Novel real-time safety algorithm for predicting multi-targets in the farmland road

Xiaoming Liang[1*], Fu'en Chen[2], Longhan Chen[3], Deyue Li[4], Bin Guo[4], Yubo Liang[5], Yubin Lan[6]

(1. *China North Vehicle Research Institute, Beijing 100072, China*;
2. *School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing 100044, China*;
3. *Department of Mechanical Engineering, Oakland University, Rochester, MI 48309, USA*;
4. *School of Electronics and Electrical Engineering, Beijing Jiaotong University Haibin College, Huanghua 061199, Hebei, China*;
5. *The First Research Institute of the Ministry of Public Security, Beijing 100048, China*;
6. *College of Engineering, South China Agricultural University, Guangzhou, 510642, China*)

Abstract: The more information obtained about the driving environment, the more ensures driving safety. Due to the complex driving environment of farmland roads, targets beside the road sometimes have an important impact on driving safety. To achieve this goal, a novel real-time detection and prediction algorithm of targets was proposed. The whole image was divided into four parts by RCM: driving region, crossroad region, roadside region, and the other region. In addition, a safety policy for every part was enforced by the algorithm, which was based mainly on the combination of the YOLACT and GPM. On this basis, a self-collected data set of 5000 test samples is used for testing. The detection accuracy of the algorithm in the data set could reach up to 90%, and the processing speed to 30.4 fps. In addition, experiments were carried out on actual farmland roads, and the results showed that the proposed algorithm was able to detect, track, and predict targets on the farmland road, and alarm to driver in time before the targets rush into the road. This study provides an important reference for the safe driving of agricultural vehicles.

Keywords: real-time, safety, algorithm for predicting, multi-target, farmland road, computer vision
DOI: 10.25165/j.ijabe.20231605.6171

Citation: Liang X M, Chen F, Chen L H, Li D Y, Guo B, Liang Y B, et al. Novel real-time safety algorithm for predicting multi-targets in the farmland road. Int J Agric & Biol Eng, 2023; 16(5): 198–203.

1 Introduction

In 2018, there were 244 937 traffic accidents in China and a direct property loss was 1384.56 million RMB yuan[1]. The accident death rate on agricultural vehicles was 45%, and the death accident rate on farmland roads reached 54.5%[2]. It is very important to predict the targets in the farmland roads and remind the driver in time to reduce accidents and improve safety[3-7].

In order to realize target detection in the driving region, some different new technologies have been applied in recent decades. These algorithms can be divided into three categories: feature-based, model-based, and machine learning-based[8]. Li et al.[9] proposed a method to determine the navigation path by analyzing the color characteristics of different regions. Liu et al.[10] introduced prior information and context information into detection. Choi et al.[11] adopted an algorithm based on the texture and morphological features of the work region to overcome the impact of environmental changes. However, the feature-based algorithm is not sensitive to road shape and is easily affected by watermarks and shadows. Compared with the feature-based algorithm, the model-based algorithm matches the model according to the location, distribution, shape, and other information of the road. Wang et al.[12] presented an algorithm for the drivable region using an M-shaped deep architecture model. Liu et al.[13] designed an improved mixed Gaussian model to improve recognition accuracy. As these algorithms have strict requirements on the shape of the road region and need accurate mathematical models, they are mainly used to detect the road in a simple environment. In another way, some researchers have made great efforts with machine learning to solve the problem[14]. Liu et al.[15] proposed a self-supervised online learning method using Support Vector Machine (SVM) for road detection. Varona et al.[16] presented a deep learning approach to automatic road surface monitoring and pothole detection. Yang et al.[17] designed a Region-based Convolutional Neural Network (RCNN) UNet model to solve the problem of road detection and centerline extraction.

The previous detection methods for farmland roads mostly focused on the road surface and edge detection, not on the recognition target beside the road. Therefore, it is difficult for them to adapt to the complex driving environment of agricultural vehicles. To solve the problem, a detection and prediction algorithm based on the combination of the GPM and YOLACT, You Only Look At CoefficienTs[18] was proposed. Experiments were designed to test the accuracy of algorithm prediction and multi-object detection.

