Proceedings of the 2019 International Conference on Advanced Mechatronic Systems, Kusatsu, Shiga, Japan, August 26 - 28, 2019

A Lightweight, High-Performance Multi-Angle License Plate Recognition Model

Cheng-Hung Lin
Dept. of Electrical Engineering
National Taiwan Normal University
Taipei, Taiwan
brucelin@ntnu.edu.tw

Chen-Hao Wu
Dept. of Electrical Engineering
National Taiwan Normal University
Taipei, Taiwan
iamphilip526@gmail.com

***Abstract*—On the streets of Taiwan, many roadside tollers are often seen riding motorcycles in one hand, and the other hand holding mobile devices to issue payment notices for cars and motorcycles parked on the roadsides. The work of roadside tollers is very dangerous. First, they must first park their motorcycles next to the roadside cars and motorcycles. They then use their eyes to confirm the license plate number, enter the license plate number into the mobile device, and finally place the bill on the car's windows or attach the bill to the motorcycles. Our idea is to implement an automated license plate recognition system in mobile devices to increase the efficiency of roadside tollers and reduce their time on the road.**

Recently, license plate recognition systems have been widely used in various aspects of life, such as parking lot toll systems, access management systems, and traffic management systems. However, existing license plate recognition systems must have good recognition rates under a number of constraints, such as fixed angles and fixed light sources. Moreover, due to the insufficient computing resources of the general mobile device, the application cannot have a good recognition rate in the complex environment or skewed angle in the license plate recognition. Therefore, this paper proposes a lightweight and high-performance multi-angle license plate character recognition model, which reduces the complexity and computational complexity of traditional license plate recognition. This paper also collects a large number of license plate images from different environments, angles and sizes as training data. Finally, we propose an optimized deep learning model to identify the characters on license plates. The experimental results show that the proposed model can recognize the license plate with a tilt of 0~60 degrees, and the overall recall rate is 84.5%. Compared with Tiny-YOLOv2, the computation of the proposed model is reduced by 61% with a little penalty of recall.

Keywords—license plate recognition system; deep learning model; lightweight; multi-angle

I. INTRODUCTION

With the changes in the environment and the development of science and technology, the density of vehicles in the world is getting higher and higher, and vehicle management is gradually receiving attention. The traditional manual vehicles management system not only requires labor, but also when the flow rate of incoming and outgoing vehicles is large, it is easy to affect the entry and exit of vehicles due to the speed of recording, thereby causing traffic problems. The modern management method takes advantage of the fast processing speed and low error rate of the computer, and has developed an automatic license plate recognition system to replace the traditional method.

Automatic license plate recognition systems have been widely used in various aspects of life, such as parking lot toll systems, access management systems, and traffic management systems. At present, most of the license plate recognition systems [1] divide the system into four stages, as shown in Fig. 1. The first stage is to input the image file captured by the photographic equipment. The second stage is to find the position of the license plate in the image and then extract the license plate from the image. The third stage is to segment the characters on the license plate by projecting color information or labeling. The fourth stage is to use the template comparison method or the classifier to perform character recognition for the divided character images.

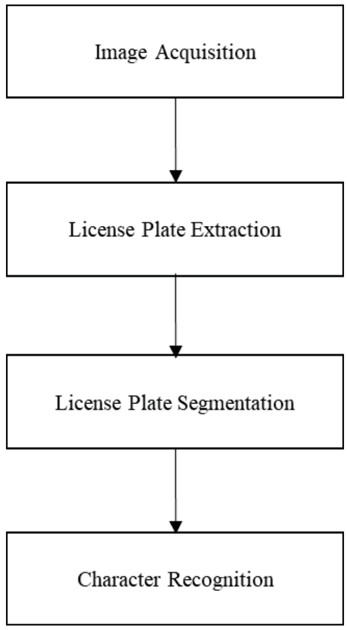

Fig. 1. The traditional four stages of a license plate recognition system

However, the current development of the license plate recognition system is limited by the inability to have a good recognition rate in complex environments or skewed angles, and requires a large amount of computing resources and energy consumption [2] [3]. Implementing a license plate recognition system on low-computing mobile devices has become a key issue.

Therefore, the purpose of this study is to propose a lightweight, high-performance multi-angle license plate recognition model that enables the license plate recognition system to have a good recall rate under different environments and angles as shown in Fig. 2.

Fig. 2. License plate recognition example

The following are the challenges encountered in this study as shown in Fig. 3.

