REAL TIME ANPR FOR VEHICLE IDENTIFICATION USING NEURAL NETWORK

Subhash Tatale[1] and Akhil Khare[2]

[1]Student & [2]Assoc. Prof., Deptt. of Info. Tech., Bharti Vidyapeeth Deemed Uni., Pune, India.

ABSTRACT

This paper deals with problematic from field of artificial intelligence, machine vision and neural networks in construction of an automatic number plate recognition system (ANPR). This paper includes brief introduction of automatic number plate recognition, which ensure a process of number plate detection, processes of proper characters segmentation, normalization and recognition. Automatic Number Plate Recognition (ANPR) is a real time embedded system which automatically recognizes the license number of vehicles. In this paper, the task of recognizing number plate is considered. First the image of number plate is captured by camera. Number plate is segmented by using horizontal and vertical projection. After that feature extraction techniques are used to extract the characters from segmented data. Neural Network algorithms are used to recognize the characters which improve the color and brightness. ANPR project is very much useful in applications like, automated traffic surveillance and tracking system, automated high-way/parking toll collection systems, automation of petrol stations, travelling time monitoring.. In this paper, introduction of number plate segmentation, feature extraction, recognition of character based on Neural Network and syntax checking analysis of recognized characters is described.

KEYWORDS: *Artificial Intelligence, Neural Networks, Optical Character Recognition, ANPR*

I. INTRODUCTION

ANPR is a mass surveillance system that captures the image of vehicles and recognizes their license number. Some applications of an ANPR system are, automated traffic surveillance and tracking system, automated high-way/parking toll collection systems, automation of petrol stations, travelling time monitoring.

Such systems automate the process of recognizing the license number of vehicles, making it fast, robust, time- efficient and cost-effective.

1.1 ANPR systems as a practical application of artificial intelligence

Massive integration of information technologies into all aspects of modern life caused demand for processing vehicles as conceptual resources in information systems. Because a standalone information system without any data has no sense, there was also a need to transform information about vehicles between the reality and information systems. This can be achieved by a human agent, or by special intelligent equipment which is to be able to recognize vehicles by their number plates in a real environment and reflect it into conceptual resources. Because of this, various recognition techniques have been developed and number plate recognition systems are today used in various traffic and security applications, such as parking, access and border control, or tracking of stolen cars.

In parking, number plates are used to calculate duration of the parking. When a vehicle enters an input gate, number plate is automatically recognized and stored in database. When a vehicle later exits the parking area through an output gate, number plate is recognized again and paired with the first-one stored in the database. The difference in time is used to calculate the parking fee. Automatic number plate recognition systems can be used in access control. For example, this technology is used in many companies to grant access only to vehicles of authorized personnel.

In some countries, ANPR systems installed on country borders automatically detect and monitor border crossings. Each vehicle can be registered in a central database and compared to a black list of stolen vehicles. In traffic control, vehicles can be directed to different lanes for a better congestion control in busy urban communications during the rush hours.

1.2 Current systems

ANPR systems have been implemented in many countries like Australia, Korea and few others. Strict implementation of license plate standards in these countries has helped the early development of ANPR systems. These systems use standard features of the license plates such as: dimensions of plate, border for the plate, color and font of characters, etc. help to localize the number plate easily and identify the license number of the vehicle.

In India, number plate standards are rarely followed. Wide variations are found in terms of font types, script, size, placement and color of the number plates. In few cases, other unwanted decorations are present on the number plate. Also, unlike other countries, no special features are available on Indian number plates to ease their recognition process. Hence, currently only manual recording systems are used and ANPR has not been commercially implemented in India.

In this section, we have given brief introduction of Automatic Number Plate Recognition system which is based on Artificial Intelligence and Neural Network. Also explained the applications of ANPR and current trends of ANPR system.

II. NUMBER PLATE AREA DETECTION

The first step in a process of automatic number plate recognition is a detection of a number plate area. The algorithms that are able to detect a rectangular area of the number plate in an original image. Humans define a number plate in a natural language as a "small plastic or metal plate attached to a vehicle for official identification purposes", but machines do not understand this definition as well as they do not understand what "vehicle", "road", or whatever else is. Because of this, there is a need to find an alternative definition of a number plate based on descriptors that will be comprehensible for machines.