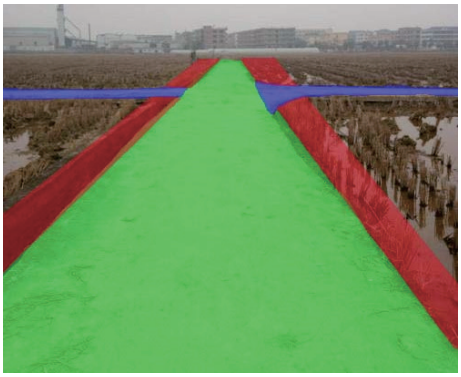
2 Materials and methods

2.1 Road Classification Model

Based on a definition of driving region[19] and the influence of

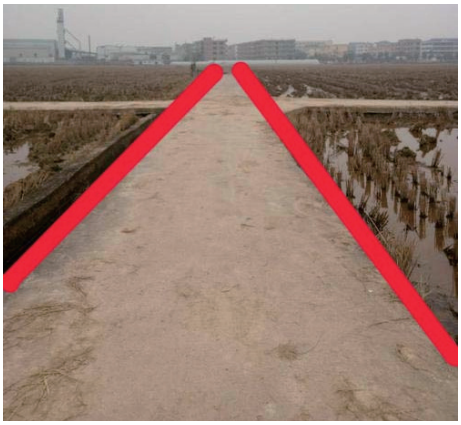
Received date: 2020-09-20 **Accepted date:** 2021-02-20
Biographies: Fu'en Chen, PhD, Associate Professor, research interest: vehicle test and automation, Email: fechen@bjtu.edu.cn; **Longhan Chen**, PhD student, research interests: vehicle optics test, Email: lchen2345678910@oakland.edu; **Deyue Li**, Master, Lecturer, research interest: photoelectric technology, Email: deyuel@bjtuhbxy.edu.cn; **Bin Guo**, Master, Lecturer, research interest: intelligent control system, Email: bguo@bjtuhbxy.edu.cn; **Yubo Liang**, Master, research interest: Object detection, Email: 164172927@qq.com; **Yubin Lan**, PhD, Professor, research interest: agricultural machinery, Email: ylan@scau.edu.cn.
***Corresponding author: Xiaoming Liang**, PhD, research interest: vehicle machine vision. China North Vehicle Research Institute, No.4 Huaishuling Fengtai District Beijing 100072, China. Email: 17111046@bjtu.edu.cn.

other regions on driving region, a Road Classification Model (RCM) was proposed, which divides a whole image into four parts (separated by different colors) as shown in Fig 1a. They are driving region (DR marked by green color), crossroad region (CR marked by blue color), roadside region (RR marked by red color), and the other region (OR), respectively. RR refers to the targets in this region that will enter the driving region after 3 s.



a. RCM segmentation image

b. Trapezoidal vertex figure



c. Trapezoidal region figure

Note: RCM: Road Classification Model proposed in this study.

Figure 1 Actual farmland road RCM segmentation and procedure images

As a road region is trapezoidal from the visual point of view, the adjacent frame images have a certain correlation, and the trapezoid of adjacent frames will not change suddenly in the video[20]. Therefore, the road region in the next frame can be predicted by the region of the first 10 frames.

The modeling procedure is as follows:

Step 1: The road region is detected, then four vertices of the road are found, next two sidelines are figured out based on the vertices, as shown in Figure 1b and Figure 1c. This region contains two parts: DR and CR.

Step 2: The two sidelines are moved initially outward parallel by 10 pixels as the RR. The region will be expanded according to the moving distance of a target in 3s, and the value is recorded before the algorithm exits, and it will be read as a starting value next time.

Step 3: The DR and CR are separated by the two corresponding sidelines, as shown in Figure 1a.

Step 4: According to the influence of driving from targets in different regions, a safety policy is proposed. DR is the most concerned, followed by CR, then RR, and finally OR. As soon as some targets are located in two or more regions, the region with high concern is selected. The predicting frequency of the target in different regions is different. The target position of the next frame in both DR and CR is predicted with the first 10 frames of data. The reaction time of drivers without specialized training is between 0.2 s and 0.3 s[21]. The target position after 0.3 s in RR is predicted with every 9 frames of data.

Step 5: All targets in those three regions are detected, tracked, and classified by the safety policy. The Gray Prediction Model (GPM)[22] is adopted to predict a position, and the targets are masked in DR, as shown in Figure 2.

