MULTI-PASS YOLOV3 FOR SCOOTER LICENSE PLATE DETECTION AND RECOGNITION

[1]*Ming-Ching Chang* (張明清), [2]*Jun-Wei Hsieh* (謝君偉), [3*]*Chun-Ming Tsai* (蔡俊明), [3]*Zong-Yi Lin* (林宗易)

[1] Department of Computer Science, University at Albany, State University of New York, USA

[2] College of Artificial Intelligence and Green Energy, National Chiao Tung University, Taiwan

[3] Department of Computer Science, University of Taipei, Taiwan

E-mail: mchang2@albany.edu, jwhsieh@nctu.edu.tw, cmtsai@go.utaipei.edu.tw, james203198191@gmail.com

ABSTRACT

We present a multi-pass approach for scooter Automatic License Plate Recognition (ALPR). Scooters are important daily vehicles in Southeast Asian countries due to their superior mobility, convenience, and low cost. Automatic scooter License Plate (LP) detection and recognition are important for transportation administration and law enforcement. Existing ALPR works relying on DNN and data augmentation are deficient against challenges due to large view variations, motion blur, day/night and lighting variations. Our multi-pass YOLOv3 object detection design can effectively detect LPs, rectify LP views via detecting the LP screws and corners, and then recognize LP characters, all using only a single YOLOv3 network. Our method can detect far-away plates in large view angles and recognize tiny characters. Evaluation is performed on a new TaiwanScooterLP dataset with 3,154 scooter images, 4,198 labeled LPs and 25,700 LP characters. Our method achieves LP detection F1 of 98.31% and character recognition F1 of 99.88%. Our system runs in about 10 FPS (excluding I/O) and can handle low lighting conditions.

Keywords: *Multi-pass Yolov3, Scooter license plate detection, Scooter license plate recognition, large view angle, TaiwanScooerLP dataset, low lighting.*

1. INTRODUCTION

Scooters are important daily vehicles in Southeast Asian countries due to their superior mobility, convenience, and low cost. License Plate (LP) detection and recognition for scooters and vehicles are important means of vehicle identification for Intelligent Transportation System (ITS), parking lot automation, and law enforcement use cases [17]. With the recent breakthrough in computer vision and deep learning, it is now possible to run Automatic License Plate Recognition (ALPR) on a hand-held device over a moving platform, i.e. just scan a street scene and all LPs there are read automatically. Unconstrained ALPR system that can be used such way in-the-wild is still under development.

ALPR for vehicle identification has long development history [7], however mostly for cars and not specifically for scooters or motorcycles. Scooter LPs are smaller in size and scooters can appear more crowdedly together, which makes the detection harder. Large view variations, clutter background, and environmental factors such as lighting conditions, motion blur all make the problem challenging. ALPR approaches relying on vehicle detection to localize LPs [6, 5, 16, 15] do not leverage the strong shape/appearance cues — License plates are rectangular, usually white or in constant backgrounds and black texts, thus they are easier to detect using a light-weight network. The Deep Neural Networks (DNN) used in most modern ALPR methods generally require large training data with data augmentation to boost detection performance, while the brute-force end-to-end design are still deficient against the aforementioned challenges. Non-maximum-suppression (NMS) of the detectors can wrongly remove true detections.

In this paper, we follow a multi-stage design for ALPR (Fig.1). In the first stage, LPs are detected and localized. View variations are handled and LP images are rectified in the second stage. Finally, character detection, segmentation, and recognition are performed in the third stage. We propose a multi-pass YOLOv3 [22] object detection scheme that can effectively and robustly carry out all three stages, namely: (1) detect LPs, (2) rectify LP views from large viewing angles via detecting the LP screws and corners, and then (3) recognize LP contents, all using a single YOLOv3 network. Our multi-pass design is unique that only one YOLOv3 model is used, thus it can run fast on edge devices such as the nVidia Jetson Nano. It can detect far-away plates in large viewing angles (about 50 degrees) and recognize tiny characters (up to 5 pixels wide). Thanks to the multi-scale capability of YOLOv3 in detecting tiny objects, our method can detect LPs directly without vehicle detection. It can handle multi-languages as long as sufficient training samples are provided.

