Advanced Car-Parking Security Platform Using Arduino Along with Automatic License Number Recognition

Bafreen Najeb Mohammed[1], Hawar Bahzad Ahmad[2]

[1] Zakho Technical Institute Information Technology (IT), Duhok Polytechnic University, Duhok, KRG – Iraq

[2] Department of Computer Science & I.T, Nawroz University, Duhok, Kurdistan Region – Iraq

ABSTRACT

The open-source electronic platform Arduino has been growing in recent years. Nowadays, the Arduino platform became one of the important parts in remote control and monitor of electrical devices. This paper aims to propose an application of digital image processing with Arduino, which can be useful in car-parking place to be more secure. Arduino will detect the car using ultra-sonic sensor and take a picture for its plat and send it to the system. Then, system applies digital image processing technique in the image. Different algorithms are incorporated in system such as segmentation, normalization, localization, orientation and optical character recognition (OCR) which analyze certain features in image and predicts car plate numbers. Afterwards the resulting data are compared against records in a stored database.

Keywords: Arduino, Optical Character Recognition, OCR, Image processing, Ultra-Sonic Sensor

1. Introduction

The digital image can be easily manipulated and edited Because of the powerfully built image processing and editing software available. In the current times, the license plate recognition (LNPR) and car identification systems are an important area of image processing and monitoring research [1,2,3]. The LNPR system has many applications, such as, border crossing control [4], identification of stolen vehicles [5], automated parking equipment [6], red-light camera [7], speed control, gas station monitoring and safety. For most of the applications mentioned, the LNPR's basic processing algorithm remain unchanged.

Arduino is an open-source integrated development environment (IDE) that receives electronic signals from solid tools such as sensors and a button using its own board as an input and sends it to the computer as an output in order to process it.

Arduino is responsible for detecting the car, capturing of images and sending image to the system. LNPR system must detect and recognize the numbers of the captured license plate [5].

The system's main equipment must be installed in car parking place. the system's purpose to detect vehicles that intended for security reasons, such as recognizing stolen or reported automobiles and so. The technique used in some LNPR systems is pattern matching [10]. Which is a fast, efficient and accurate for real time applications. However, pattern matching algorithm requires prior information regarding character shape, alignment and direction [9]. Thus, the algorithm must be modified according to the country/ providence, since each country uses different fonts and orientation for license plates [12].

The purpose of this article is to develop and implement an intelligent system for operational organizations where information and communication technology (ICT) is optimally used. Ensure that better decisions are made and more efficient methods of strategic planning and personnel management are adopted through the design and implementation of a database. Vehicles can be identified in a number of ways; however, license plate numbers are the most common and most suitable element for identifying vehicles. Vehicle identification by build and model is overhead. The special technical equipment used in model identification is not economical and cannot provide a high level of safety because more than one vehicle is built with the same model on the other hand,

recognition using license plates is more efficient, safe and practical. Many developed countries rely on LNPR for security, escape management, and theft prevention [13].

2. System Model

The proposed system is intended for the management and monitoring of vehicles in private and public organizations. The equipment required by the system are an Arduino uno with ultrasonic sensor which is required to measure the distance between the car and entrance and exit of parking spaces. When the car is within range of the sensors, Arduino will send the signal to the camera to capture the image (see figure 1). Images that are captured with these cameras and information (including driver image) are stored in the database. This enables gathering detailed information about traffic from different accesses at different times. Moreover, the system enables intelligent access control through allowing access to authorized vehicles only as shown in figure 1. Furthermore, the Fig. 1. The proposed system for opening our desired mechanized gate in a parking has proposed system poses other advantages such as number of available parking spaces, number of vehicles currently parked, details about authorized vehicles, and much more information on traffic statistics can be extracted from the database [14].
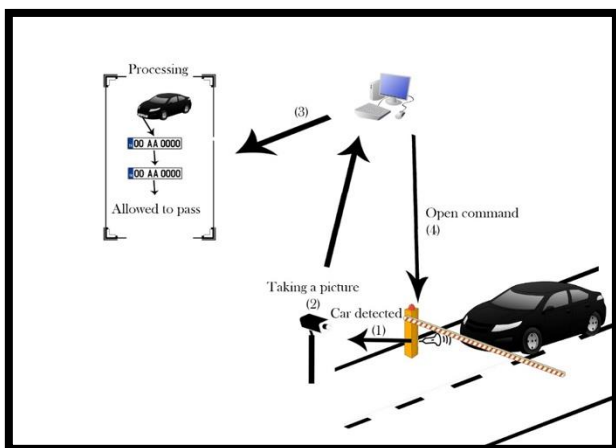