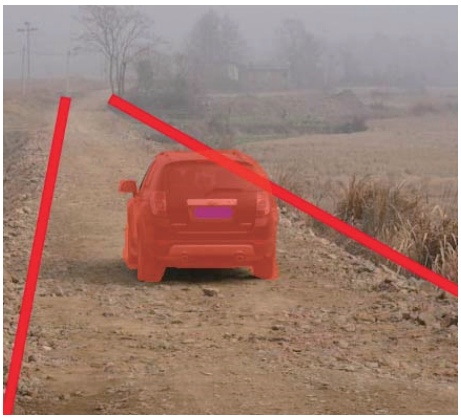


Figure 2 Detection result image on actual farmland road

2.2 Data acquisition and marking

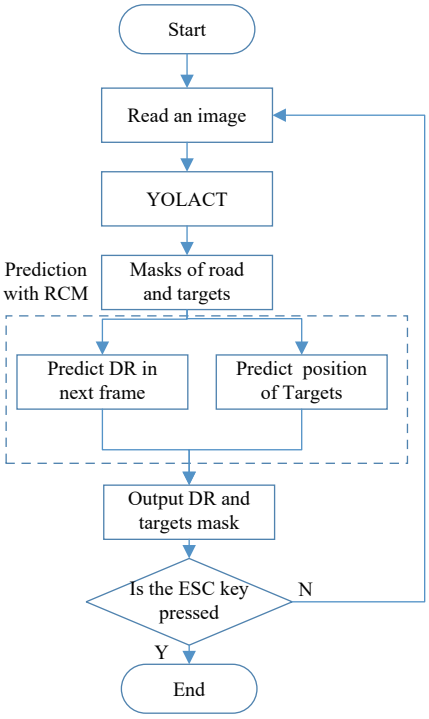
The data was collected on a farmland road in Lishu County, Siping City, Jilin Province, China, from March 11 to November 29, 2019. One day a week, three videos were recorded every day, namely, morning, noon, and evening. A total of 114 videos were recorded, the duration of each video was about 60 min, the frame rate was 30 fps, and the image resolution was 640×480 pixels. The targets detected on farmland roads can be divided into vehicles, tractors, motorcycles, tricycles, bikes, persons, and dogs. 15 000 random images were selected and added to the training sample, and 5000 images into the test sample. The Apollo scape[23] dataset, released in 2018 by Baidu Apollo, has been added to training samples to achieve a good generalization of training results. A total of 26 data instances with different semantic terms are defined. A complete training sample of Apollo Scape includes 17 062 images and corresponding semantic annotation information. The target categories in Apollo scape were classified (car and truck were classified as vehicle class, motorcycle, bike, and rider were classified as motorcycle class, sidewalk, and highway were classified as road class, and the rest was ignored). Meanwhile, 2164

tractor images selected from the ImageNet[24] dataset were added to the training sample. Therefore, there were 34 226 images in the training sample, and 5000 images in the test sample in all. Labelme[25] was adopted to mask the training sample and test sample, which was an open-source image labeling tool.

A Deep-learning framework TensorFlow[26], 16 G memory, windows 10 operating system, Intel(R) core(TM) i5-9400 CPU, and Nvidia GeForce RTX 2060 SUPER (8 G) GPU were adopted to train the network and test algorithm performance.

2.3 Algorithm

The operation procedure of the algorithm is shown in Figure 3. The algorithm starts by reading an image. Furthermore, the image is processed by YOLACT, and then road mask and target masks (area information of target image) are obtained.



Note: DR: Driving region; ESC key: Escape key.

Figure 3 Flowchart of the algorithm

Step 1: The targets are classified by RCM. The target position of the next frame in both DR and CR is predicted with the first 10 frames of data. The reaction time of a driver without specialized training is between 0.2 s and 0.3 s[21]. The target position after 0.3 s in RR is predicted with every 9 frames of data. The obtained prediction data can provide sufficient reaction time for the driver.

Then the ORB algorithm[27] with real-time speed is adopted to extract and describe the feature points, and these feature points are matched with specific targets through Hamming distance.

Step 2: A GPM[22] was built:

A grey derivative is defined as,

$$\begin{cases} x^0(k) = x^1(k) - x^1(k-1) \\ z^1(k) = ax^1(k) + (1-a)x^1(k-1), \ (k = 1,\ldots,n) \end{cases} \tag{1}$$

where, k represents the k-TH data; $x^0(k)$ is the original sequence; $x^1(k)$ is the accumulated sequence of $x^0(k)$, $z^1(k)$ is adjacent value generation sequence of $x^1(k)$, a is called development coefficient.

Then a grey differential equation model is defined as,

$$x^0(k) + az^1(k) = b, \ (k = 1,\ldots,n) \tag{2}$$

where, b is called ash dosage.

Finally, a predicted value formula is

$$y^1(k+1) = \left(x^0(1) - \frac{b}{a}\right) \times e^{(-ak)} + \frac{b}{a}, \ (k = 1,\ldots,n) \tag{3}$$

where, $y^1(k)$ is the predicted sequence.

