

Creating a Surveillance System to Detect License Plate of a Vehicle

Selman Hanci

*Faculty of Science, Department of Computer Science
Dokuz Eylul University
Izmir, Turkey
[https://orcid.org/
0000-0003-3145-7508](https://orcid.org/0000-0003-3145-7508)
selman_hanci@hotmail.com*

Fidan Nuriyeva

*Faculty of Science, Department of Computer,
Dokuz Eylul University
Izmir, Turkey
Institute of Control Systems of ANAS,
Baku, Azerbaijan
[https://orcid.org/
0000-0001-5431-8506](https://orcid.org/0000-0001-5431-8506)
fidan.nuriyeva@deu.edu.tr*

Abstract—In order to prevent accidents caused by traffic rule violations, Plate Recognizer tool is developed. Plate Recognizer can monitor the incoming video of traffic and detect the traffic rule violations with using image processing and advanced libraries. With this study, it is hoped to get more direct and accurate data of traffic characteristics and prevent the future problems.

Keywords—Traffic rule violation, plate, image processing

I. INTRODUCTION

In the modern world, cars are involved in every aspect of life. Without them, transportation is unthinkable. However, today, in parallel with the increase in the number of cars, there is an increase in traffic rule violations.

It is certain that a system is needed to examine the causes of these accidents and to prevent similar accidents in the future. The system to be produced should examine the camera images, define objects through them, process pictures and distinguish shapes, detect vehicles and detect their license plates, and report on these plates when necessary [1].

After considering the requirements and researching similar projects, a software written in c# programming language, using AForge, opencv libraries and Tesseract engine was developed.

II. REQUIREMENT

A system that uses images from cameras to detect traffic rule violations and can automatically detect problems and help us create a much more seamless transportation system needed to be created [2].

With the requirement that mentioned above, Plate Recognizer application is created. Plate Recognizer focuses on detecting car plates from a picture or from a live footage. Main screen of Plate Recognizer offers user a chance to upload a picture and detect the car license plate and chance to capture a picture from live footage when clicked and then fetch plate text from picture.

III. TECHNOLOGIES

In order to develop a surveillance system, C# programming language with Visual Studio environment tool is used [3].

Used libraries;

- AForge
- Tesseract Engine
- OpenCV

A. AForge

AForge.NET is a C# framework designed for developers and researchers in the fields of Computer Vision and Artificial Intelligence - image processing, computer vision, neural networks, genetic algorithms, machine learning, etc.

AForge.Video.DirectShow namespace contains classes, which allow to access video sources using DirectShow interface [4].

B. Tesseract Engine

Tesseract is an optical character recognition (OCR) engine for various operating systems. It is free software, released under the Apache License

Tesseract was considered one of the most accurate open-source OCR engines then available [5, 6].

C. OpenCV

OpenCV (Open-Source Computer Vision Library) is a library of programming functions mainly aimed at real-time computer vision. The library is cross-platform and free for use under the open-source Apache 2 License. Starting with 2011, OpenCV features GPU acceleration for real-time operations [7, 8].

IV. PROJECT

Plate Recognizer is a Form Application built in ASP .NET with C# language. It involves 2 main forms and has a simple yet useful interface.

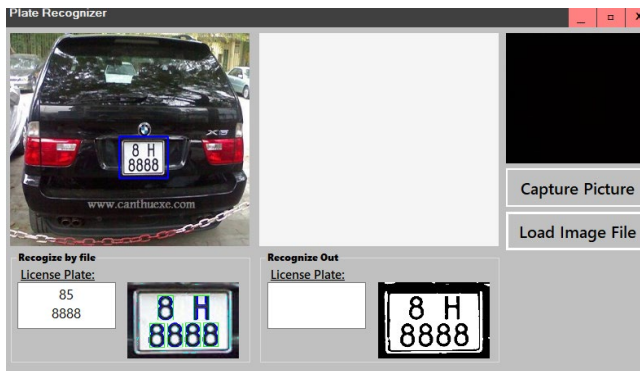


Fig. 1. Main screen of plate recognizer

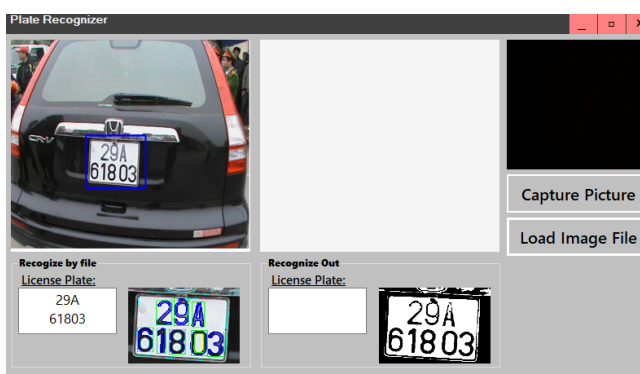


Fig. 2. Detecting plate from image file



Fig. 3. Gathering plate from live camera

MOBESE is a camera security system established in 81 provinces of Turkey to strengthen the hand of security forces fighting against crime and criminals.

Automatically MOBESE can only detect speeding violations but in future works we are planning to prove a

project to be possible to detect red light and crossing line violations with same cameras as well.

With the usage of newly developed libraries and increased detecting speed, Plate Recognizer can be embedded into existing system and the open roads would be much safer autonomously.

Working steps of the plate recognizer tool can be simplified as follows;

- **Step #1:** Detect and localize a license plate in an input image/frame
- **Step #2:** Extract the characters from the license plate
- **Step #3:** Apply some form of Optical Character Recognition (OCR) to recognize the extracted characters

V. CONCLUSION

With the proof of concept, results have shown that it is possible to create a surveillance system that can autonomously works on its own and detects traffic violations, create reports and send those reports via email or messages without any human interaction.

With positive results, there are next steps for Plate Recognizer. Currently because of the time limitation, licence plate detection is limited with square plates. Overwriting parts of Tesseract Engine will help overcome this issue.

Because the lack of equipments, only one camera is used. With multiple cameras concurrency problems will occurs but these can be handled as well.

REFERENCES

- [1] Witold Abramowicz, Dalia Kriksciuniene, Virgilijus Sakalauskas, "Business Information Systems", Springer Science & Business Media, 2012, pp 158
- [2] Ping Jack Soh, Wai Lok Woo, Hamzah Asyrani Sulaiman, "Advances in Machine Learning and Signal Processing: Proceedings of MALSIP", Springer Science & Business Media, 2016, pp 222.
- [3] Robert Bembenik, Lukasz Skonieczny, Henryk Rybinski, "Intelligent Tools for Building a Scientific Information Platform" Springer Science & Business Media, 2013, pp 198
- [4] Daisheng Luo, "Pattern Recognition and Image Processing", Woodhead Publishing, 1998
- [5] Wilhelm Burger, Mark J. Burge, "Principles of Digital Image Processing: Advanced Methods", Springer Science & Business Media, 2013
- [6] William K. Pratt, "Introduction to Digital Image Processing", CRC Press, 2013
- [7] Bernd Jähne, "Digital Image Processing", 2005, pp 25 – 35
- [8] Scott E Umbaugh, "Digital Image Processing and Analysis: Human and Computer Vision Applications with CVIptools", CRC Press, 2010