S & M 3797

Autonomous Multitask Driving Systems Using Improved You Only Look Once Based on Panoptic Driving Perception

Chun-Jung Lin,[1] Cheng-Jian Lin,[1*] and Yi-Chen Yang[2]

[1]Department of Computer Science & Information Engineering, National Chin-Yi University of Technology, Taichung 411, Taiwan
[2]Department of Electronic Engineering, National Chin-Yi University of Technology, Taichung 411, Taiwan

(Received April 30, 2024; accepted September 17, 2024)

Keywords: panoptic driving perception, YOLO, Taguchi method, drivable area detection, lane detection, vehicle detection

With the continuous development of science and technology, automatic assisted driving is becoming a trend that cannot be ignored. The You Only Look Once (YOLO) model is usually used to detect roads and drivable areas. Since YOLO is often used for a single task and its parameter combination is difficult to obtain, we propose a Taguchi-based YOLO for panoptic driving perception (T-YOLOP) model to improve the accuracy and computing speed of the model in deteching drivable areas and lanes, making it a more practical panoptic driving perception system. In the T-YOLOP model, the Taguchi method is used to determine the appropriate parameter combination. Our experiments use the BDD100K database to verify the performance of the proposed T-YOLOP model. Experimental results show that the accuracies of the proposed T-YOLOP model in deteching drivable areas and lanes are 97.9 and 73.9%, respectively, and these results are better than those of the traditional YOLOP model. Therefore, the proposed T-YOLOP model successfully provides a more reliable solution for the application of panoramic driving perception systems.

1. Introduction

Nowadays, many automobile manufacturers have invested in the research and development and application of autonomous driving assistance technology. However, with the complexity of the road environment and the increasing number of vehicles, autonomous driving assistance systems are facing various challenges. In actual driving situations, there are often twists and turns on roads and sudden lane reductions due to construction ahead. These factors may affect the normal operation of an autonomous driving system. Under such circumstances, it is difficult for vehicles to maintain an ideal distance, which may lead to traffic accidents, and drivers need to react quickly to rapidly changing conditions. Therefore, while developing autonomous driving technology, we must deeply consider various real-world scenarios to ensure the overall performance and safety of the system.

*Corresponding author: e-mail: cjlin@ncut.edu.tw

https://doi.org/10.18494/SAM5114

ISSN 0914-4935 © MYU K.K.
https://myukk.org/

Drivable area detection and lane detection are key and indispensable technologies for self-driving systems. Their main purpose is to ensure that vehicles only drive on legal and safe roads to avoid entering prohibited or dangerous road sections and reduce potential safety hazards. Aggarwal *et al.*(1) proposed detecting the drivable area of vehicles. They used first-person perspective images to identify floor areas. To adapt to the challenges of complex scenes, their method integrates surface cue classification, density cues for floor locations, and geometric cues. This comprehensive analysis helps construct a walkable floor area mask. Furthermore, this method uses the Grabcut algorithm for multiple iterations to refine the area definition. Long *et al.*(2) proposed a widely recognized semantic segmentation method called fully convolutional networks (FCNs), which is specifically used to deal with the problem of semantic segmentation. The core of this method is to directly use the ground truth obtained from image segmentation as supervision information to conduct complete network training from input to output. This training process enables the network to accurately predict each pixel, thereby greatly improving the accuracy and reliability of segmentation. In terms of lane detection, Canny(3) proposed a comprehensive and accurate target framework aimed at defining and detecting edges in images. This framework not only includes criteria for edge detection and localization, but also emphasizes unique responses to single edges to avoid multiple detections. Furthermore, on the basis of these goals, they developed an efficient algorithm that can optimize edge detection at different scales. The core of this method is to use gradient-based technology combined with Gaussian smoothing to process the image, and then accurately identify edges by marking the maximum value of the gradient amplitude. Dong *et al.*(4) used the Canny method on the road to monitor whether the car deviated from the lane and to detect lane markings in the images obtained by the forward-looking vehicle camera. The purpose of this algorithm is to detect the two lanes closest to the car and obtain a more accurate departure warning.

In deep learning, object position detection and classification methods are divided into one-stage and two-stage methods. Early deep learning methods are mainly two-stage methods, such as Region-based Convolutional Neural Networks (RCNNs), Fast-RCNN, and Faster RCNN. However, because object position detection and classification are performed separately, there are certain limitations in speed. To improve the two-stage detection speed problem, the one-stage deep learning method is used to detect and classify object position at the same time, such as You Only Look Once (YOLO).(5) The YOLO model is widely used in the field of image classification and recognition. Wang *et al.*(5) proposed the YOLOv4 model for real-time object detection and semantic segmentation. The YOLOv4 model uses CSPDarkNet53 as its backbone network. This network structure is not only powerful but also efficient and capable of processing large amounts of data and producing results rapidly. In addition, it uses PANet for feature fusion, and features extracted from different levels can be better combined together, thereby improving detection accuracy. The YOLOv4 model adopts a regression strategy to improve the flexibility of bounding box regression. This strategy solves a significant limitation in traditional object detection methods, which is the inability to effectively consider objects with asymmetric features. By considering the center of gravity in the feature map, the YOLOv4 model can better capture feature distribution, thereby achieving more accurate and detailed object detection and segmentation.

