ISSN No. 2455-5800
Journal of Scientific Research in Allied Sciences

**Review Article**

# 21 BASED FEATURE EXTRACTION OF A NUMBER PLATE IMAGE

**Sushmita Kumari, Sudha. S, K. Brindha**

*VIT University, Vellore - 632014, Tamil Nadu, India*

Article history:
Submitted on: August 2015
Accepted on: September 2015
Email: info@jusres.com

## ABSTRACT:

In this proposed paper a comprehensive survey on existing ANPR techniques by categorizing them according to the features used in each stage. Automatic car number plate recognition is a technology that enables computer system to read automatically the number plate of car. This proposed system is basically for Indian car number plates that consist of English characters and English digits. The ANPR system consists of four basic steps Plate Localization, Pre-processing, Classification and Feature Extraction. But in our System we are taking noise free number plates with font Arial, Times New Roman and Verdana. We have assumed that we are getting segmented data for training of characters and digits. The 21 based feature extraction method (zone based) and minimum distance classifier algorithms are used for this Automatic number plate recognition System. The characters and digits are recognized by using its ASCII values.

**Key Words: Plate Localization, Pre-processing, Classification, Feature Extraction.**

## INTRODUCTION

The Automatic Number Plate Recognition (ANPR) gained popularity throughout the last decade beside the advance of digital camera. It is merely the flexibility to mechanically extract and acknowledge a vehicle number plate's characters from an image.

The aim of this proposed paper is to detect and capture a vehicle number plate image. The method include extraction of number plate from the image, then using image segmentation algorithm to obtain individual character and then recognition algorithm is used for recognizing the individual character using feature extracted from the known data.

It has a wide range of applications. ANPR is utilized in several areas from speed social control and toll assortment to management of parking tons etc. It may also be accustomed observe and forestall a good vary of criminal activities and for security management of extremely restricted spaces like area around high government offices or military zones. The system estimation is economical compare to the other ANPR systems. ANPR is very useful in various applications like for tracking system, automatic traffic control and in high-way/parking as a toll collection system, or can be used in petrol stations. In this proposed paper number plate segmentation, recognition of character based on feature extraction and analyzing of recognized characters is described briefly.

In Indian number plates there are so many variations like:

- Quality: - Number plate image may contain unwanted images.
- Location: - People of different location use different languages.
- Size: - The capture image size may vary due to the zoom factor and camera distance.
- Colour: - Plates may have different character and background colour.
- Font: - Plates may be written in different languages and font style.
- Occlusion: - Plates may be covered with dirt.

**RELATED WORK**

The proposed system in [1][2][3] is based on Feature based technique using edge detection or Hough transform which is costly or use artificial neural network which need bigger data for training. In classical Hough transform boundary lines and shapes are detected in which line is changed into parameter and intersects after those two parallel lines are searched and the region in between lines is passed as a potential plate region.

In the proposed solution [4] segmentation based on black pixel projection has been implemented to recognize Turkish number plates in binary domain. In [4] localization of number plate in the image is done based on smearing technique. The binary images of vertical and horizontal runs were taken. It is followed by segmentation of the number plate from the image using threshold

values. The similar algorithm has been used to segment the character from the plate after the image has been filtered and dilated. In [4] cross correlation coefficient technique has been used to classify the text.

In existing paper [5] searching algorithm has been used which rely on color information. There are many types of searching algorithm to locate a vehicle number plate. In [5] color search algorithm has been used to extract the likelihood ROI from an image. This type of algorithm is fast but the main problem is it can detect only single color standardized number plate.

In proposed paper [6],[7] high number plate extraction rate is achieved based on mathematical morphology operations and vertical edging. The English characters are having vertical edges so it can be easily classified.

In proposed paper [8] a multi-layer neural network has been used. It consist of a input vector, several hidden layers, one output layer and a output vector where each layer consist of a set of neurons and corresponding transfer function. Parallelism and modularity is the main advantage computational characteristics of neural networks.

In proposed paper [9] Binary Associative Memories [BAM] is used in the form of neural network. It automatically read characters included in a number-plate. BAM is simple and direct neural approach which has the property of correcting the distorted input patterns. Basically BAM is having two problems. First the storage it varies by O (np) where n and p are input and output pattern lengths. Second is the capacity. The problem arose in this paper [9] is reliable retrieval of associations begun to degrade with more number of patterns stored.