A. Complex background environment

The images taken for a single license plate will also contain other information about the car body. However, additional objects on the car body, such as body advertisements and body mosaics, are likely to cause misjudgment of the identification model.

B. License plates diversity

Take Taiwan's license plate as an example. The background color includes red, green, yellow and white. Character also includes different colors such as red, black, and green. The license plate has four to seven characters. Combining the above, the style of the license plate has become quite diverse. And there are additional conditions such as dirty license plates, paint drops, inspection stickers, hanging dolls, advertisements, etc.

C. Light source problem:

The presence or absence of a light source, the angle of the light source, and changes in the weather can cause image differences, which increases the difficulty of recognition.

D. Shooting angle

Different shooting angles can cause characters to skew and affect the recognition rate of the model.

E. Approximate characters

Some of the characters have very similar appearances, and they are more likely to cause false positives under skewed angles. For example: 0 and D, R and Q, 8 and B, A and 4.

This paper proposes a lightweight and high-efficiency multi-angle license plate recognition model, which reduces the complexity of the traditional license plate recognition system, and optimizes the deep learning model Tiny-YOLOv2 [5] to identify the characters on license plates.

The experimental results show that the proposed model can recognize the license plates with a slope of 0~60 degrees, and the recall rate in the range of 0~60 degrees is 89.5%. Compared with the Tiny-YOLOv2, the calculation complexity of the proposed model is reduced by 60%, but the recall rate is slightly decreased.

(a) Complex background

(b) License plates diversity

(c) Light source problem

(d) Shooting angle

(e) Approximate characters

Fig. 3. Example image of the challenge

II. RELATED WORKS

A. Character Recognition

A variety of research methods have been proposed for license plate character recognition, which can be summarized into two categories. The following will introduce the technology of license plate character recognition in the current literature.

1) Template matching methods

The template matching method first needs to prepare a standard template database for comparing the characters to be recognized in advance, and can calculate the difference between the input characters and the template characters. The character that is most similar to the standard template in the database is the identification result.

The template matching method is simple and intuitive, and there is no excessive image processing or feature extraction. Therefore, external interference factors must be reduced during the processing to ensure the stability of the comparison. Most studies convert the characters to be recognized into grayscale and binarize them before the comparison, thereby reducing the noise impact on the character image.

The template comparison method takes a lot of time to process unimportant pixels and easily affects the accuracy due to positioning problems. Moreover, the template matching method does not easily recognize the rotation, tilt and break of characters, and is susceptible to image noise, incomplete characters, etc., thereby reducing the accuracy of recognition.

In [6] [7], to facilitate comparison of the template, the input characters are resized to match the template; In [8], the error root mean square values between the input character and each character template are calculated and the one with the smallest error is the identification result;

2) Feature matching methods

The feature matching methods use machine learning techniques to extract features that are compared with pre-stored features to measure similarity, and then to obtain identification results.

Compared with the template matching method, the feature matching method has the advantages of time-efficiency, adaptability, nonlinear processing, parallel operation, anti-noise and the like. In addition, the feature matching method still has a good recognition rate for skew, dirt, distortion, and incomplete characters. However, the feature matching method requires a large number of training samples to improve the recognition rate, and requires a large amount of time to train the network weights, and the feature types of the training samples will affect the recognition accuracy [9].

B. Object detection model

The object detection model used in this study will be introduced below.

1) YOLO (You Only Look Once)

YOLO [4] is the first one stage object detection architecture which has the advantage of using only one CNN architecture to process images without the need to extract candidate frames. YOLO considers the prediction of the bounding box and the class of the object as a regression problem, and uses only a single CNN network to predict the outcome. YOLO first divides the image into SxS grid, each grid will predict the B bounding box coordinates, confidence value, and the probability of C object categories, and finally get the best bounding box through non-maximum suppression, and the category of the object. YOLO is a one-stage architecture with the advantages of lightweight models and fast detection, but also has the disadvantage of low positioning accuracy and poor detection of small objects.

2) YOLOv2

YOLOv2 has made many improvements based on YOLO, including adding batch normalization after all convolutional layers, which improves the convergence speed of the model and avoids overfitting. When training a classification network, YOLO trained 160 epochs using a classified network with an input of 224x224. In YOLOv2, in addition to the original YOLO, the input resolution was increased to 448x448 and 10 epochs were trained. This method preserves the message of small objects and improves the accuracy of YOLO to identify small objects. Referring to the method of Faster R-CNN, not only all the fully connected layers are removed, but also the concept of the anchor is imported into YOLO, which corrects the problem of poor positioning accuracy in YOLO.