Generally, the YOLO model focuses on a single task. To improve training efficiency, multitask training methods of traditional YOLO have begun to receive attention. Wu *et al.*(6) proposed an innovative panoramic driving perception network, called the YOLOP model, whose design goal is to simultaneously perform the three key tasks of traffic object detection, drivable area detection, and lane detection. The network structure of the YOLOP model includes a shared encoder and three decoders dedicated to different tasks, thereby effectively integrating the multitask learning process. Huang(7) improved the volume base and pooling layers on the original YOLOP architecture and used module changes and the reduction of convolutional layers to improve the accuracy of the model. That is, the original convolution layer is changed into a convolution, batch normalization, exponential linear unit (ELU) function (CBE) module, and the rectified linear unit (ReLU) excitation function is changed into an ELU function. Although this can reduce training time, the training speed is still relatively low.

Because determining the parameters of a YOLOP model is difficult, some researchers(6,7) have applied the trial-and-error method. However, the parameter design in the network architecture is also a key factor related to the overall performance. In engineering, the Taguchi method is often used to optimize system parameters. The Taguchi method(8) is performed by statistically controlling the experimental and production processes to achieve the dual objectives of improving the quality and reducing the cost of an experiment. This method utilizes orthogonal tables to provide a complete factorial design for limited experiments. The accuracy of the results obtained through the Taguchi method also makes it an option for numerous optimization problems in engineering and other fields. Therefore, in this study, we applied the Taguchi method to determine the optimal parameter combination of a YOLOP.

In this study, we propose a Taguchi-based YOLO for a panoptic driving perception (T-YOLOP) model to improve the accuracy and computing speed of the model in detecting road and drivable areas, making it a more practical panoptic driving perception system. The major contributions of this study are as follows:

1) To improve the training efficiency of the traditional YOLO model, the YOLOP model is used for multitask training methods.
2) To determine the appropriate parameter combination of the YOLOP model, the Taguchi method is used.
3) Experimental results show that the accuracies of the proposed T-YOLOP in detecting drivable area and lanes are 97.9 and 73.9%, respectively, and these results are better than those of the traditional YOLOP model.

The remainder of this paper is organized as follows. In Sect. 2, we present the proposed T-YOLOP model for drivable area segmentation and lane detection. In Sect. 3, we introduce the experimental results of the proposed T-YOLOP model. We also compare our model to other models. In Sect. 4, we provide our conclusions.

2. Materials and Methods

In this section, the network architecture of the T-YOLOP model is introduced. The specific experimental process and steps are shown in detail in Fig. 1. First, we use the Taguchi method to