In paper [10] new and fast vertical edge detection algorithm [VEDA] has been implemented number plate extraction. It is 7-9 times faster than Sobel operation.

In paper [11] block based feature extraction method has been used. Blocks having higher edge magnitude has been identified as number plate area. It can be used for image with unclear number plate boundary because block processing does not depend on the edge of the number plate boundary. In paper [11] the accuracy of 180 pairs of image was 92.5%.

| **Year** | **Accuracy** | **Method** |
|---|---|---|
| 1990 | 95.00% | Threshold algorithm |
| 2009 | 93.00% | Isolating algorithms |

| | | |
|---|---|---|
| 2009 | 87.00% | Otsu's method |
| 2009 | 85.00% | 'Image Scissoring' algorithm |
| 2010 | 90.00% | Adaboost |
| 2011 | 91.00% | Threshold algorithm |
| 2011 | 92% | Template Matching |
| 2012 | 96.20% | vertical edge detection algorithm |
| 2012 | 96.62% | contour detection algorithm |
| 2012 | 98.00% | Texture Features |
| 2012 | 97.30% | Timedelay neural network |
| 2012 | 97.30% | Feed forward ANN |

**Fig.2.1. Method Used for Number Plate Recognition**

## AUTOMATIC NUMBER PLATE RECOGINITION TECHNIQUE

The proposed system works in following order:-

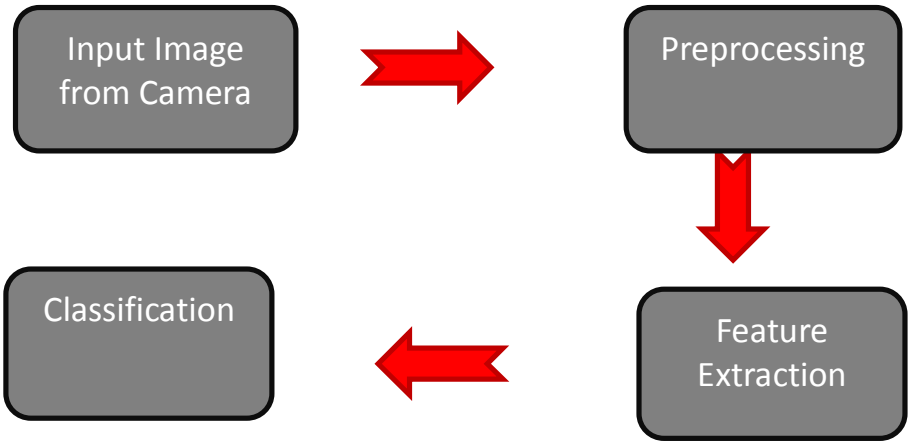


**Fig.3.1. Steps for processing**

**Input Image from camera**: - A car number plate image has been taken from a camera.


**Fig.3.2. Captured Image**

**Pre-processing: -** It can be accomplished by undergoing following process:-

- Image Cropping: - Removing unwanted areas.


**Fig.3.3. Cropped Image**

- Grayscale: - Converting the image into grayscale (0-255 pixel value)


**Fig.3.4. Grayscaled Image**

- Binarization of Image:- Binarization consists of conversion of grayscale image into binary image i.e., converting 0-255 pixels value into 0-1 pixel values.


**Fig.3.5. Binarized Image**

- Segmentation: - In this each and every character and digits of a number plate is separated.


**Fig.3.6. Segmented Image**

**Feature Extraction: -** We are using 21 based feature extraction method. In this method the whole rectangle is divided into 16 zones with corresponding mark. Then the entire rectangle is again divided diagonally starting from left side of the top of the rectangle towards the right side of the bottom of rectangle. It consists of Zone 17 consists of zone 1, zone 2, zone 5 and zone 6 values. Zone 18 consists of zone 1, zone 2, zone 3, zone 5, zone 6, zone 7, zone 9, zone 10 and zone 11 values. Zone 19 consist of all zone values i.e., from zone 1 to 16.Then again the entire rectangle is divided diagonally starting from right side of the top of the rectangle towards the left side of the bottom of rectangle. It consists of zone 20 consist of zone 3, zone 4, zone 7 and zone

8 values. Zone 21 consist of zone 2, zone 3, zone 4, zone 6, zone 7,zone 8,zone 10, zone 11,zone 12.