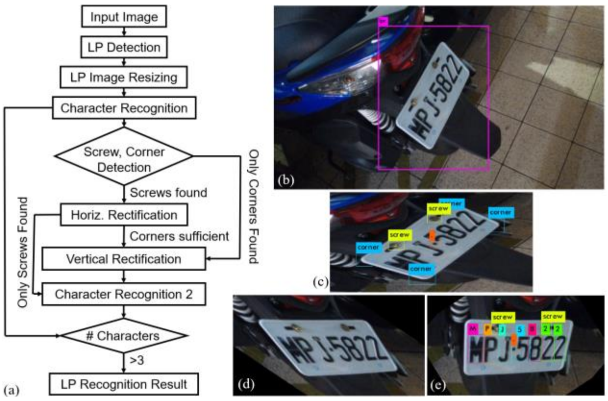

Fig. 1: Overview. (a) The proposed multi-pass YOLOv3 pipeline for scooter ALPR. (b) Initial LP detection. (c) Screw and corner detection for view rectification. (d) After horizontal rectification. (e) After vertical rectification, where the LP characters are detected and recognized. All detections and recognition are performed using only a single YOLOv3 network.

We provide a new TaiwanScooterLP dataset for scooter ALPR model training and performance evaluation. The dataset combines scooter images from our in-house collections in Taiwan and those from the Application-Oriented License Plate (AOLP) dataset [11, 31]. It consists of 3,154 scooter images, 4,198 labeled LPs and 25,700 labeled LP characters. Performance evaluation is reported in precision-recall analysis and F1 scores. Our method achieves LP detection F1 of 98.31% and character recognition F1 of 99.88%. Our system runs in about 10 FPS (excluding I/O) and can handle day/night and low lighting conditions. Without trained on car LPs (in different formats), they can be detected as well as in Fig.3.

2. BACKGROUND

Modern DNN object detectors in most ALPR systems can be organized into two categories [13]: (a) two-stage detectors: Faster-RCNN, Mask-RCNN [9] based on region proposal base networks (such as ResNet, Google Inception v3, MobileNet), and (b) single-stage detectors: SSD, DSSD, YOLO.

Traditional ALPR methods rely on computer vision techniques (Sobel or Canny edges [18], binarization [26], Otsu threshold, morphology, filtering, connected components, MSER [11] of texts, to detect LPs based on shape and color features. To handle large view angles, Radon [18] or Hough [14] transforms are used to estimate the horizontal and vertical tilting angles, then affine transform [28] can rectify the LP. In [30], Hough line scans are used for LP corner detection, and then homography is estimated for perspective correction. In [23], morphology, filtering, adaptive thresholding, contour and geometric filtering are used to find ROI that CNN classification can identify LP texts.

Modern ALPR methods are mostly based on deep object detectors. Typically multiple networks are used for various purposes and in separate stages. In [19], SSD is used for vehicle detection, and MobileNet based on depthwise separable conv is used for reading the Thai characters. A large LP dataset called CCPD is introduced in [29]. In [24], Arabic LP is detected based on Mask-RCNN. In [8], a real-time multi-task network detects Brazilian LPs. In [2], LP is detected based on semantic segmentation. Scooter LP tracking by detection is studied in [12].

YOLO based ALPR. In recent years, there are increasing amount of ALPR works [10, 15, 25, 3, 4, 16, 27] based on the YOLO detectors [20, 21, 22] due to their superior performance and light-weight network. Customization and augmentation of the original YOLO, such as network architecture change [10], hyper-parameter tuning [4], data augmentation [15], are required in order to successfully adapt YOLO toward a solution for ALPR. Several YOLO-based ALPR systems are multi-stage in design, some combines additional text detector network for OCR [3], or use multiple YOLO networks in [1, 16]. In comparison, our multi-pass design is unique that only a single YOLO network is used for LP detection, LP character

Fig. 2 Annotation and detection considerations. (a) Taiwan scooter license plate annotation. (b) Any street objects that are visually similar to a LP can be detected, and our character detection in the next step can rule these out. (c) YOLO is good at distinguishing ‘I’ vs ‘1’ in text OCR.