Fig. 1: Proposed System

The system has been developed with high quality software development and programming techniques. The software is designed in such a manner that it is easy, at lowest cost and time possible.

In the system proposed, an administration is responsible for defining and populating a "whitelist", which is a list of vehicles that are allowed access to the site, and a "blacklist", which is a list of access restricted vehicles.
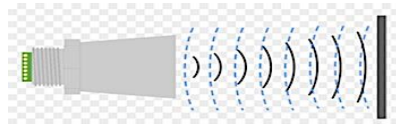
3. Algorithm

3.1 System hardware and software

This section provides the system introduction. Where an algorithm is implemented on each of the image processing modules. which are as follows:

- Ultra-sonic sensor Detecting the car: first part of the system is car detecting where the car is in the range of the Arduino sensors. Arduino then will send instruction to the camera to take a picture of the car and send it to the system to process it. as in figure 2 (a).

- Conversion. This algorithm is responsible for converting the image coloring mode into a simpler one. as in figure 2 (b).

- Plate localization algorithm: which is responsible for identifying a license plate on the image. See Figure 2 (c)

- Binarization: this algorithm is responsible for converting the gray scale image to binary image. Thereafter, compliment is required because the system dataset color is opposite of binary color. 2 (d).

- Orientation and calibration algorithm of the plate: This algorithm is responsible for readjusting the dimensions are required and reorienting the image, as shown in Figure 2 (e).

- Edge detection: edge detection is used to sharpen the edges and increase the difference between the numbers and the background. It can also omit visual noise. See figure 2 (f).

- Segmentation of symbols: this algorithm is responsible for finding individual symbols on the plate and segments related to the symbols for additional amplification, and additional noise is omitted.

- Optical Character Recognition: OCR is responsible for converting the printed text or image into digital text.

- Syntactic and geometric analysis: it locates symbols and stamps relative to specific countries.

- Averaging the recognized value over several fields / images to obtain a more reliable or confident result. Moreover, any image can contain a light flash, partially hidden or other temporary effect.

(a) Ultra-sonic sensor

(b) Taking picture by camera

(c) Conversion

(d) Plate localization algorithm:

(e) 4. Binarization

(f) Edge detection

(g) Segmentation of symbols

06CD2353

(h) Optical Character Recognition

Fig. 2: System Algorithm

3.2 Proposed Solution

In this section the process of number plate recognition is discussed, which consists of five main stages:

- Arduino sensor
- Preprocessing
- License plate localization
- Character segmentation
- Character recognition

3.2.1 Arduino sensor

As stated before, the system is faced with many challenges. This phase is concerned with enhancing the image input, making it more for the next phase. The Ultra-sonic sensor use sonar to measure the distance of the object from two centimeter to four meters. The distance used by the proposed system is 50cm to 70cm.

3.2.2 Preprocessing

In preprocessing, first a minimum filter is applied to the input images. This is done so that the dark values inside the image are enhanced by increasing the area of these dark values. The minimum filter enhances character and plate edges making them bold. Moreover, this step removes the diagonal stripe lights that appear in the Egyptian license plate (figure 1). Afterwards, the image saturation is increased to increase the difference between colors. Furthermore, the image coloring mode it turned into grey scale (taking the luminance component of NTSC). And the background is separated from the highlights by increasing image contrast (see figure 1) [8].

i.   **License plate localization**

In this phase, license plate location is identified in the image and it is extracted from the main image. A sub image is resulted from this phase that contains only the license plate. This is accomplished in two main steps.