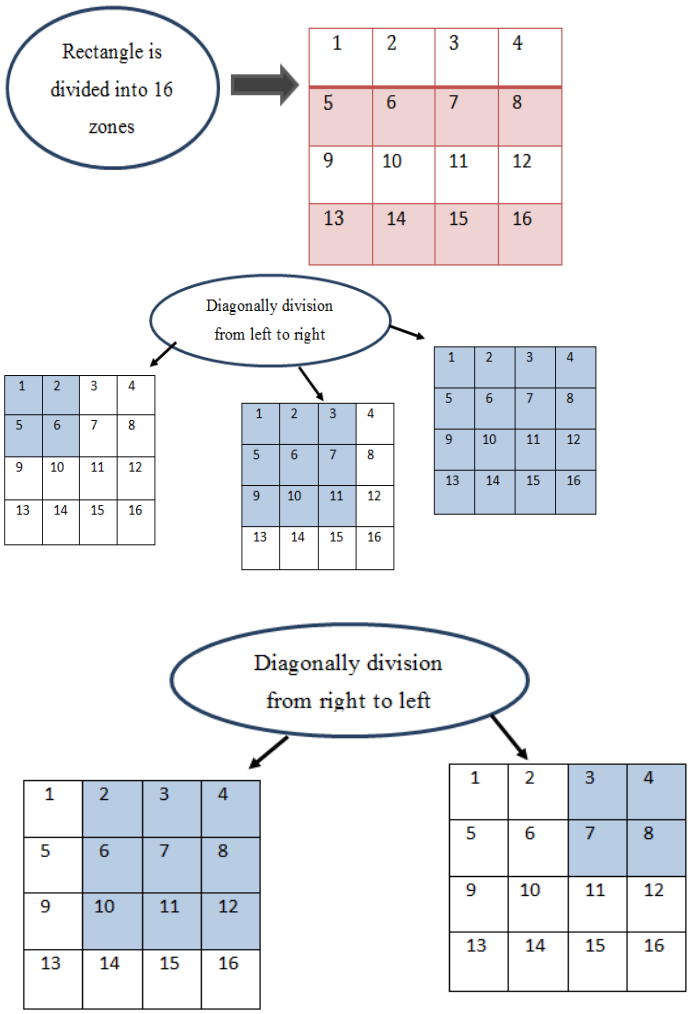


**Fig.3.7.Feature Extraction Steps**

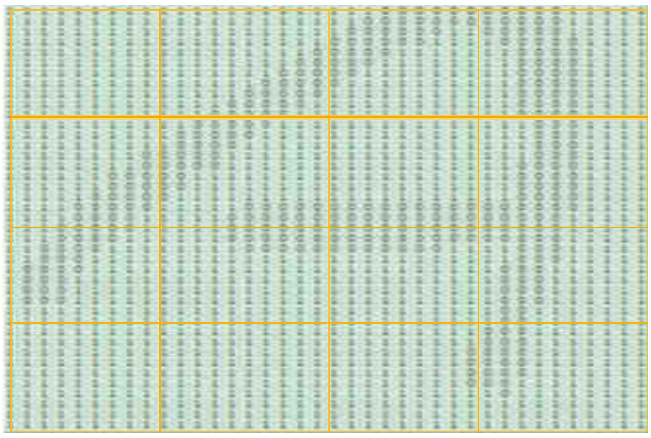

**Fig.3.8. Sample Image**

**Classification: -** In classification training and testing of the segmented numbers and character of the number plate is performed.

Training: - In training samples of same data is taken and pixel density of each sample is calculated and then average of those calculated pixel value has been taken out. At the end of training we will get 21 continues number and one ASCII value for each character and digit. The accuracy of algorithm depends on training. If number of samples is taken more the accuracy will be more and vice-versa.