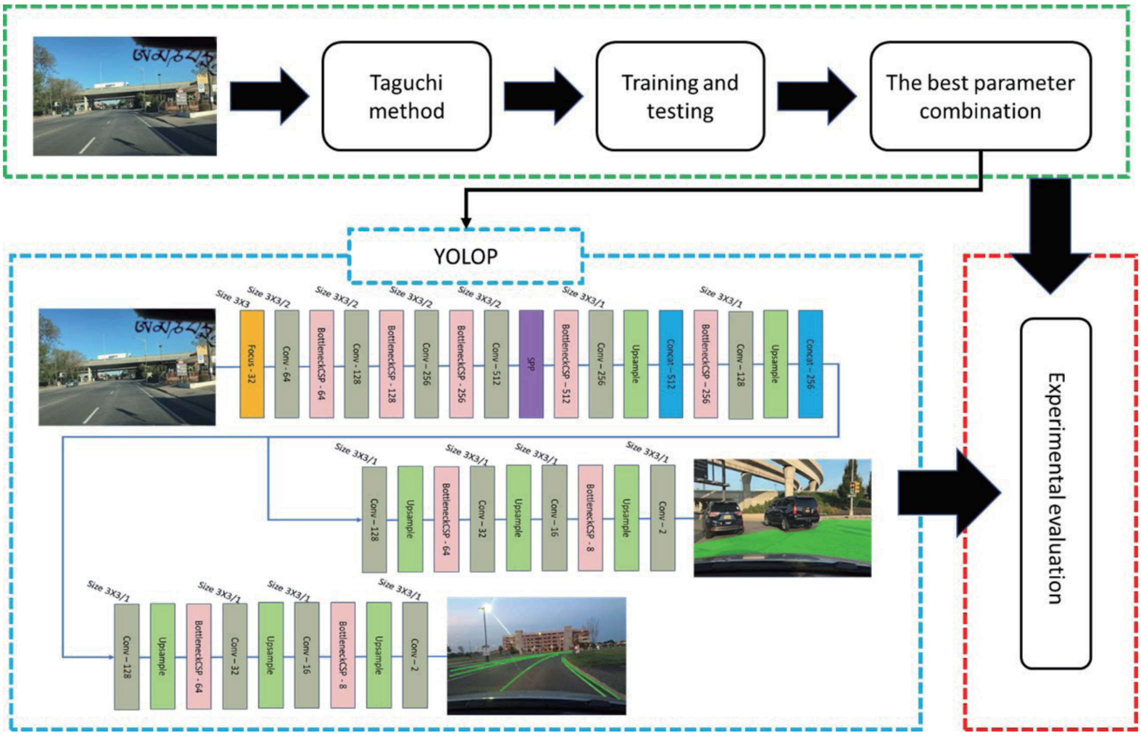


Fig. 1. (Color online) Network architecture of T-YOLOP model.

analyze the parameters of the T-YOLOP model and determine its appropriate parameter combination. This process is designed to improve the accuracy of the model. Next, we will apply these selected optimal parameter combinations to the T-YOLOP model.

2.1 Signal preprocessing

YOLOP's network architecture is divided into three parts, namely, backbone, neck, and detect head, as shown in Fig. 2. Backbone is the backbone network, whose main function is to extract feature information from the input image. YOLOP utilizes YOLOv4's CSPDarknet[9] as its backbone network because YOLOv4 performs well in object detection. Neck fuses features generated by backbone, which consists of spatial pyramid pooling (SPP)[10] and feature pyramid network (FPN).[9] SPP generates and fuses features at different resolutions, whereas FPN fuses semantic features at different levels. The generated features have both multiresolution and multilevel semantic information. Detect head includes drivable area detection and lane detection. Lane detection and drivable area detection use an anchor-based multi-resolution detection method combined with path aggregation network (PAN) and FPN. PAN is a bottom-up network structure used to transfer positioning features, whereas FPN transfers semantic features from top to bottom. By combining both PAN and FPN, better feature extraction results are obtained. The low-level output of FPN is introduced into the segmentation branch, and after three upsamplings, the output is restored to the original image size.

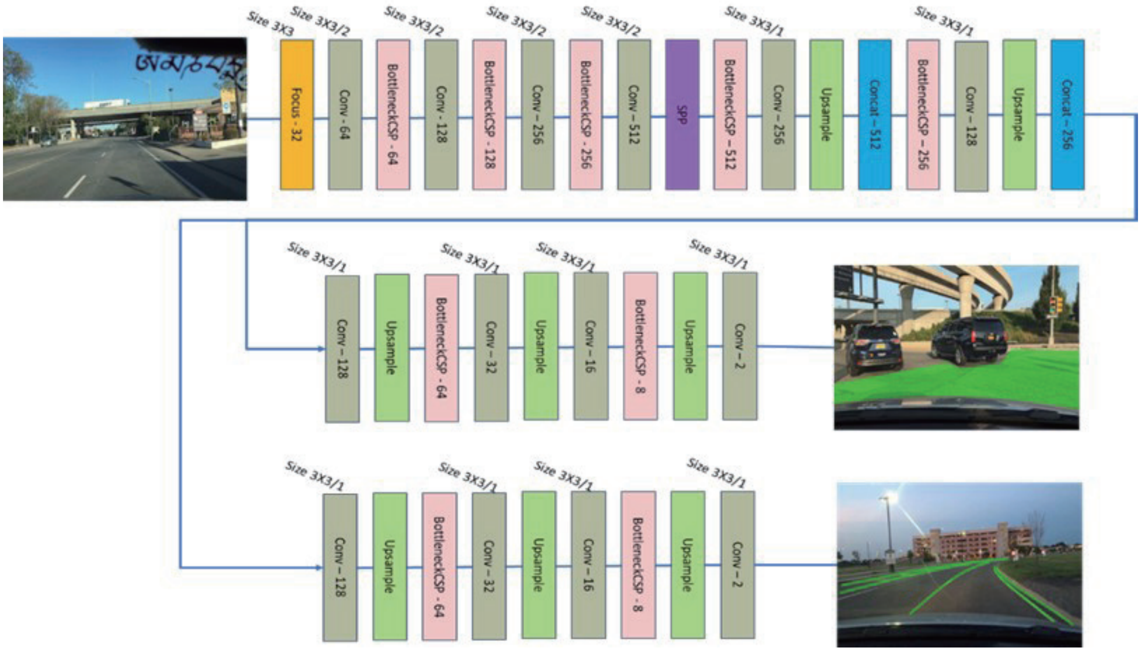


Fig. 2. (Color online) Network architecture of YOLOP model.