Fig. 3 Visual results of scooter ALPR in (a) day time, (b) strong reflectance/over-exposure, (c) night time, (d) dark lighting condition, (e) green LP, (f) yellow LP, (g) red LP for heavy motorcycles. (h) Our method can be used to detect car LPs as well.

detection/recognition, and fiducial mark detection for LP rectification. Our empirical experience suggests that as long as a reasonable amount of training samples are provided(~1300 LPs and ~10K characters in our case, see 4), YOLO multi-class, multi-scale prediction can perform well enough for LP detection, character localization and OCR. Our method extends naturally to LPs in multi-languages and various lighting conditions.

3. PROPOSED METHOD

The proposed multi-pass YOLOv3 scooter LP detection pipeline consists of three stages. Fig.1 summarizes the pipeline.

In stage 1, a LP detection is performed on the whole input image, where presumably only the LP class is detected, as the other classes (individual characters) can be too small to identify (Fig.1b). The LP image is cropped and resized for the next stage of detection.

In stage 2, the same YOLO network is perform at the resized LP image, to identify and localize three types of fiducial marks: (a) the two screws, (b) the dot '.' Character at the center, and (c) the 4 corners of the LP (Fig.1c). If the two screws are located, they will be used for LP horizontal rectification (Fig.1d). If the 4 corners are located, they will be used for LP vertical rectification (Fig.1e). The dot character are useful for locating LP center for rectification geometry calculation.

In stage 3, character detection and recognition is performed. If sufficient number of characters are found, the LP is considered recognized. The more details are described as follows.

3.1. Initial license plate detection

We adopt the original YOLOv3 model [32] without major modification, except for a few hyper-parameter adaptation for training. We use batch size 64, subdivisions 16, max batches 50,200, #conv filters = (#classes + 1 + 4) × 3 = 135. Training starts with ImageNet pre-trained model darknet53.conv.74 available from the original YOLO site. In testing runs, we lower the detection score threshold to ensure high recall of LPs, as false detections can be reliably filtered out in the subsequent stages in our pipeline.

We use the LabelImg [33] GUI to perform annotation, and save labeling in PASCAL VOC XML format, as in Fig.2a. Since YOLO is not initially designed for detecting LPs, such 'vilella' use of YOLO in LP detection will not out-perform other state-of-the-art approaches. We do not explicitly label objects that are similar to LPs (street scene texts or addresses as in Fig.2b) as a non-LP class. While this can reduce some false-alarms, it also increases miss-detections. We rely on the LP character recognition at the end to make decisions.

In a typical street scene, it is common that 5 to 10 scooters are detected simultaneously, as shown in Fig.3. All detected LP images are cropped and resized into 300 × 600 for subsequent rectification steps.

3.2. LP fiducial detection for rectification

In this stage, the YOLOv3 model is used to detect three types of fiducial marks: (a) the two screws (where the two types of screws are treated as a single class), (b) the dot '.' Character at the LP center, and (c) the 4 LP corners (where the 4 types of corners are treated as a

Table 1: The TaiwanScooterPlate dataset: # samples are shown in each object class.

class	#	class	#	class	#	class	#
LP	1364	6	585	G	163	Q	113
screw	2783	7	596	H	174	R	95
corner	4674	8	683	I	29	S	74
.	1651	9	573	J	175	T	104
0	525	A	191	K	232	U	98
1	551	B	181	L	145	V	63
2	609	C	244	M	437	W	99
3	556	D	204	N	173	X	81
4	74	E	186	O	0	Y	100
5	550	F	139	P	198	Z	100

single class). They are landmark objects to be detected by our YOLO model. Refer to the annotation in Fig.2a and a test result in Fig.1c for these landmarks.

In most test cases, the two screws can be reliably detected, and the line spanning the screws is used to calculate the horizontal rectification of the LP view. If any screw is missing, we simply perform horizontal rectification using default (precalculated) parameters. Similarly, in most cases the 4 corners can be detected and used for vertical rectification calculation.