Step 3: One key point of a target could be calculated according to a center of gravity[28] to speed up a calculation, and the formula is as follows:

$$\begin{cases} x = \dfrac{\sum_{i=1}^{n}(x_i + x_{i+1})\begin{pmatrix} x_i & x_{i+1} \\ y_i & y_{i+1} \end{pmatrix}}{3 \times \sum_{i=1}^{n}\begin{pmatrix} x_i & x_{i+1} \\ y_i & y_{i+1} \end{pmatrix}} \\ y = \dfrac{\sum_{i=1}^{n}(y_i + y_{i+1})\begin{pmatrix} x_i & x_{i+1} \\ y_i & y_{i+1} \end{pmatrix}}{3 \times \sum_{i=1}^{n}\begin{pmatrix} x_i & x_{i+1} \\ y_i & y_{i+1} \end{pmatrix}}, \ (i = 1,\ldots,n; x_{n+1} = x_1; y_{n+1} = y_1) \end{cases} \tag{4}$$

where, (x, y) is the key point; (x_i, y_i) is a boundary point of a target mask.

Step 4: According to a relationship between the last frame and the predicted value key point, the mask of a target is moved as follows:

$$\begin{cases} c_i = c_j + b - b_1 \\ d_i = d_j + a - a_1, \ (i = 1,\ldots,n; j = 1,\ldots,n) \end{cases} \tag{5}$$

where, (a, b) is a predicted key point; (a_1, b_1) is a key point in the last frame; (c_i, d_i) is a predicted point of a target mask; (c_j, d_j) is a point of a target mask in the last frame.

Step 5: After DR and target masks are output, the algorithm will wait for 10ms, during which it will detect the ESC key, the algorithm will exit when it is pressed, otherwise, it will continue to read the next image.

The Root Mean Squared Error (RMSE) test method[29] was introduced to test the dispersion degree of a predicted value.

$$RMSE = \sqrt{\frac{1}{n}\sum_{k=1}^{n}\left(y^1(k) - x^0(k)\right)^2} \tag{6}$$

3 Results

3.1 Accuracy experiment of prediction

In order to test the algorithm, an accuracy experiment of prediction was designed. As mentioned in Section 2, 5000 test images of the driving region were selected for prediction.

As listed in Table 1, the RMSE differences between mainstream prediction algorithms and the algorithm above under different training data numbers were compared in the collected data. The RMSE of Moving Average[30], Simple Exponential Smoothing[31], Holt Linear Trend[32], and Auto Regression Integrated Moving Average (ARIMA)[33], increased with the increase of training data numbers. Markov Model[34], Naive Bayesian Model[35], and Gaussian Process[36] showed great changes, and their change trends were similar. The amount of training data was less than 25, and the RMSE value was greater than 25 and gradually decreased. And that of training data was more than 50, the RMSE value was less than or equal to 10, indicating that the predicted value was close to the test value. The algorithm above can get good results in a small amount of training data (less than 25). When there were only 8 data, RMSE was less than 10, which can get good prediction effect in the collected data, and for 10 data, RMSE was only 8. With the increase in training data, the RMSE did not decrease

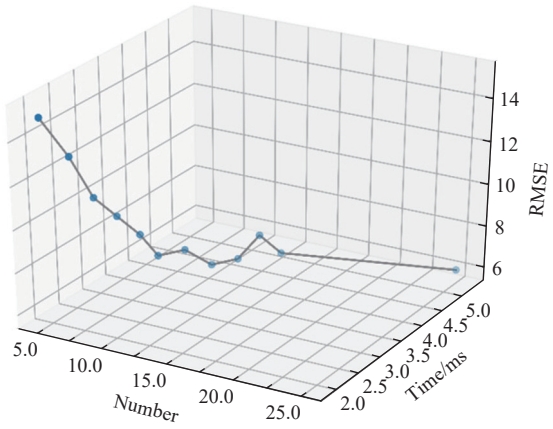
significantly. The lowest point was 25, the RMSE was 6. Then, with the increase in training data, the value of RMSE gradually increased, which should be due to the increase in data fluctuation. It showed that the algorithm of this study was suitable for the prediction of a small amount of data.

Table 1 Prediction accuracy test data in different numbers of training data

Method	Root Mean Squared Error (RMSE)						
	5 training data	8 training data	10 training data	25 training data	50 training data	75 training data	100 training data
Markov Model	65	66	40	25	9	7	6
Naive Bayesian Model	71	70	42	26	10	8	7
Gaussian Process	79	80	45	24	10	8	6
Moving Average	20	21	23	28	33	40	42
Simple Exponential Smoothing	18	20	21	25	30	35	42
Holt Linear Trend	16	18	20	28	30	33	39
ARIMA	15	17	18	24	28	30	33
Proposed method in this study	15	8	7	6	11	12	15

Note: ARIMA: Auto Regression Integrated Moving Average.

