

An Automatic Number Plate Recognition System using OpenCV and Tesseract OCR Engine

Andrew S. Agbemenu
Dept. of Computer Eng.
KNUST
Kumasi, Ghana

Jepthah Yankey
Dept. of Computer Eng.
KNUST
Kumasi, Ghana

Ernest O. Addo
Dept. of Computer Eng.
KNUST
Kumasi, Ghana

ABSTRACT

Automatic Number Plate Recognition (ANPR) is a fairly well explored problem with many successful solutions. However, these solutions are typically tuned towards a particular environment due to the variations in the features of number plates across the world. Algorithms written for number plate recognition are based on these features and so a universal solution would be difficult to realize as the image analysis techniques that are used to build these algorithms cannot themselves boast hundred percent accuracy. The focus of this paper is a proposed algorithm that is optimized to work with Ghanaian vehicle number plates. The algorithm, written in C++ with the OpenCV library, uses edge detection and Feature Detection techniques combined with mathematical morphology for locating the plate. The Tesseract OCR engine was then used to identify the detected characters on the plate.

General Terms

Number Plate Recognition, Character Localization, Character Extraction and Segmentation, Image Processing

Keywords

OpenCV, edge detection, template matching, morphology, Tesseract OCR engine

1. INTRODUCTION

Intelligent Transportation Systems (ITSs) have become commonplace in an attempt to improve the safety, security and mobility on our roads[4, 3]. Automatic Number Plate Recognition (ANPR) is one of the technologies employed in ITSs to identify vehicles by capturing and extracting vehicle registration from their number plates using image processing techniques. ANPR was invented in 1976 by the UK police[6].

The ANPR process typically involves three stages:

- plate detection
- plate segmentation
- character recognition

The plate detection stage is the most important as a failure at this stage immediately means complete failure of the algorithm. This stage largely depends on certain features that the number

plate should have. These features include shape, colour, height and width; symmetry and spatial frequency. The ability of the algorithm to detect the plate also depends on the conditions (lighting, visibility, image skew and camera quality) in which the image was captured, and the nature of image itself. Most ANPR algorithms add a preprocessing stage that involves grayscaleing, thresholding and noise removal techniques to prepare the image. The most common method of grayscaleing is simple averaging of RGB values in a colour pixel[2].

In most published ANPR systems like [7, 1, 5], proprietary tools like MATLAB are used. These tools are usually not affordable for use in developing countries.

Ghanaian vehicular number plates are standardised into two main classes:

- 'Long' plates which measure approximately 52cm by 13cm
- 'Square' plates which are 34cm by 17cm in dimension

Characters on the standard number plates typically begin with a two-letter regional or special code followed by a registration number between 1 and 9999 inclusive, and single letter or a two digit number suffix as shown in Figure 1.



Fig. 1. A standard Ghanaian number plate.

2. PROPOSED SYSTEM MODEL

The proposed ANPR system is made up of four block as shown in Figure 2. Our system is capable of using either Edge-detection or Template matching combined to mathematical morphology to extract the number plate from the input image. Character recognition is done by the open source Tesseract OCR engine.

2.1 Image Preprocessing

The preprocessing stage prepares the captured image for the whole plate recognition process. It involves reducing the cost of computing the image information. The coloured image from the camera is

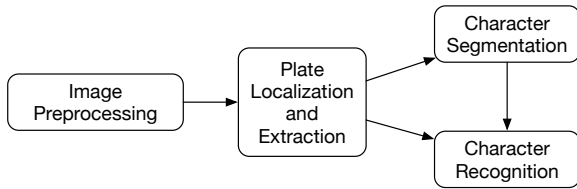


Fig. 2. Architecture of proposed ANPR model.

grayscaled using the Luma method employed by OpenCV. Equation 1 relates each resultant gray image pixel value, P_{gray} to its corresponding colour image RGB values: P_{red} , P_{green} , and P_{blue} .

$$P_{gray} = 0.299(P_{red}) + 0.587(P_{green}) + 0.114(P_{blue}) \quad (1)$$

A test image and its grayscaled version are shown in Figures 3 and 4. The gray image was then blurred to remove noise to improve plate localization and detection accuracy.