In the selection of the anchor box, in addition to improving the originally selected anchor box, the K-means Cluster method is used to find the anchor box that is most suitable for the dataset. It also improves the prediction method of the anchor box, improving the stability of the overall model. The passthrough layer has been added to solve the problem that YOLO has poor detection effect on small objects. Multi-scaling training is used to improve the stability of the model at different resolutions.

Through the above improvements, YOLOv2 has a faster detection speed and higher accuracy than YOLO, but it also slightly increases the amount of calculation.

III. OPTIMIZED CHARACTER RECOGNITION MODEL

In this paper, the optimized one-stage object detection Tiny-YOLOv2 model is used for character recognition model to realize lightweight and high-performance multi-angle license plate recognition model, as shown in Table I. In order to make a better trade-off between the calculation and recall rate of the model, we have made the following optimizations.

A. Increase the number of grids.

YOLOv2 will first resize the image of the PASCAL VOC dataset [10] to 416x416 pixels and then cut it into 13x13 Grid. The size of each grid corresponding to the original picture pixel is 28x28 to 38x38. In this paper, the original picture pixel is 1920x1080. If the image is adjusted to 416x416 pixels and then cut into a 13x13 grid, the size of each grid corresponding to the original image pixel will be 148x83. However, the characters we want to recognize occupy the original image with a pixel size of 50x65 to 300x800. In order to make the model recognize smaller characters, we increase the number of grids from 13x13 to 40x20, so that the original image pixel size corresponding to each grid is reduced to 48x54, so the model has a good recall rate when recognizing smaller characters.

B. Increase the number of anchor boxes.

The anchor box used in Tiny-YOLOv2 is obtained by K-means analysis based on PASCAL VOC. The problem to be dealt with in this paper is the recognition of license plate characters at different distances and angles. Therefore, we use K-means to analyze our dataset and find the anchor box coordinates that best fit our dataset, making it easier for the model to learn how to predict the bounding box. And for the license plate characters at different distances and angles, the number of anchor boxes is increased from 5 to 9, which makes the recall rate of the model increase.

C. Optimization model for license plate characters

The background environment in this study is not complicated enough to use so many filters, and too many filters will increase the amount of computation and unnecessary feature extraction. Therefore, this paper optimizes the network architecture of Tiny-YOLOv2, including removing the 13th convolution layer, and changing the number of filters in the 9th convolution layer to 128, and changing the number of filters in the 11th convolution layer to 256. We reduce the number of filters so that there is a better trade-off between recall and computation. Table II shows the architecture of the proposed model.

Table I. Tiny-YOLOv2 architecture

Layer	Filter	Size/Stride	Output
Convolutional	16	3*3/1	416*416*16
Maxpool		2*2/2	208*208*16
Convolutional	32	3*3/1	208*208*32
Maxpool		2*2/2	104*104*32
Convolutional	64	3*3/1	104*104*64
Maxpool		2*2/2	52*52*64
Convolutional	128	3*3/1	52*52*128
Maxpool		2*2/2	26*26*128
Convolutional	256	3*3/1	26*26*256
Maxpool		2*2/2	13*13*256
Convolutional	512	3*3/1	13*13*512
Maxpool		2*2/1	13*13*512
Convolutional	1024	3*3/1	13*13*1024
Convolutional	512	3*3/1	13*13*512
Convolutional	195	1*1/1	13*13*195
Detection			

Table II. Optimized character recognition model architecture

Layer	Filter	Size/Stride	Output
Convolutional	16	3*3/1	1280*640*16
Maxpool		2*2/2	640*320*16
Convolutional	32	3*3/1	640*320*32
Maxpool		2*2/2	320*160*32
Convolutional	64	3*3/1	320*160*64
Maxpool		2*2/2	160*80*64
Convolutional	128	3*3/1	160*80*128
Maxpool		2*2/2	80*40*128
Convolutional	128	3*3/1	80*40*128
Maxpool		2*2/2	40*20*128
Convolutional	256	3*3/1	40*20*256
Maxpool		2*2/1	40*20*256
Convolutional	512	3*3/1	40*20*512
Convolutional	312	1*1/1	40*20*390
Detection			

IV. EXPERIMENTAL RESULTS

In this paper, the equipment used in the experiments are as follows: CPU is Intel Core i7-4790, GPU is NVIDIA GeForce GTX TITAN, and operating system is Linux Ubuntu 16.04 LTS.

The test object is the license plates of general vehicles in Taiwan. The license plate number in Taiwan is the combination of English letters A~Z (excluding I and O) and the numbers 0~9, for a total of 34 characters. Each license plate has a minimum of 4 characters and a maximum of 7.