Horizontal rectification is performed by calculating the tilting angle of the screw-spanning line using arccos, and then rotating the LP image pixels back using bi-linear interpolation. Vertical rectification is performed similarly, by estimating the angles β_1 and β_2 between the left two corners and right two corners using atan2, respectively, and then applying shear transform to rectify the LP image as in Fig.1d. In case if any corner detection is missing, as long as one of β_1 or β_2 can be calculated, we can use the detected dot '.' character at the LP center or the screw positions to estimate the rough shape of the LP, assuming the LP is rectangular. This way, our vertical rectification can be performed robustly against clutter scenes and low light conditions in most cases, as demonstrated in Fig.3.

3.2. LP fiducial detection for rectification

YOLO object detection is performed on the rectified license plate for character detection and recognition. A LP is considered recognized only if there are 4 or more characters or digits together with the dot `.' character being recognized. This rule is based on Taiwan government regularization of scooter license plates.

In our ALPR pipeline, both character recognition steps (before and after rectification) can produce a valid LP recognition result. We select the most reasonable LP characters as the outcome, by considering the character locations w.r.t. the LP geometry. Specifically, all characters are sorted horizontally, and we check the average width between characters to rule out possible erroneous character detections.

4. EXPERIMENTAL RESULTS

4.1. Data preparation and model training

We provide a new TaiwanScooterLP dataset by combining the AOLP dataset [11] with our own scooter image collection and annotation. Scooter photos are taken in-the-wild from street scenes in Taiwan, which are organized into the following categories (with # images in each category): Street (708), Side-way (252), Road (59), Heavy-Motorcycle (86), Traffic-Surveillance (757), Parking-Entrance (681), Road-side (611), where the last three categories are from the AOLP dataset [11]. Our dataset contains 3,154 labeled scooter images in the street scene. There are totally 29,898 objects (10,464 training and 19,434 test). In which, there are totally 4,198 labeled LPs (1,364 training and 2,834 test). There are 25,700 characters: 9,100 training (5,302 labeled digits, and 3,798 alphabets) and 16,600 test characters.

Our YOLOv3 model is trained to detect the following 40 object classes: the LP, screw, corner, the dot '.' character, the digits (0 to 9), the upper-case characters (A to Z). Table 1 lists the number of samples (for both training and test) in each class. We use a total of 19,572 samples (including all object classes) to train our YOLOv3 detector.

The system is implemented in C and integrated with python and OpenCV. Evaluation is performed on a workstation with Intel i7 CPU, 24GB RAM and nVidia GTX 1080 GPU. The whole pipeline runs about ~10 FPS excluding I/O time.

4.1. Evaluation

Table 2 shows Precision-Recall evaluation results. We only perform evaluation for LPs and characters. Evaluation of the screw and corner detections is ignored, as they are only intermediate results used for LP

Table 2: LP detection and recognition evaluation results.

LP detection	# samples	TP	FP	FN	Precision	Recall	F1
Side-way	499	487	0	12	100	97.6	98.78
Road	221	183	0	38	100	82.81	90.58
Heavy-Motor	86	86	0	0	100	100	100
Traffic-Surveillance	792	768	2	24	99.74	96.97	98.34
Parking Entrance	667	663	0	4	100	99.4	99.7
Road-side	569	555	0	14	100	97.54	98.75
Total	2834	2742	2	92	99.93	96.75	98.31

Characters Recognition	# samples	TP	FP	FN	Precision	Recall	F1
Side-way	3080	3080	3	0	99.9	100	99.95
Road	1132	1123	10	9	99.12	99.2	99.16
Heavy-Motor	513	513	1	0	99.81	100	99.9
Traffic-Surveillance	4567	4565	12	2	99.74	99.96	99.85
Parking Entrance	3978	3978	2	0	99.95	100	99.97
Road-side	3330	3330	2	0	99.94	100	99.97
Total	16600	16589	30	11	99.82	99.93	99.88

rectification. The visual results in Fig.3 demonstrates the efficacy of our method in scooter ALPR running in the wild. The superior detection capability of YOLO enables ALPR in large variety of lighting conditions, including over-exposure and low lights, provided that these scenarios are included in the training.