Fig. 3. Original sample test image used to test the proposed system.



Fig. 4. Grayscaled image of the original test image.

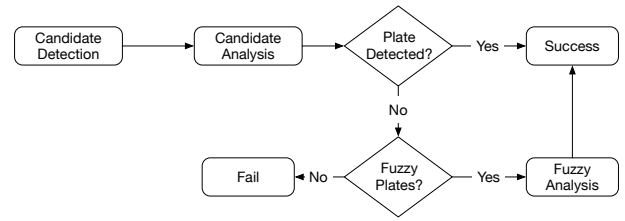


Fig. 5. Process flow for the plate detection and localization block.

2.2 Plate Localization and Extraction

After the captured image was preconditioned, the second block of our model was applied to identify the number plate and extract it for further processing. The flow chart in Figure 5 summarises the processes involved in finding and extracting the plate from the grayscaled image.

2.2.1 Candidate Detection. In localizing the plate, all areas in the image that may contain the number plate were identified and marked as candidates. Our proposed system offers two candidate detection algorithms: edge detection and template matching. The three-stage edge detection algorithm used focusses on enhancing the image and identifying the number plate edges in the image. In this algorithm, the grayscaled image is first convolved with a Gaussian kernel to reduce detail and remove noise as illustrated in Figure 6. Typically, number plates have strongly defined edges thus the Gaussian blur have no effect on them thereby reducing possible areas that the algorithm must consider.



Fig. 6. Grayscaled image smoothed using Gaussian image smoothening.

Vertical and horizontal Sobel-Feldman convolution kernels are applied separately to the blurred image, producing vertical and horizontal edge images shown in Figures 7 and 8 respectively. These filters locate areas of high spatial frequency or large pixel value differences and mark them as edges. After both convolutions, the two edge images are combined into a full edge image and binarized using Otsu's simple but effective global thresholding method [8]. Figure 9 is the result of the edge image unification and binarization.

Connected component analysis was performed on the binarized image to locate the objects embedded within (Figure 10). The bounding rectangle of each identified object is run through a size filter to take out those that are either too large or too small to be number



Fig. 7. Horizontal Sobel-Feldman convolution applied to image to determine horizontal edges of the plate.



Fig. 8. Vertical Sobel-Feldman convolution applied to image to determine vertical edges of the plate.



Fig. 9. Binarized image of combined horizontal and vertical edges.

plates. An aspect ratio filter to select rectangles whose width-height ratios correspond to specified plates aspect ratios. Objects that pass both filters are tagged as plate candidates. A number plate typically appear as a rectangular area with a white border after edge detection and binarization as shown in Figure 11.

Template matching, on the other hand, involves the use of trained classifiers to detect features of number plates learned. In training

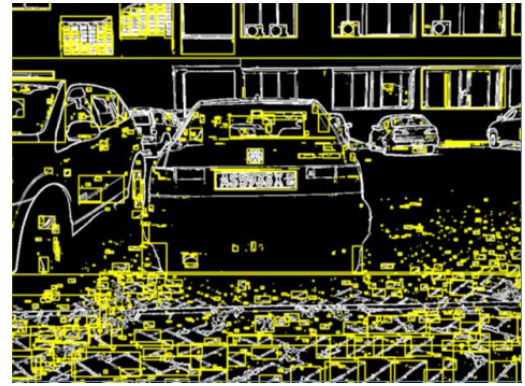


Fig. 10. Image showing all detected edge components.

