Journal of Robotics, Networking and Artificial Life, Vol. 3, No. 4 (March 2017), 274-278

Android-Based Patrol Robot Featuring Automatic Vehicle Patrolling and Automatic Plate Recognition

Chian C. Ho*, Shih-Jui Yang, Jian-Yuan Chen, Chang-Yun Chiang, and Hsin-Fu Chen
Department of Electrical Engineering, National Yunlin University of Science and Technology
Douliou, Yunlin 64002, Taiwan
*E-mail: futureho@yuntech.edu.tw**
www.yuntech.edu.tw

Abstract

This work develops an Android-based patrol robot featuring Automatic Vehicle Patrolling (AVP) and Automatic Plate Recognition (APR). The AVP feature integrates 3 novel methods, wheel-wheelcover-based AdaBoost wheel detection, contour-wheel-oriented vehicle approaching, and Ad-Hoc-based remote motion control. The APR feature integrates 4 novel methods, Wiener-deconvolution vertical edge enhancement, AdaBoost plus vertical-edge plate detection, vertical-edge horizontal-projection histogram-segmentation stain removal, and customized optical character recognition. Implementation results show the vehicle detection rate and plate recognition rate of the Android-based patrol robot are over 96% and over 94%, respectively, under various scene conditions. On the other hand, the average execution time of AVP and APR of the Android-based patrol robot takes at most 8 second per round and at most 0.8 second per frame, respectively.

Keywords: Android, robot, automatic vehicle patrolling, automatic plate recognition.

1. Introduction

With the ever-increasing demand of anti-terrorism and public security worldwide, the global law enforcement has severely been fighting against stolen vehicles or vehicles hung with stolen vehicle plates so far. Because these vehicles, both automobiles and motorcycles, are most likely used for terrorist activities or criminal vehicles. In addition, these vehicles are always parking in or running from unimaginable corners. It is necessary for the global law enforcement to carry out the vehicle plate investigation anywhere, anytime, even under exhausted manpower condition. Therefore, the automatic patrol robot equipped with Automatic Plate Recognition (APR) functionality can satisfy the growing demand and gain the expanding attention.

On the other hand, global digital video surveillance manufacturers are unexceptionally interested in the mobile robotic APR technology so as to evolve the existing fixed APR systems to portable embedded ones

Fig. 1. Mechanism of Android-based patrol robot featuring AVP and APR.

Copyright: the authors

or mobile robotic ones. Thus this work develops the Android-based patrol robot offering accurate APR and reliable Automatic Vehicle Patrolling (AVP) motion control for this emerging demand. It is composed of an Android-based smartphone platform and an Ad-Hoc-based mobile robot as shown in Fig. 1, and the Android-based smartphone platform as the host is steadily mounted onto the Ad-Hoc-based mobile robot.

In this work, the motion control of the Ad-Hoc-based mobile robot is actually steered by the Android-based smartphone platform through peer-to-peer Ad Hoc networking, while the AVP feature is routinely began by the Ad-Hoc-based mobile robot. As soon as the Android-based smartphone platform detects the vehicle contour, vehicle wheel, or vehicle plate, the Ad-Hoc-based mobile robot will be remotely controlled to approach the vehicle and further approach the plate. As for the APR feature, it is still performed by the Android-based smartphone platform, especially by the built-in auto-focus camera and low-power embedded processor, for long-endurance and cost-effective requirements. Besides, it is also a critical consideration that plenty of open-source computer vision libraries, like OpenCV Tesseract OCR, etc., are increasingly ported and supported on Android operating system [1].

This work can meet both specifications of portable embedded and mobile robotic APR systems, and it can be further transformed into any ubiquitous and diverse devices. This work not only can be applied to stolen vehicle plate tracking and roadside inspection, but also can be extensively applied to parking lot patrolling, container logistic investigation, or automotive manufacturing management.

Fig. 2 illustrates the overall algorithm flowchart of the developed Android-based patrol robot featuring AVP and APR. The flowchart in Fig. 2 is made up of 2 main stages. 1) Automatic Vehicle Patrolling (AVP) and 2) Automatic Plate Recognition (APR). Referring to Fig. 2, the organization of this paper is as follows. Sections 2 and 3 describe AVP feature and APR feature, respectively, in detail. Here, Section 2 is divided into 3 subsections: wheel-wheelcover-based AdaBoost wheel detection, contour-wheel-oriented vehicle approaching, and Ad-Hoc-based remote motion control. Section 3 is divided into 4 subsections: Wiener-deconvolution vertical edge enhancement, AdaBoost plus vertical-edge plate detection, vertical-edge horizontal-projection histogram-segmentation stain removal, and customized