Fig.3.9. Training of A of Arial Font Type

```
>> Digit_Recognition_Method3('A.jpg',65,1)

 Number of connected components = 53

feature_vector =

  Columns 1 through 8

    0.0046    0.8246    0.8280    0.0051    0.1703    0.8438    0.8572    0.1745

  Columns 9 through 16

    0.5057    0.8470    0.8500    0.5076    0.8423    0.5008    0.5045    0.8470

  Columns 17 through 22

    0.4608    0.6368    0.5696    0.4662    0.6375   65.0000

data already exist

Unicode Number = 65
 Trainning Completed

ans =

  Columns 1 through 8

    0.0046    0.8246    0.8280 `  0.0051    0.1703    0.8438    0.8572    0.1745

  Columns 9 through 16

    0.5057    0.8470    0.8500    0.5076    0.8423    0.5008    0.5045    0.8470

  Columns 17 through 22

    0.4608    0.6368    0.5696    0.4662    0.6375   65.0000
```

**Fig.3.10.Result of Training of A of Arial Font Type**

Testing: - Each set of character image is used for testing. In testing the process will first segment the car number plate. Then it will ask user to enter the expected Unicode for the corresponding character and then it will compare with trained data. At the end it will give the accuracy of the tested data. After training is completed a test pattern is given to test. The input character is compared with all trained digits and characters. It will show that value which has minimum difference with given input value.


**Fig.3.11. Testing of A of Arial Font Type**

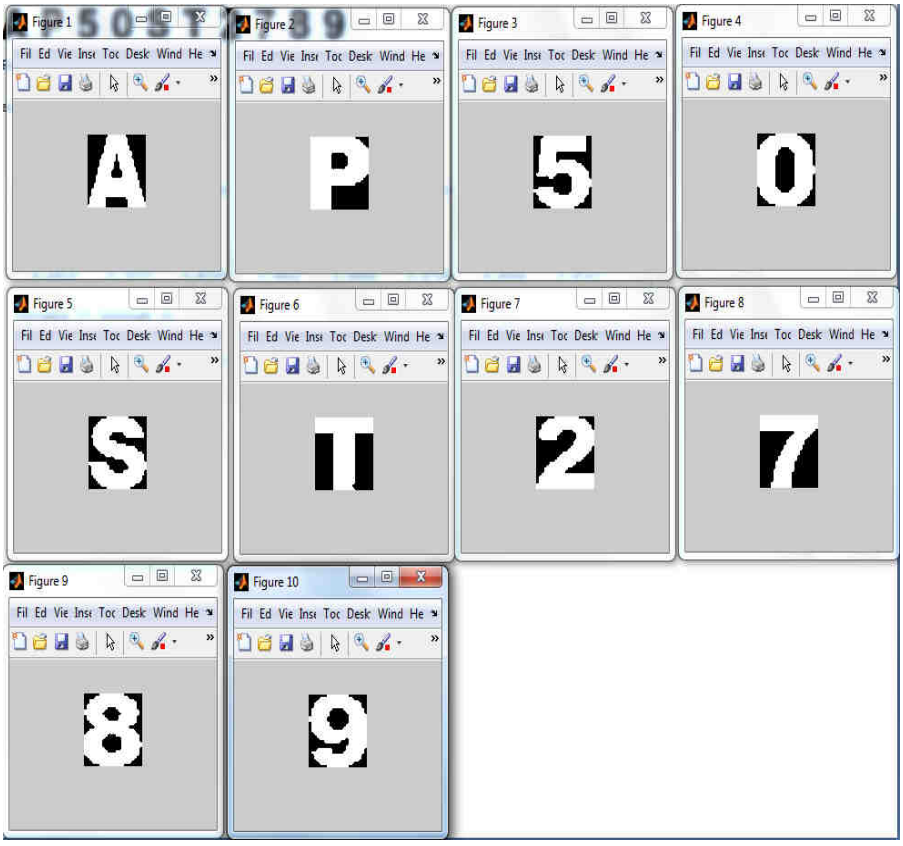


**Fig.3.12. Segmented Image**

```
Result is  = 100.000000

ans =

  Columns 1 through 8

    0.5078    0.8555    0.8555    0.4727    0.8711    0.4570    0.4023    0.9922

  Columns 9 through 16

    0.2813    0.7070    0.6758    0.9922    0.4648    0.8750    0.8555    0.4727

  Columns 17 through 21

    0.6729    0.6237    0.6711    0.6807    0.7122
```