2.2 Taguchi method

The application of statistical methods in engineering aims to improve product quality and reduce production costs. The Taguchi method[8] is an experimental design technique that effectively conducts experiments through statistical methods. The Taguchi method not only significantly reduces the number of experiments, manpower, and time costs required, but also improves the quality of the product. The Taguchi method mainly designs the internal parameters of the model (such as design factors and their level values) by selecting an appropriate orthogonal array (OA). The signal-to-noise (S/N) ratio is used to analyze and evaluate the specific impact of different design parameters on model quality, and then determine the optimal parameter combination. The steps of the Taguchi method are shown in Fig. 3.

2.2.1 Problem definition

When this experiment was conducted, we set the initial parameters of the T-YOLOP model. However, the suitability of these parameters for classification applications has not yet been confirmed. Therefore, we used the Taguchi method to optimize the model parameters. Through this optimization process, we not only reduce the number of experiments but also analyze the interactions between different factors. The optimal combination of parameters is determined, which is crucial to improving the stability and accuracy of the model architecture.

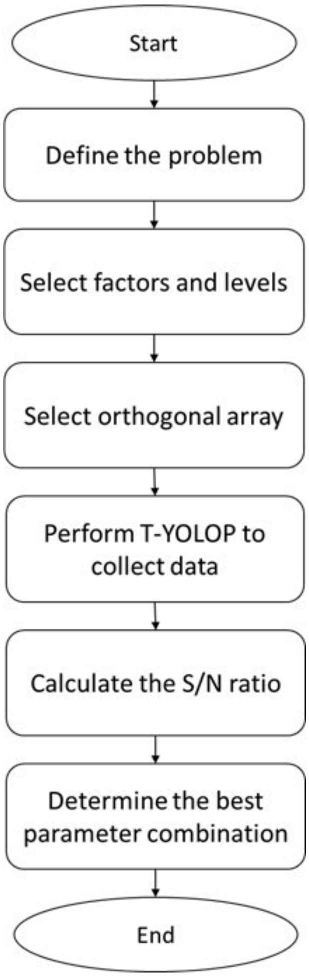

Start
Define the problem
Select factors and levels
Select orthogonal array
Perform T-YOLOP to collect data
Calculate the S/N ratio
Determine the best parameter combination
End

Fig. 3. Steps of Taguchi method.

2.2.2 Selection of factors and levels

In the T-YOLOP model, we focus on the selection of parameters in the output convolutional layer of the network. To improve the overall performance of the model by optimizing these parameters, we identified four main factors: Conv_256, Conv_64, Conv_32, and Conv_8. These factors are selected because they have a significant impact on the model's output and improve the model's feature extraction capability and accuracy at different levels.

On the basis of past experience and prior knowledge, we set three different levels for each factor to explore changes in model performance under different settings. These levels reflect the performance of each factor under different parameter settings, and the impact of each factor on the model output is fully evaluated. Detailed information is shown in Table 1.

Table 1
Impact factors and their levels.

No.	Abbreviation	Factor	Level 1	Level 2	Level 3
A	C1	Conv_256	1	3	5
B	C2	Conv_64	1	3	5
C	C3	Conv_32	1	3	5
D	C4	Conv_8	1	3	5

2.2.3 Choosing a suitable OA

OA is an efficient statistical tool widely used in experimental design and parameter optimization. OAs have uniform distribution and balance properties, which not only ensure that experiments cover a wide range of scenarios, but also significantly reduce experimental costs. Especially when multiple variables and levels should be considered, the application of OAs is particularly important.

2.2.4 Conduct of experiments to collect data

According to Table 1, the L9 OA was selected and experiments were conducted. Then, the average *S*/*N* ratio of each factor at different levels was calculated to evaluate the impact on the experimental results. We use the large characteristic, that is, the larger the *S*/*N* ratio, the greater the impact on the experimental results. The formula for the *S*/*N* ratio is as follows:

$$S/N = -10\log\left(\frac{1}{n}\sum_{i=1}^{n}\frac{1}{y_i^2}\right), \tag{1}$$

where n is the number of repetitions of the experiment and y is the observed value of the ith repeated experiment.

2.2.5 Determination of the best parameter combination

After calculating the *S*/*N* ratio of each of the different parameter combinations, the *S*/*N* response map and the optimal parameter combination were obtained.

3. Experimental Results

To verify the proposed T-YOLOP model, we use the BDD100K data set[11] to conduct experiments. This data set contains a large number of high-quality images and precise annotations. Confusion matrices were used to evaluate the results of classification experiments. Finally, the proposed T-YOLOP model is compared with other models.