As shown in Figure 4, the time required with different training data volumes and the RMSE generated were compared. With the increase of data volume (less than 25), the RMSE value decreased. When the amount of data was less than 10, the value of RMSE decreased obviously, but the decrease was not obvious between 10 and 25. Therefore, the data volume was 10, and the best balance between consumption time and prediction accuracy can be obtained.

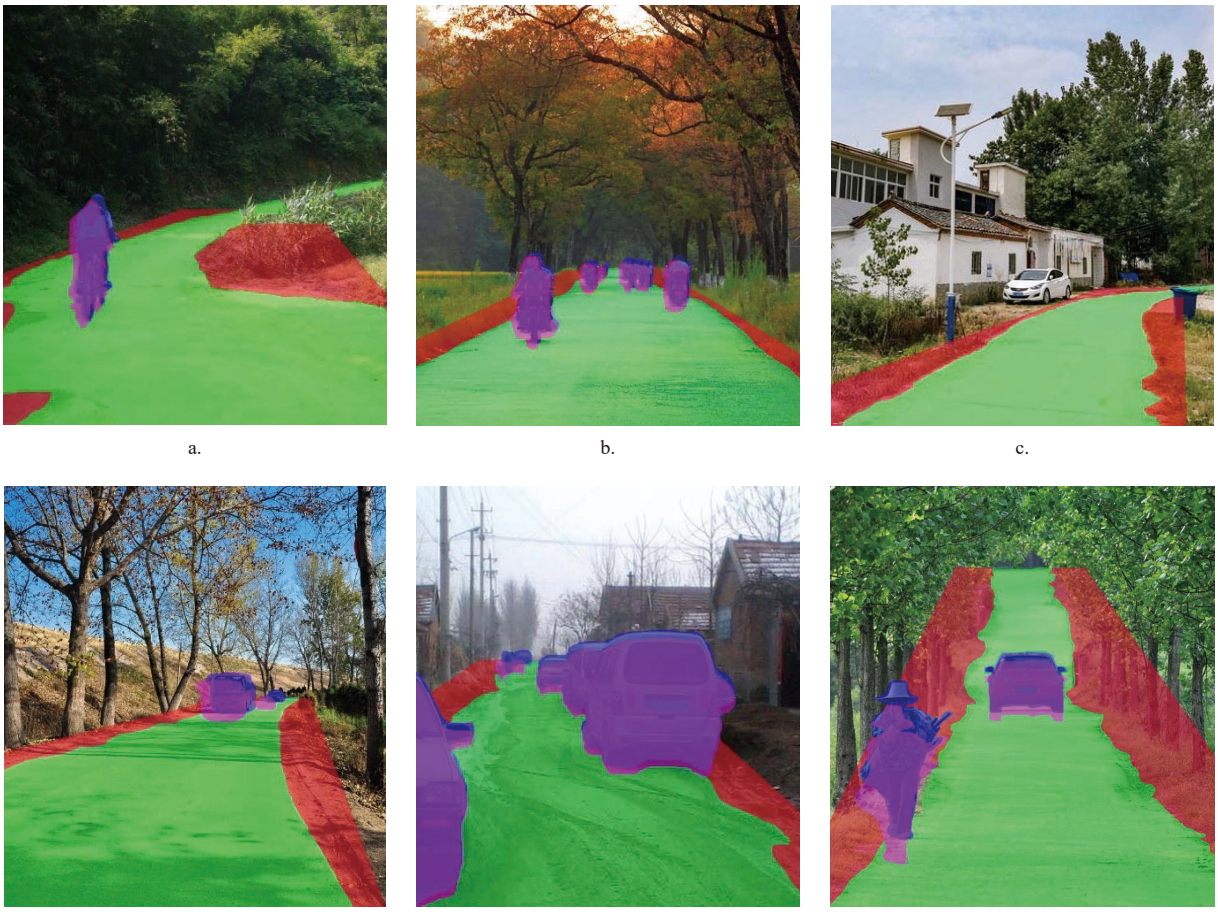


Note: RMSE: Root Mean Squared Error.

Figure 4 Time and RMSE of the algorithm under different numbers of test data

3.2 Object detection and tracking

In order to test the algorithm, a multiple-target detection experiment had been designed. As mentioned in Part 2, 5000 test images of farmland roads were selected for detection, in addition, we carried out a test on an actual farmland road, and the results are shown in Figure 5.



a. b. c.

d. e. f.

Note: The blue mask represents the original region of target; the purple presents predicted region; green presents driving region (DR); and red presents RR.