Let us define the number plate as a "rectangular area with increased occurrence of horizontal and vertical edges". The high density of horizontal and vertical edges on a small area is in many cases caused by contrast characters of a number plate, but not in every case. This process can sometimes detect a wrong area that does not correspond to a number plate. Because of this, we often detect several candidates for the plate by different algorithms.

In general, the captured snapshot can contain several number plate candidates. Because of this, the detection algorithm always clips several bands, and several plates from each band. There is a predefined value of maximum number of candidates, which are detected by analysis of projections. By default, this value is equals to nine.

There are several heuristics, which are used to determine the cost of selected candidates according to their properties. These heuristics have been chosen ad hoc during the practical experimentations. The recognition logic sorts candidates according to their cost from the most suitable to the least suitable. Then, the most suitable candidate is examined by a deeper heuristic analysis. The deeper analysis definitely accepts, or rejects the candidate. As there is a need to analyze individual characters, this type of analysis consumes big amount of processor time.

The basic concept of analysis can be illustrated by the following steps:

1. Detect available number plate inputs.
2. Sort them according to their cost which is based on heuristics.
3. Cut the first plate from the list with the best cost.
4. Segment the number plate.
5. Analyze it by a deeper analysis which is time consuming.
6. If the deeper analysis rejects the plate, return to the step 3.

In this section, we have given the introduction of detection of number plates once the image is captured by camera. The basic steps are discussed for analysis of segmentation of number plate

III. NUMBER PLATE SEGMENTATION

The next step after the detection of the number plate area is a segmentation of the plate. The number plate can be segmented based on horizontal or vertical projection. The segmentation is one of the most important processes in the automatic number plate recognition, because all further steps rely on it. If the segmentation fails, a character can be improperly divided into two pieces, or two characters can be improperly merged together.

We can use a horizontal projection of a number plate for the segmentation, or one of the more sophisticated methods, such as segmentation using the neural networks. If we assume only one-row plates, the segmentation is a process of finding horizontal boundaries between characters. The segment of plate contains besides the character also redundant space and other undesirable elements. We understand under the term "segment" the part of a number plate determined by a horizontal segmentation algorithm. Since the segment has been processed by an adaptive thresholding filter, it contains only black and white pixels. The neighboring pixels are grouped together into larger pieces, and one of them is a character. Our goal is to divide the segment into the several pieces, and keeps only one piece representing the regular character.

The second phase of the segmentation is an enhancement of segments. The segment of a plate contains besides the character also undesirable elements such as dots and stretches as well as redundant space on the sides of character. There is a need to eliminate these elements and extract only the character. The piece chosen by the heuristics is then converted to a monochrome bitmap image. Each such image corresponds to one horizontal segment. These images are considered as an output of the segmentation phase of the ANPR process.

In this section, we have given the introduction of segmentation of number plates once the number plate is detected. For the segmentation of number plate, we have used horizontal and vertical projection.

IV. FEATURE EXTRACTION

Before extracting feature descriptors from a bitmap representation of a character, it is necessary to normalize it into unified dimensions. We understand under the term "resampling" the process of changing dimensions of the character. As original dimensions of unnormalized characters are usually higher than the normalized ones, the characters are in most cases downsampled. When we downsample, we reduce information contained in the processed image.

There are several methods of resampling, such as the pixel-resize, bilinear interpolation or the weighted-average resampling. We cannot determine which method is the best in general, because the successfulness of particular method depends on many factors. For example, usage of the weighed-average downsampling in combination with a detection of character edges is not a good solution, because this type of downsampling does not preserve sharp edges. Because of this, the problematic of character resampling is closely associated with the problematic of feature extraction.

To recognize a character from a bitmap representation, there is a need to extract feature descriptors of such bitmap. As an extraction method significantly affects the quality of whole OCR process, it is very important to extract features, which will be invariant towards the various light conditions, used font type and deformations of characters caused by a skew of the image.