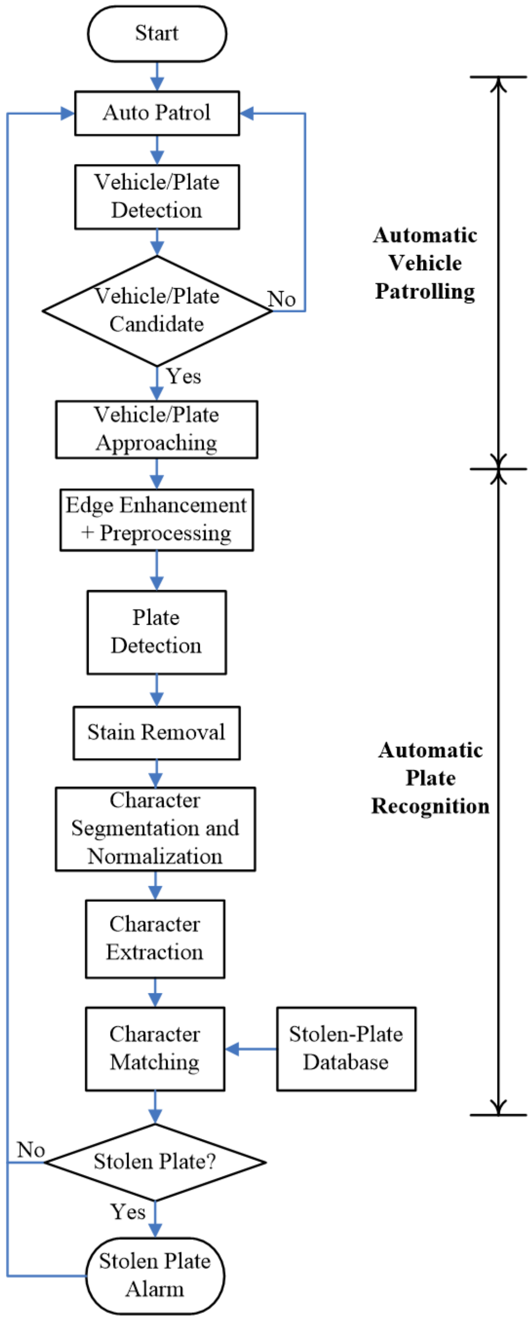

Fig. 2. Algorithm flowchart of Android-based patrol robot featuring AVP and APR.

Optical Character Recognition (OCR). Implementation results about the vehicle detection rate and the plate recognition rate improved by these proposed methods in Sections 2 and 3 are also exhibited, and so is the execution time of the overall implementation. Finally, Section 4 draws conclusions and future work.

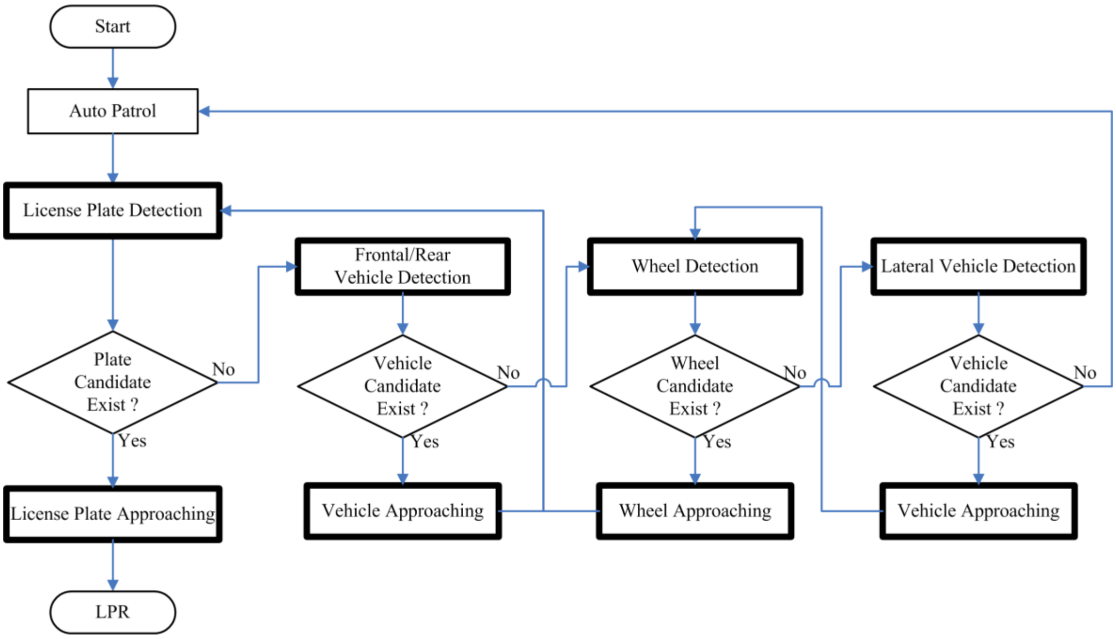

Fig. 3. Algorithm flowchart of developed AVP feature.