Figure 5 Processing results of six random images

As shown in Figure 5a, a bike belonging to DR can be detected and predicted. In Figure 5b, three bikes and four motorcycles can be detected and predicted, the second motorcycle from the left was not moving, as it was located in DR and then marked. In Figure 5c, as the car was parked inside the OR, it was not marked. In Figure 5d, the speed of the car was higher than that of the truck, therefore, the moving distance predicted by the car was longer than that of the truck. In Figure 5e, the difference between a predicted value and the original was also obvious between people and vehicles that moved and parked on the roadside. In Figure 5f, due to the opposite movement direction, the position difference of the person between the true value and the predicted one was larger than that of vehicle. As shown above, this algorithm could achieve the detection and prediction of multiple uncertain targets on unstructured farmland.

5000 images with 550×550 were selected to test multiple targets. The model was trained on one GPU with ImageNet[24] training parameters as preset parameters. The batch size was set to 8, which was suitable for batch normalization, there was no additional BN layer added, the SGD method was used to train 800 k iterations, and the initial learning rate was 10^{-3}. When the iteration number was 280 k, 600 k, 700 k, and 750 k respectively, the learning rate would be divided by 10 using a weight decay of 5×10^{-4}, momentum of 0.9. The backbone was ResNet-101[37].

As listed in Table 2, the performance of the algorithm mentioned above and those of the most advanced algorithms in the collected data sample were compared. This table was all the speeds calculated on one RTX2060, so some of the speeds listed may be different from those in the original paper. The algorithm speed proposed was 30.4 fps, which can be used for real-time detection. YOLACT is a one-stage instance segmentation model, while the others are two-stage, which can achieve high-speed detection by removing the second stage. Taking average precision (AP) into account, the vehicle was of highest AP, up to 90.5%, motorcycle was followed by 89.8%, tractor was a little lower at 89.4%, tricycle, road, person, and dog were to 88.9%, 85.6%, 81.4%, and 77.5% respectively. The experimental results show that this method has a good effect on the real-time processing of multi-target detection.

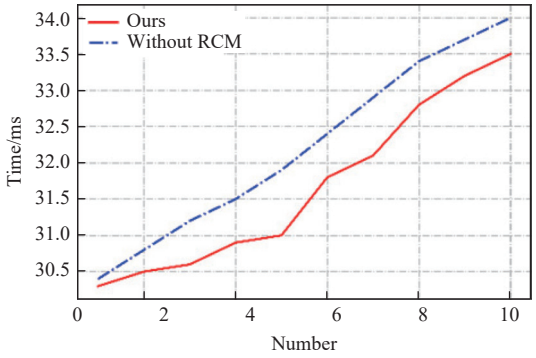
Table 2 Comparison of the performance of algorithms

Method	FPS	Time/ms	AP/%							mAP
			Road	Vehicle	Tractor	Motorcycle	Tricycle	Person	Dog	
FICS[38]	6.3	158.7	83.7	89.7	88.9	88.5	88.0	79.9	76.8	85.1
MS R-CNN [39]	8.2	122.0	92.1	96.4	95.6	96.4	95.0	88.1	83.5	92.4
Mask R-CNN [40]	8.1	116.3	91.2	95.1	94.3	95.5	93.7	86.8	82.3	91.3
Ours	30.4	32.9	85.6	90.5	89.4	89.8	88.9	81.4	77.5	86.1

Note: FPS: Frames per second; mAP: Mean average precision.

As described in Sections 2.1 and 2.3, a comparative test of RCM can be performed.

As shown in Figure 6, with the increase of detection targets in the image, the running time of the algorithm was also increasing, but the running time of the algorithm with RCM was smaller than that of the algorithm without RCM. The algorithm without RCM detected and predicted all the targets in an image, resulting in a positive correlation between the running time and the number of targets. In the algorithm with RCM, although the time was also increased, when the average number of targets reached 10, the running time was only 33.5 ms. By adding RCM, the real-time prediction range of the algorithm was increased from 8 to 10 targets.



Note: Ours means the method proposed in this study.

Figure 6 Time difference caused by different number of targets

4 Conclusions

A real-time algorithm based on the combination of the YOLACT and GPM was proposed and verified. The algorithm can effectively segment an image into four parts with RCM, and a safety policy was developed for each part to predict the targets that would rush into the road. In the data set, the accuracy of road, vehicle, tractor, motorcycle, tricycle, person, and dog could reach 85.6%, 90.5%, 89.4%, 89.8%, 88.9%, 81.4%, and 77.5%, respectively. Furthermore, the algorithm has good detection accuracy for multiple targets, and its speed can reach 30.4 fps. Two RMSE difference experiments were carried out, and the results showed that the algorithm has good prediction data when the amount of training data is small (less than 25). This algorithm can be applied to agricultural vehicle navigation and alarm on farmland roads, and it would be helpful for the further study of roadkill. The algorithm is suitable for urban roads, highways, and rural roads, but not suitable for desert and mountain areas, as the last two kinds of roads cannot be segmented.