**Fig.3.13. Result**

## EXPERIMENTAL RESULT

The proposed algorithm is implemented in Matlab software. The algorithm is tested and the system gives satisfactory results. The number plate for three different font style where tested. The overall accuracy rate is found to be 95% since the system get confused with 0[zero] and O. So the system should be improved further to avoid confusion.

| | ARIAL | TIMES NEW ROMAN | VERDANA |
|---|---|---|---|
| A | **100** | **100** | **100** |
| B | **100** | **98.214** | **100** |
| C | **100** | **91.379** | **100** |
| D | **100** | **94.444** | **100** |
| E | **100** | **96.666** | **97.368421** |
| F | **100** | **95.588** | **100** |
| G | **100** | **86.44** | **100** |

| | | | |
|---|---|---|---|
| H | 100 | 100 | 100 |
| I | 100 | 96.428 | 97.674419 |
| J | 100 | 100 | 100 |
| K | 100 | 0 | 100 |
| L | 85.294 | 84.126 | 100 |
| M | 96 | 95.744 | 100 |
| N | 100 | 97.826 | 100 |
| O | 100 | 92.727 | 98.507463 |
| P | 98.484 | 100 | 100 |
| Q | 100 | 23.333 | 100 |
| R | 100 | 100 | 98.571429 |
| S | 96.296 | 97.52 | 100 |
| T | 94.202 | 93.75 | 100 |
| U | 100 | 92.307 | 100 |
| V | 95.161 | 100 | 100 |
| W | 91.304 | 100 | 100 |
| X | 65.454 | 100 | 100 |
| Y | 100 | 98.305 | 100 |

| | | | |
|---|---|---|---|
| Z | **96.875** | **83.116** | **100** |
| 0 | **100** | **92.307** | **95.23** |
| 1 | **100** | **100** | **100** |
| 2 | **100** | **91.428** | **100** |
| 3 | **100** | **98.484** | **100** |
| 4 | **98.148** | **95.384** | **100** |
| 5 | **100** | **97.101** | **100** |
| 6 | **98.412** | **92.957** | **100** |
| 7 | **100** | **100** | **100** |
| 8 | **100** | **100** | **98** |
| 9 | **100** | **94.936** | **0** |