3.1 Data set

The BDD100K data set is the largest and most diverse driving image data released by UC Berkeley in 2018. The main characteristics of these data are large scale, diverse, and collected on real streets, as shown in Fig. 4. These image data include ten tasks: image labeling, lane detection, drivable area segmentation, road object detection, semantic segmentation, instance segmentation, multi-object detection and tracking, multi-object segmentation tracking, domain adaptation, and imitation learning. In this study, we used data from the two tasks of lane detection and drivable area segmentation as experimental bases. The total data is 80000 images. We divided the image data into 56000 images as training data and the remaining 24000 images as verification data.

3.2 Evaluation methods

To evaluate the results of classification experiments, we adopted a confusion matrix, as shown in Table 2. The accuracy, true positive (TP), false positive (FP), true negative (TN), and false negative (FN) of the confusion matrix are used to judge the performance of model identification.

When the performance of lane and drivable area markings is evaluated, indicators such as accuracy, intersection over union (*IOU*), and mean *IOU* (*mIOU*) are usually used. Accuracy is a

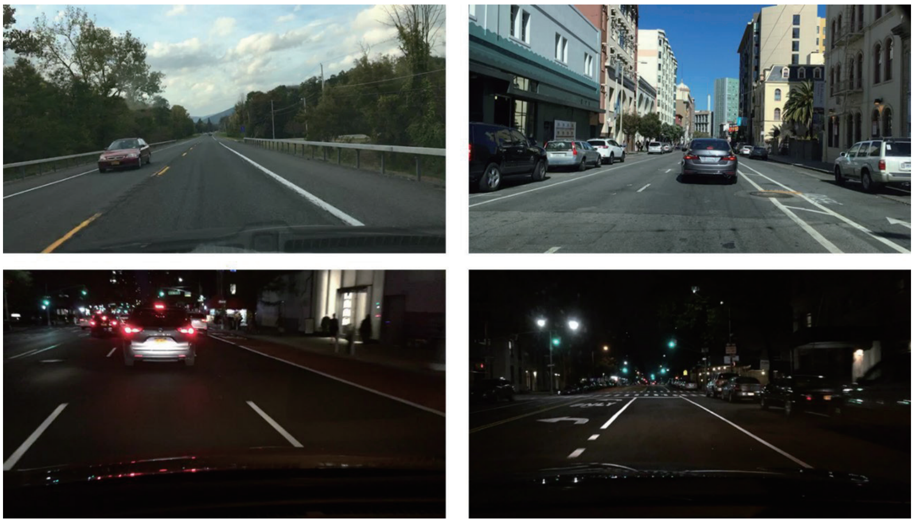


Fig. 4. (Color online) Day and night data sets.

Table 2
Confusion matrix.

		Actual output	
		Positive	Negative
Predictive output	Positive	TP	FP
	Negative	FN	TN

measure of the predictive power of a classification model, which reflects the model's ability to make correct predictions across all test samples. The formula of accuracy is as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}. \tag{2}$$

In the image segmentation task, *IOU* is called the Jaccard index, which is an important indicator for evaluating model performance. It is measured by calculating the ratio of the intersection and union between the binary segmentation results predicted using the model and the actual binary segmentation labels. *IOU* provides an intuitive way to understand the degree of overlap between model predictions and the real situation. *IOU* is calculated by dividing the intersection area of the prediction result and the real label by their union area. The higher the *IOU*, the higher the accuracy of the prediction. The formula for *IOU* is as follows:

$$IOU = \frac{GT}{DR}, \tag{3}$$

where *GT* represents the intersection of prediction and real results, and *DR* represents the union of prediction and real results.

mIOU is a commonly used performance evaluation metric in multicategory image segmentation tasks. It is used to calculate *IOU* for each category separately, and then to average these *IOU* values to obtain a comprehensive performance index. *mIOU* provides a way to measure the overall performance of a model when processing images with multiple categories of objects. It takes into account performance across all categories to make performance evaluations more comprehensive and balanced. This evaluation method is particularly important to understand the consistency and reliability of the model across different categories. *mIOU* is calculated by summing up the *IOU* values of all categories and dividing the sum by the total number of categories. This way, an average can be obtained, which reflects the overall segmentation effect of the model on all categories. The formula for *mIOU* is as follows:

$$mIOU = \frac{class1IOU + class2IOU + \cdots + classNIOU}{N}, \tag{4}$$

where *class* is the category and *N* is the total number of categories.

3.3 Experimental results using the proposed T-YOLOP

In this experiment, we obtained the *S*/*N* ratio of each factor and level for each experiment. The *S*/*N* ratios of the detection results in the drivable areas and lanes are shown in Tables 3 and 4, respectively.