5. CONCLUSIONS

We presented a multi-pass YOLOv3 ALPR system that can effectively detect and recognize scooter license plates in the wild. Our multi-pass YOLOv3 design is unique that it cannot only detect LPs but also detect fiducial marks such as LP screws and corners for view rectification. The same YOLO model is also used for LP character recognition. Our model is light-weight and runs fast. Results show that it can effectively detect and recognize LPs in various lighting conditions.

In the future, the proposed method will include LP tracking integration and handheld edge device deployment for further use in traffic administration and law enforcement. Furthermore, the YOLO v4 [34] will be used to train our TaiwanScooterPlate dataset to detect Scooter Plate with more accuracy.

ACKNOWLEDGEMENT

This work is supported by the Ministry of Science and Technology, Taiwan, under Grants MOST 108-2221-E-845 -003 -MY3.

REFERENCES

[1] L. Xie, T. Ahmad, L. Jin, Y. Liu and S. Zhang, “A New CNN-Based Method for Multi-Directional Car License Plate Detection,” in *IEEE Transactions on Intelligent Transportation Systems*, Vol. 19, No. 2, pp. 507-517, Feb. 2018.

[2] D. Tang, H. Kong, X. Meng, RZ. Liu, T. Lu, “SEE-LPR: A Semantic Segmentation Based End-to-End System for Unconstrained License Plate Detection and Recognition,” In: Ro Y. et al. (eds) *MultiMedia Modeling*. MMM 2020. Lecture Notes in Computer Science, Vol 11961. Springer, Cham.

[3] S. Abdullah, M. Mahedi Hasan and S. Muhammad Saiful Islam, “YOLO-Based Three-Stage Network for Bangla License Plate Recognition in Dhaka Metropolitan City,” in *International Conference on Bangla Speech and Language Processing* (ICBSLP), Sylhet, 2018, pp. 1-6.

[4] R. Al-qudah and C. Y. Suen. “Enhancing YOLO deep networks for the detection of license plates in complex scenes,” in *Proceedings of the Second International Conference on Data Science, E-Learning and Information Systems*, No. 21, pp.1-6, December 2019.

[5] S. M. Awan, S. Khattak, G. Z. Khan and Z. Mahmood, “A Robust Method to Locate License Plates under Diverse Conditions,” *2019 International Conference on Applied and Engineering Mathematics* (ICAEM), Taxila, Pakistan, 2019, pp. 92-98.

[6] S. Chen, C. Yang, J. Ma, F. Chen and X. Yin, “Simultaneous End-to-End Vehicle and License Plate Detection with Multi-Branch Attention Neural Network,” in *IEEE Transactions on Intelligent Transportation Systems*, pp. 1-10, 2019.

[7] S. Du, M. Ibrahim, M. Shehata and W. Badawy, “Automatic License Plate Recognition (ALPR): A State-of-the-Art Review,” in *IEEE Transactions on Circuits and*

Systems for Video Technology, Vol. 23, No. 2, pp. 311-325, Feb. 2013

[8] G. Resende Gonçalves, M. Alves Diniz, R. Laroca, D. Menotti and W. Robson Schwartz, "Real-Time Automatic License Plate Recognition through Deep Multi-Task Networks," in the 31st *SIBGRAPI Conference on Graphics, Patterns and Images* (SIBGRAPI), Parana, 2018, pp. 110-117.

[9] K. He, G. Gkioxari, P. Dollár and R. Girshick, "Mask R-CNN," in *IEEE International Conference on Computer Vision* (ICCV), Venice, 2017, pp. 2980-2988.

[10] G. Hsu, A. Ambikapathi, S. Chung and C. Su, "Robust license plate detection in the wild," in the 14th *IEEE International Conference on Advanced Video and Signal Based Surveillance* (AVSS), Lecce, 2017, pp. 1-6.

[11] G. Hsu, J. Chen and Y. Chung, "Application-Oriented License Plate Recognition," in *IEEE Transactions on Vehicular Technology*, Vol. 62, No. 2, pp. 552-561, Feb. 2013.

[12] G. J. Hsu and C. Chiu, "A comparison study on real-time tracking motorcycle license plates," in *IEEE 12th Image, Video, and Multidimensional Signal Processing Workshop* (IVMSP), Bordeaux, 2016, pp. 1-5.