The description of normalized characters is based on its external characteristics because we deal only with properties such as character shape. Then, the vector of descriptors includes characteristics such as number of lines, bays, lakes, the amount of horizontal, vertical and diagonal or diagonal edges, and etc. The feature extraction is a process of transformation of data from a bitmap representation into a form of descriptors, which are more suitable for computers. If we associate similar instances of the same character into the classes, then the descriptors of characters from the same class should be geometrically closed to each other in the vector space. This is a basic assumption for successfulness of the pattern recognition process.

4.1 Feature extraction algorithm

At first, we have to embed the character bitmap $f(x, y)$ into a bigger bitmap with white padding to ensure a proper behavior of the feature extraction algorithm. Let the padding be one pixel wide. Then, dimensions of the embedding bitmap will be $w+2$ and $h+2$.

The embedding bitmap $f'(x, y)$ is then defined as:

$$f'(x,y) = \begin{cases} 1 & if\ x = 0 \vee y = 0 \vee x = w+1 \vee y = h+1 \\ f(x-1, y-1) & if\ \neg(x = 0 \vee y = 0 \vee x = w+1 \vee y = h+1) \end{cases}$$

where w and h are dimensions of character bitmap before embedding. Color of the padding is white (value of 1). The coordinates of pixels are shifted one pixel towards the original position.

The structure of vector of output descriptors is illustrated by the pattern below. The notation $h_j@r_i$ means "number occurrences of an edge represented by the matrix h_j in the region r_i".

$X= (h_0@r_0, h_1@r_{0,}... h_{n-1}@r_0, h_0@r_1, h_1@r_{1,}..., h_{n-1}@r_1, h_0@r_{p-1}, h_1@r_{p-1},..., h_{n-1}@ r_{p-1})$

We compute the position k of the $h_j@r_i$ in the vector **x** as $k = i.n + j$, where n is the number of different edge types (and also the number of corresponding matrices).

The following algorithm demonstrates the computation of the vector of descriptors **x:**

```
zerosize vector x
for each region r_i , where i∈0,⋯,ρ−1 do
begin
    for each pixel [x, y] in region r_i do
    begin
        for each matrix h_j, where j∈0,⋯,n−1 do
        begin
            if h_j = [ f'(x,y)      f'(x+1,y)   ] then
                     [ f'(x,y+1)    f'(x+1,y+1) ]
            begin
                let k=i.n +j
                let x_k= x_k+1
            end
        end
    end
end
```

In this section, feature extraction algorithm is explained.

V. NORMALIZATION OF CHARACTERS

The first step is a normalization of a brightness and contrast of processed image segments. Second step is the characters contained in the image segments must be then resized to uniform dimensions. Third step is, the feature extraction algorithm extracts appropriate descriptors from the normalized characters. The brightness and contrast characteristics of segmented characters are varying due to different light conditions during the capture. Because of this, it is necessary to normalize them. There are many different ways, but this section describes the three most used: histogram normalization, global and adaptive thresholding.

Through the histogram normalization, the intensities of character segments are re-distributed on the histogram to obtain the normalized statistics. The areas of lower contrast will gain a higher contrast without affecting the global characteristic of image.

Techniques of the global and adaptive thresholding are used to obtain monochrome representations of processed character segments. The monochrome (or black and white) representation of image is more appropriate for analysis, because it defines clear boundaries of contained characters.

5.1 Adaptive Scheduling

The number plate can be sometimes partially shadowed or non-uniformly illuminated. This is most

frequent reason why the global thresholding fail. The adaptive thresholding solves several disadvantages of the global thresholding, because it computes threshold value for each pixel separately using its local neighborhood.

5.2 Chow and Kaneko approach

There are two approaches to finding the threshold. The first is the Chow and Kaneko approach, and the second is a local thresholding. The both methods assumes that smaller rectangular region are more likely to have approximately uniform illumination, more suitable for thresholding. The image is divided into uniform rectangular areas with size of (m x n) pixels. The local histogram is computed for each such area and a local threshold is determined. The threshold of concrete point is then computed by interpolating the results of the sub images.
In this section, character normalization techniques are discussed. Adaptive Scheduling and Chow and Kaneko approach are the techniques which are used to normalization of characters.