The background color of the license plate is white, yellow, red, or green. The color of the character is black, white, or red. However, since the actual number of license plates issued varies greatly depending on the number of vehicles, black and white license plates are still the majority.

This study collects and screens data according to the working environment and perspective of roadside toll collectors. Considering that roadside toll collectors will take license plates in different weather conditions, the environment for shooting images will be divided into indoor, outdoor and rainy, and data will be collected from actual roads, open parking lots and underground parking lots. Image data with different weather, background, exposure, elevation angle, rotation angle, and license plate image ratio is used as our data sets.

The training data set of this paper has a total of 5,403 license plate images, and the testing data has a total of 600 license plate images. The examples of our data set are shown in Fig. 3. The image parameters are shown in Fig.4 [12], and the image parameter statistics are shown in Table III.

Fig. 3. Examples in our dataset.

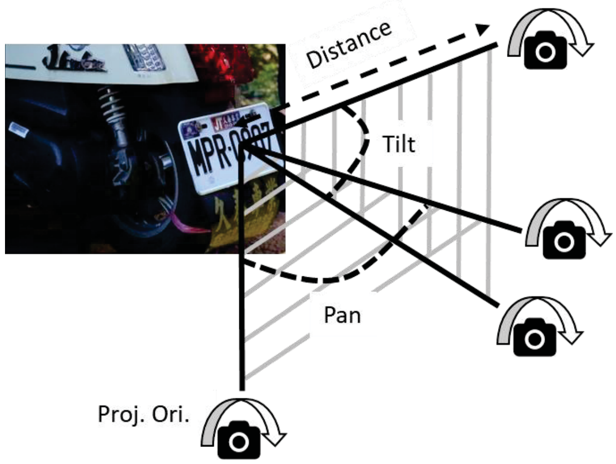

Fig. 4. Distance determines the size of license plate in an image, pan and tilt determine the out-of-plane rotation of the plate projected onto the image, and orientation refers to the in-plane rotation of the view.

Table III. Statistics of three category parameters

	Indoor	Outdoor	Rainy
Pan	75°~-75°	75°~-75°	75°~-75°
Tilt	0°~75°	0°~75°	0°~75°
Width Ratio	0.3~0.95	0.3~0.95	0.3~0.95
Distance	< 1m	< 1m	< 1m
Proj. Ori.	<60°	<60°	<60°
# of Samples	442	4739	822

A. *Anchor Box experiment*

In the experiment, we used different numbers of anchor boxes to train the proposed model, and used the collected data sets to calculate the computational load and recall rate as the basis for the evaluation. The experimental results are shown in Table IV. In Table IV, The recall rate is the best in the number of anchor boxes of 8, and the amount of calculation does not increase too much.

Table IV. Comparison of Recall rate and BFLOPS under different anchor box numbers

Number of anchor boxes	Recall rate	BFLOPS
6	85.3%	9.891
7	83.5%	9.923
8	89.5%	9.955
9	88.9%	9.986
10	85.7%	10.018
11	87.7%	10.05

B. *Character recognition experiment*

In this experiment, we used the Tiny-YOLOv2 model to train with our dataset under the same conditions, and compared mAP rate, computational load, and inference time with our model. The experimental results show that the proposed model reduced the computational complexity by 61%, and the processing time by 30%, but the mAP was only reduced by 0.1%, as shown in the Table V.

Table V. Comparison of mAP, FLOPS and Inference time

	Tiny-YOLOv2	The proposed model	Improvement
mAP	99.63%	99.52%	- 0.1 %
BFLOPS	25.559	9.955	61.1 %
Inference time	13 ms	9 ms	30.8 %

C. *Multi-distance and multi-angle experiments*

This experiment is a study on the design of license plate recall rate for different angles and different distances. we used the Tiny-YOLOv2 model to train with our dataset under the same conditions, and compare the recall rate at different angles and distances with our model. The experimental results are shown in the Table VI.

In the experiment, the shooting angle starts from the normal direction of the lens, which is aligned with the normal direction of the center of the license plate, and then traverses every 15 degrees to 75 degrees, with a total of 5 intervals. In Fig. 4., pan denotes the shooting angle. In Fig. 4. , the distance is the vertical distance between the lens and the license plate center, including 50cm, 100cm, 150cm and 200cm. Each part contains left and right oblique data, each data is 50%. A total of 4,129 photos were used in this experiment, and the total number of characters was 28,903. Since the distance affects the proportion of the license plate in an image and thus affects the recall rate, we have added the variation of the distance to the experiment.