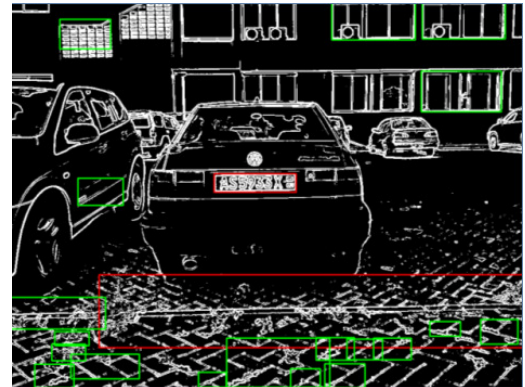


Fig. 11. Selected plate candidates from edge detection.

these classifiers, two types of images were obtained and used in the process: positive images and negative images. Positive images are those that contained number plates while negative images did not contain any plate at all. A library of 303 rectangular plate and 359 square plate images was compiled and used in training 10 rectangular and square plate classifiers each. The trained classifiers were then applied to the grayscale image for candidate plate identification. Figure 12 illustrates the detection of plate candidates using classifiers.

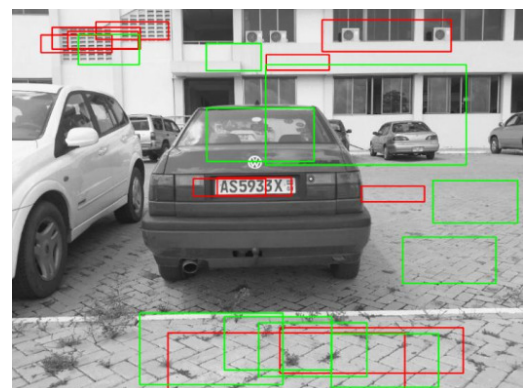


Fig. 12. Plate candidate detection with classifiers.

2.2.2 *Candidate Analysis.* After detection, the coordinates of candidate areas were used to examine corresponding areas on the binarized form of the original image. Connected component analyses, summarised in Figure 13, were performed on corresponding candidate areas to locate the objects on the number plate. The size and aspect ratio of each object is then tested. A character count is maintained for each object that passes both filters. Each successful object was also stored. Ghanaian number plates have at least four characters thus any candidate with a character count of four or higher is marked as a strong candidate. Any candidate with a component count of two or three is tagged as a fuzzy plate. Figures 14 and 15 illustrate component and character detection on a candidate plate.

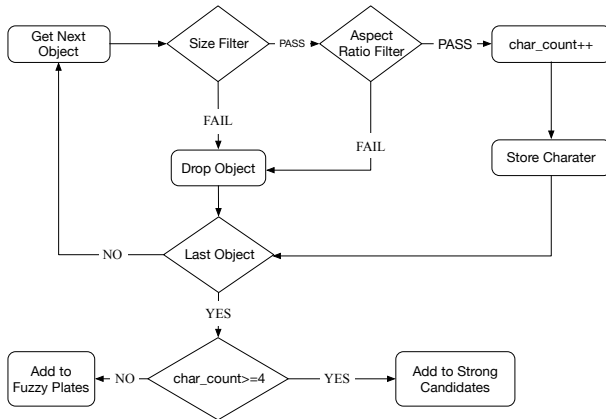


Fig. 13. Process flowchart for candidate analysis.

The set of strong candidates were re-evaluated to get the best candidate after initial candidate analysis. The candidate with the highest character count in the set is selected. If multiple candidates have that maximum value, the candidate with the best bump score is selected. If the set of strong candidates is empty, the set of fuzzy plates is evaluated. This secondary assessment is done using both the character counts and their bump scores of the fuzzy candidates. The selected plate is then cut out of the original image and binarized for further processing.

2.3 Character Segmentation

Our system implements character segmentation before character recognition to forestall effects of noise, character arrangement and skew especially with square plates. Each character was first cut out of the plate image and de-noised. A new image was created and the de-noised characters were placed on a single line in the new image as in Figure 16.



Fig. 14. Object detection on successful plate candidate.



Fig. 15. Character detection on successful plate number object candidates.



Fig. 16. Image with segmented characters.

2.4 Character Recognition

This extremely crucial process forms the final stage in the ANPR process. Here, the Tesseract OCR engine is applied to the detected plate to recognize the alphanumeric characters on it. The Tesseract engine used was trained in order to increase the accuracy of the recognition process. The training involved creating images of the characters to be recognized, using the expected fonts, and compiling a dictionary of possible character combinations to be found on a number plate, that is: regional codes, suffixes and registration numbers.