2. Automatic Vehicle Patrolling Feature

As shown in upper half part of Fig. 2, in the stage of AVP, autonomous patrol is initially launched by the mobile robot according to the predefined patrolling point and path planning. Next, the Android-based smartphone platform keeps working on the vehicle/plate detection to search the vehicle/plate candidates. Once one or more vehicle/plate candidates are found, the Android-based smartphone platform will evaluate the relative distance and the visual pose angle of the nearest vehicle/plate candidate, and will drive the mobile robot through Ad Hoc networking to approach the nearest vehicle/plate until the plate is close enough, that is, the character on the plate is clear enough to be recognized.

Because reliable AVP is the fundamental step toward the developed Android-based patrol robot, especially when vehicle/plate patrolling, searching, and approaching are proceeding under various conditions of illumination, scene, perspective, etc. This work proposes 3 novel methods for vehicle/plate detection and approaching, and implements peer-to-peer Ad Hoc networking mode for remote motion control [2]. The proposed 3 novel methods are summarized as follows.

2.1. *Wheel-wheelcover-based AdaBoost wheel detection*

This work develops an accurate and reliable AVP to approach the vehicle and further approach the plate closely enough, as shown in Fig. 3, whether the Android-based patrol robot is located on the frontal/rear side of the vehicle or lateral side of the vehicle. When the robot is on the frontal/rear side of the vehicle, the plate detection or frontal/rear vehicle detection is easy and straightforward. However, when the robot is on the lateral side of the vehicle, no plate can be detected, and lateral vehicle detection depends upon nothing but ambiguous features of wheel or lateral vehicle contour. Thus, this work proposes Wheel-wheelcover-based AdaBoost wheel detection based on wheel feature and unique wheelcover feature, and verifies the proposed wheel detection achieves better detection rate than conventional wheel detection methods, like Hough Transform wheel detection [3]-[5], Histogram of Oriented Gradients (HOG) wheel detection [6], [7], and AdaBoost wheel detection [8], [9].

2.2. *Contour-wheel-oriented vehicle approaching*

Once the Android-based patrol robot can smoothly detect vehicle plate or frontal/rear vehicle contour, the robot simply moves ahead to approach the vehicle/plate. But, if the Android-based patrol robot detects the vehicle wheel/wheelcover or lateral vehicle contour, it must move ahead and turn around to face the frontal/rear vehicle side, and then must go through the aforementioned steps of the frontal/rear vehicle side again for approaching the vehicle and further approaching the plate, illustrated as shown Fig. 3.

2.3. *Ad-Hoc-based remote motion control*

At first, the WiFi driver of the Android-based smartphone platform is elaborately switched from Access Point mode to Ad Hoc mode. Then, the WiFi mode of the mobile robot is also switched from Access Point mode to Ad Hoc mode, and makes the mobile robot log into the Ad Hoc network whose Service Set IDentifier (SSID) is broadcasted by the Android-based smartphone platform. Finally, after the mobile robot registers an IP address to the Ad Hoc networking gateway on the Android-based smartphone platform successfully, the Android-based smartphone platform can request any motion control commands to steer the Ad-Hoc-based mobile robot remotely.

3. Automatic Plate Recognition Feature

After approaching some nearest vehicle and plate quite closely, APR feature is engaged in detecting and recognition the vehicle plate, as shown in lower half part of Fig. 2. this work proposes 4 novel methods, Wiener-deconvolution vertical edge enhancement, AdaBoost plus vertical-edge plate detection, vertical-edge horizontal-projection histogram-segmentation stain removal, and customized OCR, for better plate detection rate and character recognition rate. These 4 novel methods are clarified in order as below.

3.1. *Wiener-deconvolution vertical edge enhancement*

In general, most vehicle plate detection systems make use of the vertical edge density feature to discriminate the vehicle plate candidate region from the background of the scene image, because most characters on the vehicle plate have higher vertical edge density than the background. If some vertical edge enhancement preprocessing method is put on the original scene image in advance, vehicle plate detection will work better. Because Wiener convolution is originally used for image blurring, this work proposes to deconvolute the scene image with a horizontal-direction Wiener filter for vertical edge enhancement. The experimental result shows the vertical edge density and intensity of the vehicle plate are strengthened by a horizontal-direction Wiener deconvolution method, and those of the background, like radiator grilles, are simultaneously weakened.

3.2. *AdaBoost plus vertical-edge plate detection*

Although AdaBoost cascaded classifier with proper and adequate training of vehicle plate patterns is almost feasible for exact plate detection, this work proposes two auxiliary methods to make plate-based AdaBoost plate detection more accurate and more reliable under various scene conditions. One method is character-based AdaBoost plate verification to filter out the false positive outcomes of plate-based AdaBoost plate detection by polling, and the other method is vertical-edge-based alternative plate detection which is activated when no plate candidate can be found by plate-based AdaBoost plate detection and character-based AdaBoost plate verification.