In addition, the closer the distance is, the more obvious the effect of the angle change on the recognition effect. For example, in the case of a distance of 50 cm, the skew of the character will be more obvious than when the distance is 200 cm. But, the proportion of characters will be much larger than 200cm, which causes the recall rate to be seriously affected.

The proposed model has 9.955 BFLOPS while the Tiny-YOLOv2 has 25.559 BFLOPS. In the range of 100 cm and 0 to 60 degrees, the average recall rate achieves 89.625%. The computation is reduced by 61% and inference time is reduced by 30% with a little penalty of recall.

Table VI. Comparison of recall rate of each angular range

Model	Tiny-YOLOv2				The proposed model				Improvement
Distance / Angle	50cm	100cm	150cm	200cm	50cm	100cm	150cm	200cm	
0° ~15°	99%	100%	100%	95%	91%	100%	100%	98%	-1.2%
15° ~30°	96%	100%	98%	81%	92%	99%	97%	91%	1.0%
30° ~45°	91%	97%	90%	49%	87%	98%	85%	70%	3.9%
45° ~60°	68%	75%	47%	14%	67%	83%	51%	29%	12.7%
60° ~75°	30%	30%	10%	1%	45%	34%	21%	7%	50.7%

V. CONCLUSION

In this paper, we have proposed a lightweight and highly efficient multi-angle license plate recognition model. The experimental results show that our proposed model has a recall rate of 84.5% at 0 to 60 degrees and 61% less computational load than Tiny-YOLOv2.

ACKNOWLEDGMENT

This work was financially supported by the “Chinese Language and Technology Center” of National Taiwan Normal University (NTNU) from The Featured Areas Research Center Program within the framework of the Higher Education Sprout Project by the Ministry of Education(MOE) in Taiwan, and Ministry of Science and Technology, Taiwan, under Grants no. MOST 108-2634-F-003-002 and MOST 108-2634-F-003-003 through Pervasive

Artificial Intelligence Research (PAIR) Labs. We are grateful to the National Center for High-performance Computing for computer time and facilities to conduct this research.

REFERENCES

[1] S. Du, M. Ibrahim, M. Shehata and W. Badawy, "Automatic License Plate Recognition (ALPR): A State-of-the-Art Review," IEEE Trans. Circuits and Systems for Video Technology, vol. 23, no. 2, pp. 311-325, Feb. 2013.

[2] L. Connie, C. Kim On and A. Patricia Patricia, "A Review of Automatic License Plate Recognition System in Mobile-based Platform," Journal of Telecommunication, Electronic and Computer Engineering (JTEC), Vol 10, No 3-2, pp. 77-82, 2018.

[3] N. N. Kyaw, G. R. Sinha and K. L. Mon, "License Plate Recognition of Myanmar Vehicle Number Plates A Critical Review," IEEE 7th Global Conference on Consumer Electronics (GCCE), Nara, pp. 771-774., 2018.

[4] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016, pp. 779-788.

[5] J. Redmon, and A. Farhadi, "YOLO9000:Better, Faster, Stronger", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, Hawaii, 2017, pp.1-9.

[6] V. Shapiro and G. Gluhchev, "Multinational license plate recognition system: Segmentation and classification," in Proc. Int. Conf. Pattern Recognit., vol. 4. 2004, pp. 352–355.

[7] Y. P. Huang, S. Y. Lai, and W. P.Chuang, "A template-based model for license plate recognition," in Proc. IEEE International Conference on Networking,Sensing and Control, Vol. 2, pp. 737-742, March. 2004.

[8] M. A. Ko and Y. M. Kim, "License plate surveillanc system using nd weighted template matching," in Proc. The 32 Applied Imagery Pattern Recognition Workshop, pp. 269-274, Oct. 2003.

[9] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and F.F. Li "ImageNet: A Large-Scale Hierarchical Image Database," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2009, pp. 248–255

[10] M. Everingham, L. Van, C. K. I. Williams, J. Winn, A. Zisserrnan, "The pascal visual object classes (voc) challenge", International Journal of Computer Vision, vol. 88, no. 2, pp. 303-338, 2010.

[11] FFmpeg Developers. (2016). ffmpeg tool (Version be1d324) [Software].Available from http://ffmpeg.org/

[12] G. Hsu, J. Chen, and Y. Chung, "Application-oriented license plate recognition," IEEE Trans. Veh. Technol.,vol. 62, no. 2, pp. 552–561, 2013.