-    Locating a Large Bounding Rectangle Over the License Plate.

- Determining the exact location of the license plate.

ii. Character segmentation

At this stage, the characters are separated from the plate. A set of monochrome images of the characters are output from this stage.

iii. Character recognition

this stage is concerned with identifying the recognizing the binary images of characters received from the previous phase. After this each character must have a label and an error factor. The error factor is used to identify false characters which might be received from the previous phase by error. If the error factor is greater than a known range, then characters are disregarded. With regards to classifying the characters, some features must be collected from the characters to achieve the classification. The feature that this system relies on is the chain code of the contour of the image. Which is used after dividing it into four tracks which in turn are divided into four sectors [10].
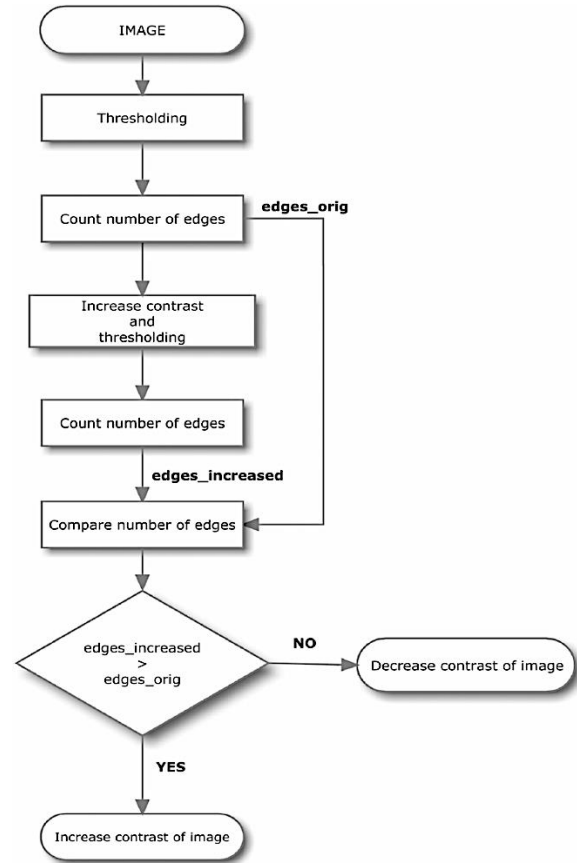

Fig. 3: Steps to Locate a License Number

Fig. 3. Final located plate; After adaptive threshold application, after small area and incorrect aspect ratio components are removed, which is done using flood fill and character separation using horizontal projection. Notice the gaps between characters create peaks in the projection while characters create plummets.

The chain code features of each character are acquired and fed to the Feed Forward Neural Network (FFNN). After this, the neuron class with the heights values will predict the character based on the character class represented by the neuron. Moreover, if the error value is more than a predefined threshold. The characters are considered false and discarded. Furthermore, plates are known to a fixed range of number of characters, thus if the total number of characters that appear on a plate does not be within range, the plate is rejected. Else the license plate number is determined.

4. Experimental result

Testing is the next phase for this system development where a number of image sample are tested in order to

assess the efficiency and performance of the proposed system, a comprehensive number of experiments for pattern recognition are performed on a large number of images. The result of the experiments signify that the system is more accurate and requires less time for training when compared to other recognition systems [11].

The system is designed in the MATLAB r2015a and Simulink Support Packages for Arduino were used to connect Arduino with MATLAB in order to use ultra-sonic sensor to detect the car and then car plate number recognition proceeded. The accuracy measurement is given by Table 1.

30 images were used to test the accuracy of the system, only 26 are extracted correctly. The reasons behind not extracted 4 images was either the image isn't clear because of the position of the car or the cleaning process of the unwanted objects in images deleted some important information as shown in Fig.13.

There are 23 images correctly recognized from total of 26 extracted images, as shown in Table 1.

the result of recognition of some sample of image are given by Fig. 3.