[13] J. Huang et al., "Speed/Accuracy Trade-Offs for Modern Convolutional Object Detectors," in *IEEE Conference on Computer Vision and Pattern Recognition* (CVPR), Honolulu, HI, 2017, pp. 3296-3297.

[14] F. Jun and D. Shuguang, "A vehicle license plate location and correction method based the characteristics of license plate," in Proceedings of the 10th *World Congress on Intelligent Control and Automation*, Beijing, 2012, pp. 42-46.

[15] R. Laroca et al., "A Robust Real-Time Automatic License Plate Recognition Based on the YOLO Detector," in *International Joint Conference on Neural Networks* (IJCNN), Rio de Janeiro, 2018, pp. 1-10.

[16] R. Laroca, L. A. Zanlorensi, G. R. Gonc¸alves, E. Todt, W. R. Schwartz, and D. Menotti, "An efficient and layout-independent automatic license plate recognition system based on the YOLO detector," arXiv preprint arXiv:1909.01754, 2019.

[17] R. P. Loce, E. A. Bernal,W.Wu, and R. Bala, "Computer vision in roadway transportation systems: a survey," *Journal of Electronic Imaging*, Vol. 22, No. 4, 041121, 2013.

[18] VinhDu Mai, Duoqian Miao, Ruizhi Wang and Hongyun Zhang, "An improved method for Vietnam License Plate location," in *International Conference on Multimedia Technology*, Hangzhou, 2011, pp. 2942-2946.

[19] W. Puarungroj and N. Boonsirisumpun, "Thai license plate recognition based on deep learning," *Procedia Computer Science*, Vol. 135, pp. 214-221, 2018

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

[21] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *IEEE Conference on Computer Vision and Pattern Recognition* (CVPR), 2017, pp. 6517-6525.

[22] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," arXiv:1804.02767, 2018

[23] Z. Selmi, M. Ben Halima and A. M. Alimi, "Deep Learning System for Automatic License Plate Detection and Recognition," in 14th *IAPR International Conference on Document Analysis and Recognition* (ICDAR), Kyoto, 2017, pp. 1132-1138.

[24] Z. Selmi, M. B. Halima, and U. P. M. A. Alimi, "DELP-DAR system for license plate detection and recognition," *Pattern Recognition Letters*, Vol. 129, pp. 213–223, 2020

[25] S. M. Silva and C. R. Jung, "License plate detection and recognition in unconstrained scenarios," in *European Conference on Computer Vision* (ECCV), pp. 593–609, 2018.

[26] M. Sovani, D. Vo, S. Challa and M. Palaniswami, "A Feedback based method for License plate image binarization," in 9th *International Conference on Signal Processing and Communication Systems* (ICSPCS), Cairns, QLD, 2015, pp. 1-5.

[27] U. Upadhyay, F. Mehfuz, A. Mediratta and A. Aijaz, "Analysis and Architecture for the deployment of Dynamic License Plate Recognition Using YOLO Darknet," in *International Conference on Power Electronics, Control and Automation* (ICPECA), New Delhi, India, 2019, pp. 1-6.

[28] Y. Wang, Y.-T. Chen, and M.-N. Wang, "A new vehicle license plate correction method based on sobel operator and priori knowledge," in 6th *International Conference on Machinery, Materials, Environment, Biotechnology and Computer (*MMEBC), 2016.

[29] Z. Xu,W. Yang, A. Meng, N. Lu, and H. Huang, "Towards end-to-end license plate detection and recognition: A large dataset and baseline," in *European Conference on Computer Vision* (ECCV), pp. 255–271, 2018.

[30] S. Yang, C. C. Ho, J. Chen and C. Chang, "Practical Homography-based perspective correction method for License Plate Recognition," in International Conference on Information Security and Intelligent Control, Yunlin, 2012, pp. 198-201.

[31] AVLab-AOLP, https://sites.google.com/site/avlabaolp/

[32] YOLOv3, https://pjreddie.com/darknet/yolo/

[33] labelImg, https://github.com/tzutalin/labelImg

[34] Yolo-v4 and Yolo-v3/v2 for Windows and Linux, https://github.com/AlexeyAB/darknet