VI. CHARACTER RECOGNITION AND SYNTAX CHECKING

The segmentation algorithm can sometimes detect redundant elements, which do not correspond to proper characters. The shape of these elements after normalization is often similar to the shape of characters. Because of this, these elements are not reliably separable by traditional OCR methods, although they vary in size as well as in contrast, brightness or hue. Since the feature extraction methods do not consider these properties, there is a need to use additional heuristic analyses to filter non-character elements. The analysis expects all elements to have similar properties. Elements with considerably different properties are treated as invalid and excluded from the recognition process.
The analysis consists of two phases. The first phase deals with statistics of brightness and contrast of segmented characters. Characters are then normalized and processed by the piece extraction algorithm. Since the piece extraction and normalization of brightness disturbs statistical properties of segmented characters, it is necessary to proceed the first phase of analysis before the application of the piece extraction algorithm.
In addition, the heights of detected segments are same for all characters. Because of this, there is a need to proceed the analysis of dimensions after application of the piece extraction algorithm. The piece extraction algorithm strips off white padding, which surrounds the character.
Respecting the constraints above, the sequence of steps can be assembled as follows:
1. Segment the plate (result is in figure 6.1.a).
2. Analyze the brightness and contrast of segments and exclude faulty ones.
3. Apply the piece extraction algorithm on segments (result is in figure 6.1.b).
4. Analyze the dimensions of segments and exclude faulty ones.

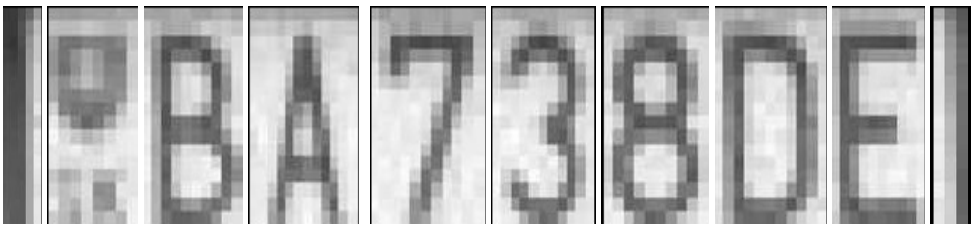

Figure 6.1 (a): Character segments before application of the piece extraction algorithm.

Figure 6.1(b): Character segments after application of the piece extraction algorithm.

In some situations when the recognition mechanism fails, there is a possibility to detect a failure by a syntactical analysis of the recognized plate. If we have country-specific rules for the plate, we can evaluate the validity of that plate towards these rules. Automatic syntax-based correction of plate numbers can increase recognition abilities of the whole ANPR system.

For example, if the recognition software is confused between characters "8"and "B", the final decision can be made according to the syntactical pattern. If the pattern allows only digits for that position, the character "8"will be used rather than the character "B". This is the most critical stage of the ANPR system. Direct template matching can be used to identify characters. However, this method yields a very low success rate for font variations which are commonly found in Indian number plates.

Artificial Neural Networks like BPNNs can be used to classify the characters. However, they do not provide hardware and time optimization. Therefore statistical feature extraction has been used. In this method, initially the character is divided into twelve equal parts and fourteen features are extracted from every part. The features used are binary edges (2X2) of fourteen types. The feature vector is thus formed is compared with feature vectors of all the stored templates and the maximum value of correlation is calculated to give the right character. Lastly syntax checking is done to ensure that any false characters are not recognized as a valid license number.

In this section, we have discussed how the characters are recognitioned by using Neural Network techniques. Also discussed the syntax analysis techniques once the character is recognitioned.

VII. RESULTS

ANPR solution has been tested on static snapshots of vehicles, which has been divided into several sets according to difficultness. Sets of blurry and skewed snapshots give worse recognition rates than a set of snapshots, which has been captured clearly. The objective of the tests was not to find a one hundred percent recognizable set of snapshots, but to test the invariance of the algorithms on random snapshots systematically classified to the sets according to their properties.