Fig. 3: Recognition results

5. Conclusion

In conclusion, this paper focus on automatic vehicle identification systems. The proposed system is used for automotive license plate number detection and recognition. The software has been implemented based on the aforementioned algorithms and functions in MATLAB. The resultant system has elevated performance regarding real time image processing. Moreover, the implemented system has been simulated and practically tested and has been able to detect vehicle license plates efficiently and reliably in different circumstances regarding lighting and weather conditions. The implemented system can be integrated or installed in any desired location such as restricted areas, roads or junctions for the purpose of

control, theft identification, entrance management or for any required application.

6. References

1. B.D. Acosta, Experiments in image segmentation for automatic US license plate recognition, M.Sc. thesis, Department of Computer Science, Faculty of the Virginia Polytechnic Institute and State University (2004).

2. C.N.E. Anagnostopoulos, I.E. Anagnostopoulos, I.D. Psoroulas, V. Loumos and E. Kayafas, License Plate Recognition From Still Images and Video Sequences: A Survey, IEEE Transactions on Intelligent Transportation Systems 9 (2008), no. 3, 377–391.

3. C.N.E. Anagnostopoulos, I.E. Anagnostopoulos, V. Loumos and E. Kayafas, A License Plate. Recognition Algorithm for Intelligent Transportation System Applications, IEEE Transactions on Intelligent Transportation Systems 7 (2006), no. 3, 377–392.

4. S..L. Chang; L..S. Chen; Y..C. Chung; S..W. Chen, Automatic License Plate Recognition, IEEE Transactions on Intelligent Transportation Systems 5 (2004), no. 1, 42–53.

5. S. Draghici, A neural network based artificial vision system for licence plate recognition, Inter. national Journal of Neural Systems 8 (1997), no. 1, 113–126.

6. [6] B. Enyedi, L. Konyha, C. Szombathy and K. Fazekas, Strategies for fast license plate number localization, Proceedings of the 46th International Symposium Electronics in Marine, Elmar 2004, Zadar, Croatia, June 16.18, 2004, IEEE Press (2004), 579–584 .

7. J..W. Hsieh, S..H. Yu and S..H. Yu, Morphology.based License Plate Detection from Complex Scenes, IEEE Proceedings of the 16th International Conference on Pattern Recognition, Qubec City, Canada, August 11.15, 2002, Vol. 3 (2002), 176.179.

8. O. Martinsky, Algorithmic And Mathematical Principles Of Automatic Number Plate Recognition Systems, B.Sc. thesis, Department of Intelligent Systems, Faculty of Information Technology, Brno University of Technology (2007).

9. V. Shapiro, D. Dimov, S. Bonchev, V. Velichkov and G. Gluhchev, Adaptive License Plate Image Extraction, Proceedings of the 5th international conference on Computer systems and technologies, Rousse, Bulgaria, June 17.18, 2004 (K. Boyanov), ACM New York, NY, USA (2004), 1.7.

10. H. Sarukhanyan, S. Alaverdyan, and G. Petrosyan, Automatic Number Plate Recognition System, Proceedings of the 7th International Conference on Computer Science and Information Technologies, Yerevan, Armenia, September 28.2 October, 2009, Electronic Copy of the CSIT 2009 Proceedings, 347.350.

11. C. A. Rahman, W. Badawy, and A. Radmanesh, "A Real Time Vehicle? s License Plate Recognition System," in null, 2003, p. 163.

12. T. Sirithinaphong and K. Chamnongthai, "The recognition of car license plate for automatic parking system," in Signal Processing and Its Applications, 1999. ISSPA'99. Proceedings of the Fifth International Symposium on, 1999, pp. 455.457.

13. S. Du, M. Ibrahim, M. Shehata, and W. Badawy, "Automatic license plate recognition (ALPR): A state.of.the.art review," Circuits and Systems for Video Technology, IEEE Transactions on, vol. 23, pp. 311.325, 2013.

14. M. V. Srinu and B. S. Shankar, "Real Time Car Parking System and Parking Fee Display Using Raspberry Pi," International Journal of Research, vol. 3, pp. 421.426, 2016.