Through the *S*/*N* ratio of each factor and level combination, Tables 5 and 6 indicate the optimal level and optimal parameter combination for drivable area detection and lane detection. If the difference in *S*/*N* ratio is large, the effects of this factor and level are significant. The

Table 3
S/*N* ratio of drivable area detection results.

No.	Factors				Results				
	C1	C2	C3	C4	Y_1	Y_2	Y_3	Y_{ave}	*S*/*N* ratio
1	1	1	1	1	0.883	0.88	0.882	0.881	−1.09394
2	1	3	3	3	0.900	0.898	0.899	0.899	−0.92482
3	1	5	5	5	0.912	0.904	0.911	0.909	−0.82892
4	3	1	3	5	0.906	0.900	0.905	0.903	−0.87994
5	3	3	5	1	0.913	0.91	0.912	0.911	−0.80330
6	3	5	1	3	0.916	0.915	0.913	0.914	−0.77477
7	5	1	5	3	0.917	0.911	0.915	0.914	−0.77801
8	5	3	1	5	0.917	0.915	0.916	0.916	−0.76210
9	5	5	3	1	0.923	0.92	0.922	0.921	−0.70855

Table 4
S/*N* ratio of lane detection results.

No.	Factors				Results				
	C1	C2	C3	C4	Y_1	Y_2	Y_3	Y_{ave}	*S*/*N* ratio
1	1	1	1	1	0.602	0.630	0.631	0.621	−4.14440
2	1	3	3	3	0.656	0.655	0.655	0.655	−3.67076
3	1	5	5	5	0.713	0.712	0.712	0.712	−2.94634
4	3	1	3	5	0.689	0.685	0.688	0.687	−3.25673
5	3	3	5	1	0.695	0.690	0.694	0.693	−3.18546
6	3	5	1	3	0.694	0.690	0.691	0.691	−3.20214
7	5	1	5	3	0.702	0.730	0.731	0.721	−2.84590
8	5	3	1	5	0.710	0.708	0.710	0.709	−2.98302
9	5	5	3	1	0.713	0.712	0.712	0.712	−2.94634

Table 5
Best combination of parameters in drivable area detection.

Factors levels	C1	C2	C3	C4
1	−0.9492	−0.9173	−0.8769	−0.8686
2	−0.8193	−0.8301	−0.8378	−0.8259
3	−0.7496	−0.7707	−0.8034	−0.8237
Delta	0.1997	0.1466	0.0735	0.0449
Rank	1	2	3	4
Best levels	3	3	3	3
Optimal parameter combination	5	5	5	5

Table 6
Best combination of parameters in lane detection.

Factors levels	C1	C2	C3	C4
1	−3.587	−3.416	−3.443	−3.425
2	−3.215	−3.280	−3.291	−3.240
3	−2.925	−3.032	−2.993	−3.062
Delta	0.662	0.384	0.451	0.363
Rank	1	2	3	4
Best levels	3	3	3	3
Optimal parameter combination	5	5	5	5

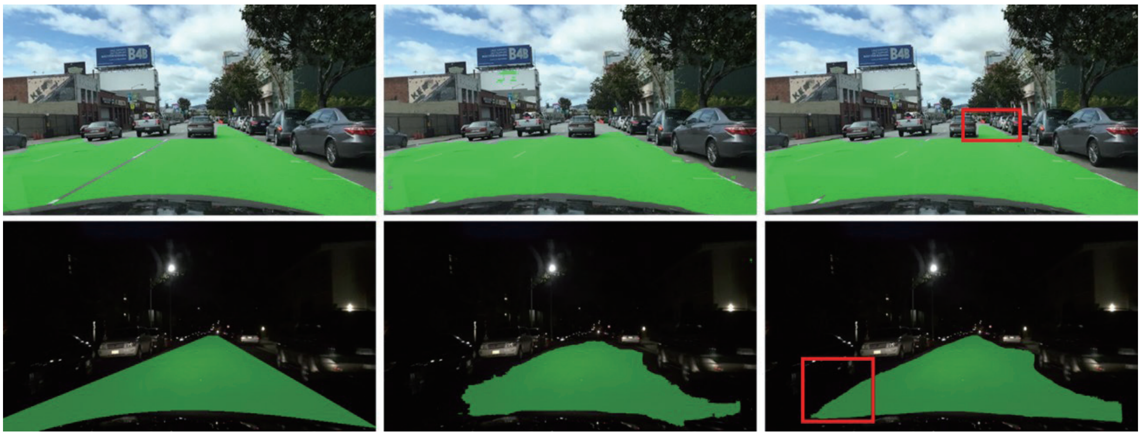
results indicate that the optimal parameter combination for both drivable area detection and lane detection is C1(Conv_256) = 5, C2(Conv_64) = 5, C3(Conv_32) = 5, and C4(Conv_8) = 5.