Acknowledgements

This study was financially supported by Beijing Jiaotong University (C18A800090) and China North Vehicle Research Institute. All the support from the above organizations is gratefully acknowledged.

[References]

[1] National Bureau of Statistics of China. Statistical bulletin of national economic and social development of the People's Republic of China in 2019. Available:http://www.stats.gov.cn/tjsj/zxfb/202002/t20200228_1728913.html. (in Chinese)

[2] Xu X, Analysis of Road Traffic Accidents in China and Measures to Prevent Them. China Safety Science Journal, 2013; 23(11): 120–125 (in Chinese).

[3] Yang L L, Xu Y Y, Liang Y J, Qin J, Li Y B, Wang X X, et al. Extraction of Straight Field Roads between Farmlands based on Agricultural Vehicle-mounted LiDAR. Int J Agric & Biol Eng, 2022; 15(5): 155–162.

[4] Guo J B, Wang G Q, Guan W, Chen Z R, Liu Z B. A Feasible Region Detection Method for Vehicles in Unstructured Environments based on PSMNet and Improved RANSAC. Multimed Tools Appl, 2023; 82: 43967–43989.

[5] Vivek K P, Akash C G, Ayesha C and Ayesha C. Computer Vision and Deep Learning Based Framework for Road Scene and Surface Segmentation in Unstructured Environment. 2022 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI), Delhi, India, 2022; pp.1–6. doi: 10.1109/SOLI57430.2022.10294507.

[6] Lei G N, Yao R T, Zhao Y D, Zheng Y L. Detection and Modeling of Unstructured Roads in Forest Areas Based on Visual-2D Lidar Data Fusion. Forests, 2021; 12(7): 820.

[7] Zhu M Y, Wang H P, Li P P, Liu J T. An ANN-based Real-time

Unstructured Road Detection Approach under Time-varying Illumination. 2019 IEEE International Conference on Real-time Computing and Robotics, Kandima, Maldives, 2019; pp.293–298. doi: 10.1109/RCAR 47638.2019.9044119

[8] Chen W W, Wang W X, Wang K, Li Z Y, Li H, Liu S. Lane Departure Warning Systems and Lane Line Detection Methods based on Image Processing and Semantic Segmentation: A Review. Journal of Traffic and Transportation Engineering (English Edition), 2020; 7(6): 748–774.

[9] Li J B, Chen B Q, Liu Y, Zha T. Detection for Navigation Route for Cotton Harvester based on Machine Vision. Transactions of the CSAE, 2013; 29(11): 11–19 (in Chinese).

[10] Liu Z G, Du J, Tian F, Wen J Z. Traffic Sign Recognition Using an Attentive Context Region-Based Detection Framework. Chinese Journal of Electronics, 2021; 30(6): 1080–1086.

[11] Choi K H, Han S K, Han S H, Park K H, Kim K S, Kim S. Morphology-based Guidance Line Extraction for An Autonomous Weeding Robot in Paddy Fields. Computers and Electronics in Agriculture, 2015; 113: 266–274.

[12] Wang X W, Li S H, Liang X, Zheng J J. Drivable Region Recommendation Model for Unstructured Road Based on M-shaped Deep Architecture. China Journal of Highway and Transport, 2022; 35(12): 205–218.

[13] Liu Y P, Cui T, Zhou C B, Li X C, Liu T. Abnormal Vehicle Detection Based on Improved Gaussian Mixture Model and Graph Handle. Computer Engineering and Science, 2020; 2020(2): 266–272. (in Chinese).

[14] Ahn J, Lee Y, Kim M, Park J. Intersection Navigation in Unstructured Environment Using Multi-Task Network. Journal of Advanced Transportation, 2022; 2022: 1–13.

[15] Liu Y G, Wang X, Li L, Cheng S, Chen Z. A Novel Lane Change Decision-making Model of Autonomous Vehicle based on Support Vector Machine. IEEE Access, 2019; 7: 26543–26550.

[16] Varona B, Monteserin A, Teyseyre A. A Deep Learning Approach to Automatic Road Surface Monitoring and Pothole Detection. Pers Ubiquit Comput, 2020; 24: 519–534.

[17] Yang X F, Li X T, Ye Y M, Lau R Y K, Zhang X F, Huang X H. Road detection and centerline extraction via deep recurrent convolutional neural network U-Net. IEEE Transactions on Geoence and Remote Sensing, 2019; 57(9): 7209–7220.

[18] Bolya D, Zhou C, Xiao F Y, Lee Y J. YOLACT: Real-time Instance Segmentation. 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South): IEEE, 2019; pp.9157–9166. doi: 10.1109/ICCV.2019.00925.