3.3. *Vertical-edge horizontal-projection histogram-segmentation stain removal*

The localized vehicle plate candidate is inevitably accompanying with some border stains, so the subsequent OCR procedure tends to misunderstand border stains as characters and is prone to make recognition mistakes. It is easier to remove interior stains on the vehicle plate because interior stains are similar to salt, and pepper noise can be easily eliminated by mathematical morphology. But it is quite harder to remove trapezoid border stains residing at the right side and left side, and interlinking border stains at the upside and downside. In this work, vertical-edge horizontal-projection histogram-segmentation stain removal based on fill-in and connected component labeling is proposed to effectively remove trapezoid border stains at the right side and left side, and interlinking border stains at the upside and downside.

3.4. *Customized optical character recognition*

Because Tesseract OCR open-source library is originally used for OCR of general documents, not for OCR of vehicle plates, this work elaborately retrains the font characteristics of each character on real Taiwanese vehicle plate into the character dictionary of Tesseract OCR [10]. Then, through Java Native Interface (JNI) technique, this work ports vehicle-plate-customized

Fig. 4. Field trial about OCR results of the developed Android-based patrol robot featuring AVP and APR.

Tesseract OCR onto the Android-based smartphone platform as a plug-in module of Android operating system. In addition, this work also proposes to adjust hybrid-pitch character segmentation of Tesseract OCR for the font characteristics of Taiwanese vehicle plate.

Fig. 4 demonstrates the field trial about OCR results of the developed Android-based patrol robot featuring AVP and APR. In Fig. 4, the upper row demonstrates AVP and APR results of automobile cases, and the lower row demonstrates those of motorcycle cases. The field trial result with the document recognition rate of over 99% and the plate recognition rate of over 98% verifies that Tesseract OCR can perform well on APR in Taiwan or elsewhere by customizing its character dictionary elaborately. Besides, Fig. 4 also demonstrates the vehicle detection rate and plate recognition rate of the Android-based patrol robot are over 96% and over 94%, respectively, under various scene conditions. On the other hand, the average execution time of AVP and APR of the Android-based patrol robot takes at most 8 second per round and at most 0.8 second per frame, respectively. Here, the clock frequency of the processor on Android-based smartphone platform is simply 1 GHz.

4. Conclusions and Future Work

In this work, the Android-based patrol robot featuring mobile AVP and real-time APR already can work well. In the near future, more smart AVP features and efficient APR features will continuously be developed and integrated into the Android-based patrol robot.

References

1. Google, Android Developers, [Online]. Available: http://developer.android.com/index.html
2. WowWee, API Specification for Rovio Version 1.ˇ3, [Online]. Available: http://breckon.eu/toby/teaching/dip/rovio/Rovio_API_Specifications_v1.3.pdf
3. F. Tan, L. Li, B. Cai and D. Zhang, Shape Template Based Side-View Car Detection Algorithm, *Proceedings of International Workshop on Intelligent Systems and Applications (ISA)*, pp. 1-4, 2011.
4. V. Vinoharan, A. Ramanan and S. R. Kodituwakku, A wheel-based side-view car detection using snake algorithm, *Proceedings of IEEE International Conference on Information and Automation for Sustainability*, pp. 185-189, 2012.
5. G. Xu, J. Su, H. Pan and D. Zhang, A Novel Method for Wheel Rim Recognition, *Proceedings of International Symposium on Electronic Commerce and Security*, pp. 1051-1054, 2008.
6. R. Brehar, S. Nedevschi and L. Dăian, Pillars detection for side viewed vehicles, *Proceedings of IEEE International Conference on Intelligent Computer Communication and Processing*, pp. 247-250, 2010
7. L. Liu, F. Zhou and Y. He, Vision-based fault inspection of small mechanical components for train safety, *IET Intelligent Transport Systems*, vol. 10, no. 2, pp. 130-139, Feb. 2016.
8. D. Dooley, B. McGinley, C. Hughes, L. Kilmartin, E. Jones and M. Glavin, A Blind-Zone Detection Method Using a Rear-Mounted Fisheye Camera With Combination of Vehicle Detection Methods, *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 1, pp. 264-278, Jan. 2016.
9. L. Liu, F. Zhou and Y. He, Automated Visual Inspection System for Bogie Block Key Under Complex Freight Train Environment, *IEEE Transactions on Instrumentation and Measurement*, vol. 65, no. 1, pp. 2-14, Jan. 2016.
10. Tesseract OCR, Core Developers, [Online]. Available: http://code.google.com/p/tesseract-ocr/