Table 7 shows the experimental results of the proposed T-YOLOP model with the optimal parameter combination for drivable area detection and lane detection. In Table 7, the evaluation index consists of accuracy, *IOU*, and *mIOU*. The experimental results indicate that the accuracy, *IOU*, and *mIOU* of the proposed T-YOLOP model with the optimal parameter combination are 97.9, 87.5, and 92.8% in the drivable area detection and 73.9, 27.8, and 63.1% in the lane detection, respectively. Compared with the traditional YOLOP model, the accuracy, *IOU*, and *mIOU* of the proposed T-YOLOP model are improved by 0.6, 1.7, and 1.3 percentage points in the drivable area detection and by 4.2, 1.6, and 0.2 percentage points in the lane detection, respectively.

The detection results in the drivable areas and lanes are shown in Figs. 5 and 6, respectively. The proposed T-YOLOP model is significantly better than the traditional YOLOP model. In terms of drivable area detection, we found that the proposed T-YOLOP model is significantly better than the traditional YOLOP model at longer distances and special angles, as shown in Fig. 5. In terms of lane detection, we found that the proposed T-YOLOP model can clearly mark the portion close to the side road or the stop line, as shown in Fig. 6.

Table 7
Comparison results of YOLOP and T-YOLOP.

	Model	*Accuracy* (%)	*IOU* (%)	*mIOU* (%)
Drivable area detection results	YOLOP	97.3	85.8	91.5
	T-YOLOP	97.9	87.5	92.8
Lane detection results	YOLOP	69.7	26.2	62.6
	T-YOLOP	73.9	27.8	63.1



(a) (b) (c)

Fig. 5. (Color online) Drivable area detection results: (a) ground truth, (b) YOLOP, and (c) T-YOLOP.

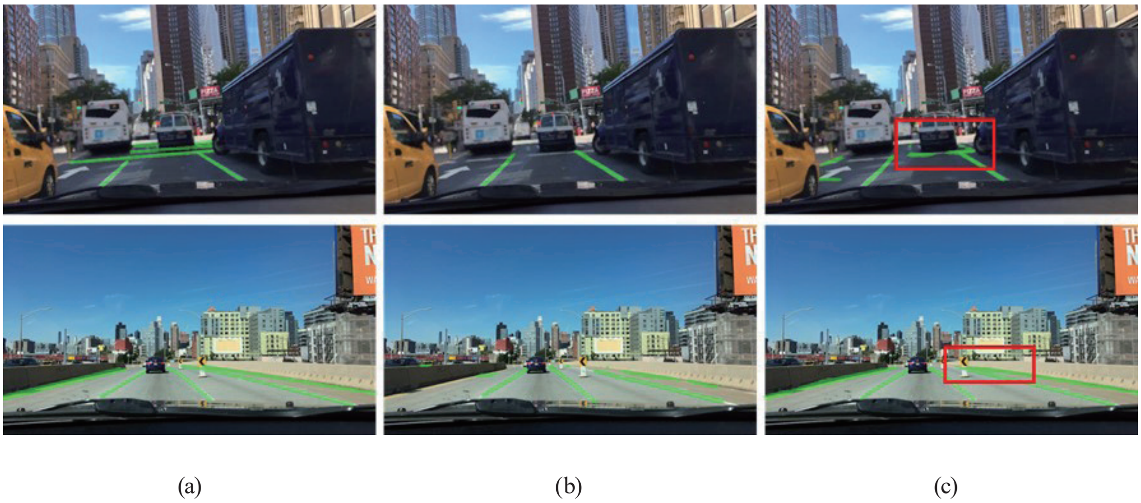


Fig. 6. (Color online) Lane detection results: (a) ground truth, (b) YOLOP, and (c) T-YOLOP.

Table 8
Comparison results of various models in drivable area detection.

Model	*Accuracy* (%)	*IOU* (%)	*mIOU* (%)
MultiNet	—	—	71.6
DLT-Net	—	—	71.3
PSPNet	—	—	89.6
YOLOP	97.3	85.8	91.5
Improved-YOLOP	97.6	87	92.1
T-YOLOP	97.9	87.5	92.8

Table 9
Comparison results of various models in lane detection.

Model	*Accuracy* (%)	*IOU* (%)	*mIOU* (%)
ENet	34.1	14.6	34.1
SCNN	35.8	15.8	35.7
ENet-SAD	36.6	16.0	36.5
YOLOP	69.7	26.2	62.6
Improved-YOLOP	73.6	27.5	62.9
T-YOLOP	73.9	27.8	63.1

3.4 Comparison results with other models

In this subsection, we compare the proposed T-YOLOP model with other models in the drivable area detection and lane detection. In the drivable area detection, the proposed T-YOLOP model is compared with some models, such as MultiNet,(12) DLT-Net,(13) PSPNet,(14) and YOLOP.(6) Comparison results of various models in the drivable area detection are shown in Table 8. Experimental results indicate that the accuracy, *IOU*, and *mIOU* of the proposed T-YOLOP model outperform those of the other models.