[19] Qian Y, Dolan J M, Yang M. DLT-Net: Joint Detection of Drivable Areas, Lane Lines, and Traffic Objects. IEEE Transactions on Intelligent Transportation Systems; 2020, 21(11): 4670–4679.

[20] Thakare K V, Sharma N, Dogra D P, Choi H, Kim I J. A multi-stream deep neural network with late fuzzy fusion for real-world anomaly detection. Expert Systems with Applications, 2022; 201: 117030.1–117030.13.

[21] Yang W, Zhang X, Lei Q, Cheng X. Research on Longitudinal Active Collision Avoidance of Autonomous Emergency Braking Pedestrian System (AEB-P). Sensors, 2019; 19(21): 4671.1–4671.34.

[22] Wu W L. An Intelligent Gray Prediction Model Based on Fuzzy Theory. International Transactions on Electrical Energy Systems, 2022; 25: 1–8.

[23] Huang X Y, Wang P, Cheng X J, Zhou D F, Geng Q C, Yang R G. The ApolloScape Open Dataset for Autonomous Driving and Its Application. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020; 42(10): 2702–2719.

[24] Vaishaal S, Rebecca R, Horia M, Alex F, Benjamin R, Ludwig S. Evaluating Machine Accuracy on ImageNet. Proceedings of Machine Learning Research, 2020; 119: 8634–8644.

[25] Kentaro Wada. labelme: Image Polygonal Annotation with Python. 2019. Available:https://github.com/wkentaro/labelme.

[26] Marijn H, Anneroos P H. Machine Learning and the Platformization of the Military: A Study of Google's Machine Learning Platform TensorFlow. International Political Sociology, 2022; 16(2): 1–19.

[27] Campos C, Elvira E, Rodríguez J J G, Montiel J M M, Tardós J D. ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual–Inertial, and Multimap SLAM. IEEE Transactions on Robotics, 2021; 37(6): 1874–1890.

[28] Wang J X, Zhao G C, Lu F N, Zhang G B, Ceng T, Qiao T R. Sphere geodesic octree grid method for true three-dimensional geological model construction. Journal of Geo-Information Science, 2019; 21(8): 1161–1169 (in Chinese).

[29] Hodson T O. Root-mean-square error (RMSE) or mean absolute error (MAE): when to use them or not. Geosci. Model Dev, 2022; 15(14): 5481–5487.

[30] Su Y Q, Cui C R, Qu H. Self-attentive moving average for time series prediction. Journal of Nanjing University (Natural Science), 2022; 4: 649–657.

[31] Devon B, Nikolaos K, Rickard S, Jacek N. Automatic robust estimation for exponential smoothing: Perspectives from statistics and machine learning. Expert Systems with Applications, 2020; 160: 1–41.

[32] Şahinli M A. Potato Price Forecasting with Holt-Winters and ARIMA Methods: A Case Study. Am. J. Potato Res, 2020; 97: 336–346.

[33] Khan S, Hela A. ARIMA Model for Accurate Time Series Stocks Forecasting. International Journal of Advanced Computer Science and Applications, 2020; 11(7): 524–528.

[34] Mor B, Garhwal S, Kumar A. A Systematic Review of Hidden Markov Models and Their Applications. Arch Computat Methods Eng, 2021; 28: 1429–1448.

[35] Wickramasinghe I, Kalutarage H. Naive Bayes: applications, variations and vulnerabilities: a review of literature with code snippets for implementation. Soft Computing, 2021; 25: 2277–2293.

[36] Mahdi S, Alexandra S L, Nilay S. Machine-learning methods for integrated renewable power generation: A comparative study of artificial neural networks, support vector regression, and Gaussian Process Regression. Renewable and Sustainable Energy Reviews, 2019; 108: 513–538.

[37] Ye X Y, Mao C L, Wang H S, Yuan S L. Research on Road Terrain Recognition based on Resnet-101 Deep Learning Network. Auto Time, 2021; 16: 30–31. (in Chinese)

[38] Li Y, Qi H Z, Dai J F, Ji X Y, Wei Y C. Fully convolutional instance-aware semantic segmentation. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA: IEEE, 2017; pp.4438–4446. doi: 10.1109/CVPR.2017.472.

[39] Huang Z J, Huang L C, Gong Y C, Huang C, Wang X G. Mask Scoring R-CNN. 2019 IEEE/CVF Conference Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA: IEEE, 2019; pp.6409–6418. doi: 10.1109/CVPR.2019.00657.

[40] He K M, Gkioxari G, Dollár P, Girshick R. Mask R-CNN. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020; 2(42): 386–397.