The output of this stage is the text representing the vehicle number.

3. TESTING, RESULTS AND DISCUSSION

500 number plates were processed with the proposed ANPR system to test its performance in terms accuracy and speed. of 10 results The edge detection algorithm detected 397 plates (79.4%), with an average speed of 0.037s. Feature detection was much slower, but had higher accuracy, detecting 454 plates (90.8%) with an average speed of 0.185s. The optical character recognition added an average of 0.031s to the process, and successfully recognised about 60% of detected plates. Faded characters and certain plate decorations, along with noise on the plates due to dirt, greatly hampered the plate segmentation process, which in turn affected character recognition. Both plate detection algorithms proved reasonably accurate up to a distance of 5 metres. Feature Detection exhibited random successes with variations in view angle, while Edge Detection stayed accurate for a range of up to 30 degrees.

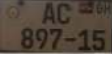

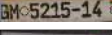
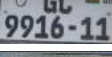
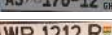
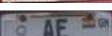

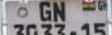
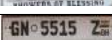

4. CONCLUSION

The proposed ANPR detected most of the Ghanaian number plates tested with a successful recognition rate of 60% with an average processing time of about 0.2s to complete the entire image capturing to character recognition stage. This system will produce more accurate results upon further training.

5. ACKNOWLEDGEMENT

Our thanks to all vehicle owners who allowed us to run our system on their number plates. We would also like to thank Jeffery Hooper for his immense contribution to the work.

Table 1. Sampled results detected and non-detected plates using and proposed ANPR system

Plate in image	Template Matching			Edge Detection		
	Plate detected?	Result	Latency(secs)	Plate detected?	Result	Latency(secs)
	Yes	A 89715	0.146	No	-	-
	Yes	AE 1L01 12	0.153	Yes	AE 110112	0.062
	Yes	GM 5215 14	0.220	No	-	-
	Yes	GC 9916 11	0.193	Yes	GC9916 11	0.067
	Yes	AS 176 12	0.126	Yes	AS 176 12	0.053
	Yes	WR1212R	0.226	Yes	WR1212R	0.076
	Yes	AE 1380 12	0.140	Yes	AE 1380 12	0.059
	Yes	AW 5658 14	0.251	Yes	QAW 5658 14	0.082
	Yes	GN 3033 15	0.199	Yes	GN 3033 1	0.079
	Yes	GN 5515 Z	0.201	Yes	GN 5515 Z	0.057

6. REFERENCES

- [1] K. M. Babu and M. V. Raghunadh. Vehicle number plate detection and recognition using bounding box method. In *2016 International Conference on Advanced Communication Control and Computing Technologies (ICACCCT)*, pages 106–110, May 2016.
- [2] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8, 1986.
- [3] W. L. Hao and H. T. Yong. Detection of license plates in natural scenes with msr and sift unigram classifiers. *IEEE Conference on Sustainable Utilization and Development in Engineering and Technology*, pages 95–98, 2010.
- [4] T. B. Joewono and H. Kubota. Safety and security improvement in public transportation based on public perception in developing countries. *IATSS Research*, 30(1):86 – 100, 2006.
- [5] J. A. Mayan, K. A. Deep, M. Kumar, L. Alvin, and S. P. Reddy. Number plate recognition using template comparison for various fonts in matlab. In *2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, pages 1–6, Dec 2016.
- [6] M. T. Qadri and M. Asif. Automatic number plate recognition system for vehicle identification using optical character recognition. In *2009 International Conference on Education Technology and Computer*, pages 335–338, April 2009.
- [7] S. Singh and B. Kaur. Number plate recognition through image using morphological algorithm. In *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, pages 3157–3160, March 2016.
- [8] H. V. Vala and A. Baxti. A review on otsu image segmentation algorithm. *International Journal of Advanced Research in Computer Engineering and Technology*, page 387, 2013.