**Table .4.1. Comparison of Alphanumeric characters**

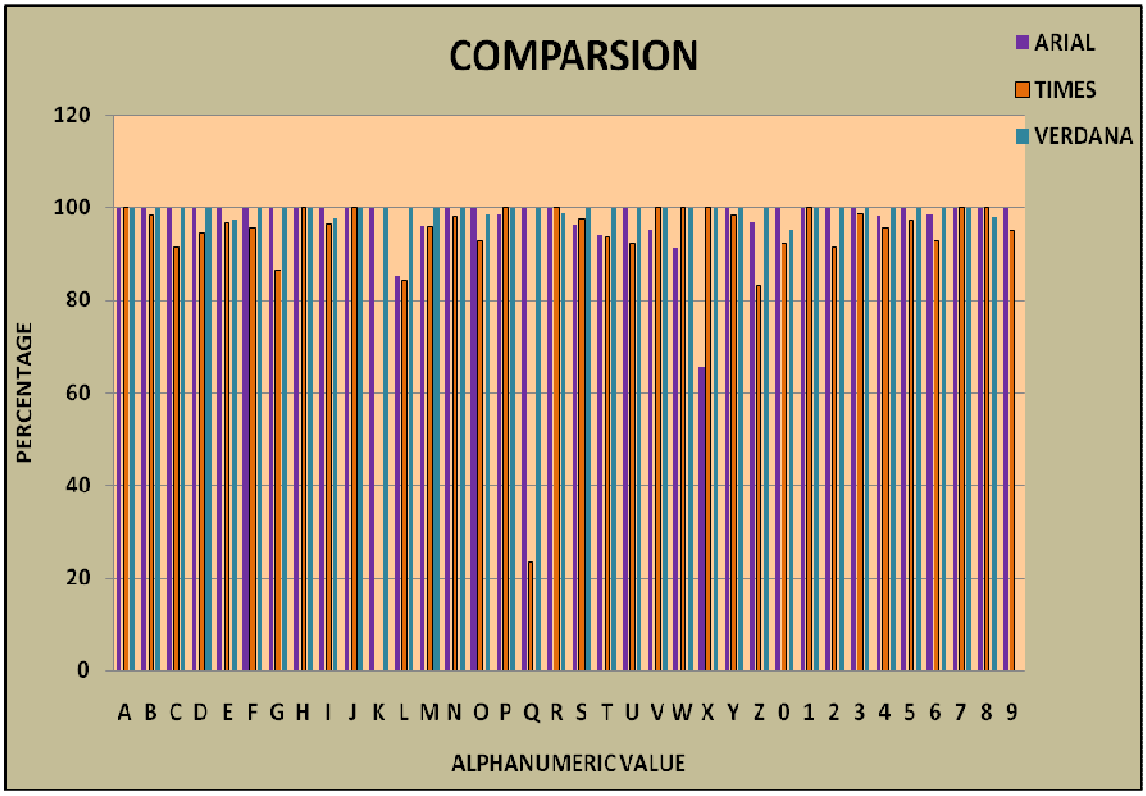


**Fig.4.2. Comparison of test result**

## CONCLUSION

The system provides satisfactory performance for wide deviation in illumination situation and for various kinds of number plates normally found in India. The proposed system is preferred more than the existing manual systems in India. But there are some constraints regarding their values like vehicle speed, script written on the number plate and screw which appear in the image can be replaced. The advantage of proposed system is that it provides high computational speed and it can be used for any font size because normalization is done before extraction. The drawback of this proposed system is it can be used only for three different font type Arial, Times New Roman and Verdana. It can be applicable for only English characters and digits because in India people use different language in number plate.In future we can implement different font types and different languages also and can use widely.

## REFERENCES


1. J.W.Hsieh, S.H.Yu and Y. S. Chen."Morphology based license plate detection from complex scenes". 16th International Conference on Pattern Recognition (ICPR'02), pp.79–179, 2002.
2. V. Kasmat, and S. Ganesan, "An efficient implementation of the Hough transform for detecting vehicle license plates using DSP's," IEEE International Conference on Real-Time Technology and Application Symposium, Chicago, USA, pp. 58-59, 2005.
3. S.H. Park, K.I. Kim, K. Jung and H.J. Kim, "Locating car license plate using Neural Network," Electronic Letters, Vol. 35, No. 17, pp. 1474 – 1477, 1999.
4. J.A.G. Nijhuis, M.H terBrugge, and K.A. Helmolt, "Car License Plate recognition with network and fuzzy logic", in Proc.Of IEEE International Conference on Neural Networks., volume 5, pp 2232-2236, Dec 1995.
5. M. Tahir Qadri, M. Asif "Automatic Number Plate Recognition System for Vehicle Identification using OCR," International Conference on Education Technology and Computer, pp 335 – 338, 2009.
6. A. Tahir, H. Adnan Habib, M. Fahad Khan "License Plate Recognition Algorithm for Pakistani License Plates," Canadian Journal on Image Processing and Computer Vision Vol. 1, No. 2, pp 30-36, April 2010.
7. F. Faradji, A. Hossein Rezaie, M. Ziaratban "A Morphological Based License Plate Locating System," IEEE International Conference on Image Processing (ICIP), pp 57-60, 2007.
8. H. Demuth, M. Beale and M. Hagan, Neural Network Toolbox 6 User's Guide, TheMath Works, Inc., 2008.
9. Kosko, B. (1987) Constructing an Associative Memory.BYTE, September 1987.
10. S. Wang and H. Lee, "Detection and recognition of license plate charactersWith different appearances," *Int. Conf. Intell. Transp. Syst.*, vol. 2, pp. 979-984, 2003.
11. H.-J. Lee *et al*, "Extraction and recognition of license plates of and vehicles on highways," *Int. Conf. Patt.Recog.*, pp. 356-359