The table 7.1 shows recognition rates, which has been achieved while testing on various set of number plates. According to the results, this system gives good responses only to clear plates, because skewed plates and plates with difficult surrounding environment causes significant degradation of recognition abilities.

Table 7.1: Recognition rates of the ANPR system.

	Total Number of Plates	Total Number of Characters	Weighted score
Clear plates	62	425	88.76
Blurred plates	41	324	50.43
Skewed plates	34	264	54.26
Average plates	104	1137	75.34

VIII. CONCLUSIONS

The system works satisfactorily for wide variations in illumination conditions and different types of number plates commonly found in India. It is definitely a better alternative to the existing manual systems in India.

Currently there are certain restrictions on parameters like speed of the vehicle, script on the number plate, cleanliness of number plate, quality of captured image, skew in the image which can be aptly removed by enhancing the algorithms further.

REFERENCES

[1] Peter M. Roth, Martin K¨ostinger, Paul Wohlhart, and Horst Bischof, Josef A. Birchbauer (2010): Automatic Detection and Reading of Dangerous, 2010 Seventh IEEE International Conference on Advanced Video and Signal Based Survillance.

[2] Ping Dong, Jie-hui Yang, Jun-jun Dong (2006): The Application and Development Perspective of Number Plate Automatic Recognition Technique.

[3] W. K. I. L. Wanniarachchi, D. U. J. Sonnadara and M. K. Jayananda (2007): License Plate Identification Based on Image Processing Techniques, Second International Conference on Industrial and Information Systems.

[4] Ankush Roy Debarshi Patanjali Ghoshal(2011): Number Plate Recognition for Use in Different Countries Using an Improved Segmentation, IEEE.
[5] Luis Salgado, Jose' M. Mene'ndex, Enrique Renddn and Narciso Garcia (1999): Automatic Car Plate Detection and Recognition through Intelligent Vision Engineering, IEEE.
[6] Hwajeong Lee, Daehwan Kim, Daijin Kim, Sung Yang Bang (2003): Real-Time Automatic Vehicle Management System Using Vehicle Tracking and Car Plate Number Identification, IEEE.
[7] Ping Dong, Jie-hui Yang, Jun-jun Dong (2006): The Application and Development Perspective of Number Plate Automatic Recognition Technique, IEEE.
[8] B. Raveendran Pillai, Prof. (Dr). Sukesh Kumar. A (2008): A Real-time system for the automatic identification of motorcycle - using Artificial Neural Networks, International Conference on Computing, Communication and Networking.
[9] Muhammad Tahir Qadri, Muhammad Asif (2009): Automatic Number Plate Recognition System For Vehicle Identification Using Optical Character Recognition, International Conference on Education Technology and Computer.
[10] Chen-Chung Liu, Zhi-Chun Luo(2010): An Extraction Algorithm of Vehicle License Plate Numbers Using Pixel Value Projection and License Plate Calibration, International Symposium on Computer, Communication, Control and Automation.
[11] Pletl Szilveszter, Gálfi Csongor(2010): Parking surveillance and number plate recognition application, IEEE 8th International Symposium on Intelligent Systems and Informatics.
[12] Mohamed El-Adawi, Hesham Abdel Moneim Keshk, Mona Mahmoud Haragi: Automatic license plate recognition.
[13] A. S. Johnson B. M. Bird, Department of Elect. & Electron. Engineering, University of Bristol: Number-plate Matching for Automatic Vehicle Identification.
[14] Maged M. M. FAHMY: Toward Low Cost Traffic Data collection: Automatic Number-Plate Recognition, The University of Newcastle Upon Tyne Transport Operations Research Group.
[15] Hwajeong Lee, Daehwan Kim, Daijin Kim, Sung Yang Bang: Real-Time Automatic Vehicle Management System Using Vehicle Tracking and Car Plate Number Identification.

Authors Biographies

Subhash Tatale, I am M.Tech Student.Having 4 yrs of experience in which 2 yrs of industry and 2 yrs of academic.My reasearch area is Image Processing.

Akhil Khare, I am associate professor working in Department of Information Technology.Completed M.Tech.and Pursuing Ph.D. in software engineering field.