low in cost and high in stability and does not require large-scale construction or installation work on existing monitoring equipment. According to the research reported in this paper, the surveillance camera can be further calibrated to obtain the internal and external parameters of the camera. The position information of the vehicle trajectory is thereby converted from the image coordinate system to the world coordinate system. The vehicle speed can be calculated based on the calibration result of the camera. Combined with the presented vehicle detection and tracking methods, abnormal parking events and traffic jam events can be detected to obtain more abundant traffic information.

In summary, vehicles in Europe, such as in Germany, France, the United Kingdom, and the Netherlands, have similar characteristics to the vehicles in our vehicle dataset, and the angle and height of the road surveillance cameras installed in these countries can also clearly capture the long-distance road surface. Therefore, the methodology and results of the vehicle detection and counting system provided in this analysis will become important references for European transport studies.