In the lane detection, the proposed T-YOLOP model is compared with some models, such as ENet,(15) SCNN,(16) ENet-SAD,(17) and YOLOP.(6) In the BDD100K data set, each lane is usually marked by two lines on its left and right sides, but this marking method is not ideal for direct application to the detection model. Therefore, our method first calculates the center line of the lane on the basis of these two annotation lines, and then to better simulate actual road conditions, the width of the lane is set to 8 pixels, which improves the accuracy of detection and generalization ability. Comparison results of various models in the lane detection are shown in Table 9. Experimental results indicate that the accuracy, *IOU*, and *mIOU* of the proposed T-YOLOP model outperform those of the other models.

4. Conclusions

In this study, we proposed a T-YOLOP model to improve the accuracy and computing speed of the model in detecting drivable areas and lanes, making it a more practical panoptic driving perception system. In the T-YOLOP model, the Taguchi method was used determine the appropriate parameter combination. Experimental results indicated that the accuracy, *IOU*, and *mIOU* of the proposed T-YOLOP model with the optimal parameter combination are 97.9, 87.5, and 92.8% in the drivable area detection and 73.9, 27.8, and 63.1% in the lane detection, respectively. Compared with the traditional YOLOP model, the accuracy, *IOU*, and *mIOU* of the proposed T-YOLOP model are improved by 0.6, 1.7, and 1.3 percentage points in the drivable area detection and by 4.2, 1.6, and 0.2 percentage points in the lane detection, respectively.

References

1 S. Aggarwal, A. M. Namboodiri, and C. V. Jawahar: 2014 22nd Int. Conf. Pattern Recognit. (2014) 4275. http://doi.org/10.1109/ICPR.2014.733

2 J. Long, E. Shelhamer, and T. Darrell: 2015 IEEE Conf. Comput. Vis. Pattern Recognit. (2015) 3431. http://doi.org/10.1109/CVPR.2015.7298965

3 J. Canny: IEEE Trans. Pattern Anal. Mach. Intell. **8** (1986) 679. http://doi.org/10.1109/TPAMI.1986.4767851

4 Y. Dong, J. Xiong, L. Li, and J. Yang: 2012 Int. Conf. Computational Problem-Solving (2012) 461. http://doi.org/10.1109/ICCPS.2012.6384266

5 C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao: 2021 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (2021) 13024. http://doi.org/10.1109/CVPR46437.2021.01283

6 D. Wu, M. Liao, W. Zhang, and X. Wang: arXiv (2021). https://arxiv.org/abs/2108.11250

7 C.-C. Huang: Master thesis of Department of Electric Engineering, Formosa University (2023). https://hdl.handle.net/11296/ucjqj9

8 L. Yu, J. Chen, G. Ding, Y. Tu, J. Yang, and J. Sun: IEEE Access **6** (2018) 45923. http://doi.org/10.1109/ACCESS.2018.2864222

9 T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie: 2017 IEEE Conf. Comput. Vis. Pattern Recognit. (2017) 936. http://doi.org/10.1109/CVPR.2017.106

10 K. He, X. Zhang, S. Ren, and J. Sun: IEEE Trans. Pattern Anal. Mach. Intell. **37** (2015) 1904. http://doi.org/10.1109/TPAMI.2015.2389824

11 F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell: 2020 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (2020) 2633. http://doi.org/10.1109/CVPR42600.2020.00271

12 M. Teichmann, M. Weber, M. Zöllner, R. Cipolla, and R. Urtasun: 2018 IEEE Intelligent Vehicles Symp. (2018) 1013–1020. http://doi.org/10.1109/IVS.2018.8500504

13 Y. Qian, J. M. Dolan, and M. Yang: IEEE Trans. Intell. Transp. Syst. **21** (2020) 4670. http://doi.org/10.1109/TITS.2019.2943777

14 H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia: 2017 IEEE Conf. Comput. Vis. Pattern Recognit. (2017) 6230. http://doi.org/10.1109/CVPR.2017.660

15 A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello: arXiv (2016). https://arxiv.org/abs/1606.02147

16 X. Pan, J. Shi, P. Luo, X. Wang, and X. Tang: Proc. AAAI Conf. Artificial Intelligence **32** (2018) 7276. https://dl.acm.org/doi/10.5555/3504035.3504926

17 Y. Hou, Z. Ma, C. Liu, and C. C. Loy: 2019 IEEE/CVF Int. Conf. Comput. Vis. (2019) 1013. http://doi.org/10.1109/ICCV.2019.00110