Journal of King Saud University – Computer and Information Sciences 33 (2021) 399–407

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

Vehicle plate number localization using a modified GrabCut algorithm

Ayodeji Olalekan Salau[a,*], Thomas Kokumo Yesufu[a], Babatunde Sunday Ogundare[b]

[a] Department of Electronic and Electrical Engineering, Obafemi Awolowo University, Ife-Ife, Nigeria
[b] Department of Mathematics, Obafemi Awolowo University, Ife-Ife, Nigeria

ARTICLE INFO

Article history:
Received 6 November 2018
Revised 23 January 2019
Accepted 23 January 2019
Available online 29 January 2019

Keywords:
Vehicle plate number
License plate (LP)
Localization
GrabCut
Aspect ratio
Graph-cut

ABSTRACT

Vehicle plate number recognition plays an important role in traffic control and surveillance systems. A key stage in any vehicle plate number recognition system is to first locate the vehicle plate number. In this paper, we present a modified GrabCut algorithm for localizing vehicle plate numbers. In contrast with the traditional interactive GrabCut technique, a modified GrabCut algorithm was designed to identify and extract vehicle plate numbers in a completely automatic manner. Our approach extends the use of the traditional GrabCut algorithm with addition of a feature extraction method which uses geometric information to give accurate foreground extraction. Finally, to evaluate the performance of the proposed technique, the localization accuracy is tested with a dataset of 500 vehicle images with vehicle plates from different countries. An accuracy of 99.8% was achieved for the localization of vehicle plates. Comparative analysis is also reported.

1. Introduction

Vehicle plate number (VPN) recognition finds applications in a number of traffic control, monitoring and surveillance systems (Fu et al., 2004; Anagnostopoulos et al., 2006; Badr et al., 2011). In most studies, the stages identified as the major areas of interest for VPN recognition are: locating the license plate (LP) in the image (plate number localization) (Li and Xie, 2007; Al-Hmouz and Aboura, 2014) and reading the text from license plates (LPs) (character recognition) (Ghahnavieh et al., 2014).

An important and basic stage before character recognition as stated in Ghahnavieh et al. (2014), is the detection of the license plate (Al-Hmouz and Aboura, 2014). This is identified as the most crucial stage in any VPN recognition system (Halin et al., 2013; Hemayat et al., 2014).

Over the years, license plate localization has been an active area of research and still continues to stir up interest of most researchers in the field of VPN recognition and intelligent transportation systems (Liu et al., 2017). Localization of LP is a challenging task due to the wide variation in LP features namely: plate size, shape, color, texture, spatial orientation and position of plates. The non-adherence of LPs to any particular country-specific standard, style or font format has caused VPN recognition systems to be computationally intensive, error prone and to have long processing times (Saha et al. 2009; Chen et al. 2009; Salau, 2018). Subsequently, these factors make it difficult for VPN recognition systems to easily locate LPs from captured vehicle images. This paper, addresses the problem of vehicle plate number localization from simple and complex backgrounds using a novel approach which is based on a modified GrabCut algorithm as an automatic localization technique for identification and extraction of LPs from captured vehicle images.

The remainder of this paper is organized as follows. Section 2 provides a review of license plate localization techniques and image segmentation techniques. The modified GrabCut algorithm is presented in Section 3 and a detailed explanation of its modification is given therein. Section 4 presents the experimental results and gives an evaluation of the performance of the proposed technique, while the conclusion is presented in Section 5.

2. Related works

License plate detection and extraction is a major stage in VPN recognition. Most of the proposed algorithms for LP detection have located the LP from the vehicle image but have issues with sensitivity to brightness, long execution time and low rate of accuracy (Hemayat et al., 2014). In Musoromy et al. (2010), a comparison of the performance of image enhancement filters such as Sobel, Canny

* Corresponding author.
E-mail address: ayodejisalau98@gmail.com (A.O. Salau).

Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.jksuci.2019.01.011
1319-1578/

and others was presented. They combined connected component analysis (CCA) to extract the potential LPs. Existing LP localization techniques use information from the vehicle image to aid the identification of the potential LP region. This information can be categorized as either boundary or region based information or both.

In Hongliang and Changping (2004), boundary information was used to extract the horizontal and vertical edge maps of LPs. Furthermore, smoothing and normalization was performed on the resulting image and edge detection was performed on the gray image to locate the LP. The final decision was made using CCA and edge statistics. Similarly, another approach was proposed in Saha et al. (2009) which used an edge-based multi-stage method to detect the various edges of LPs. The approach works well in varying weather conditions and also for images with low contrast which have poor image quality. Hough Transform (HT) was proposed in Yanamura et al. (2003) to find the edges of the rectangular plate. In addition, the authors have stated the drawbacks of the HT approach which are; it requires longer processing time and demands a large storage space. Authors in Ullah and Lee (2016), proposed an effective LP extraction algorithm based on geometrical features and multilevel thresholding to identify and segment LPs from a captured vehicle image. The extraction algorithm gave an average localization accuracy of 75% while detecting license plates. However, it had some challenges in detecting plates with poor image quality and also damaged plates.

In addition to boundary based approaches, region based approaches have also been employed. Using region information, Zheng et al. (2005) employed color characteristics of plates to detect license plates. The RGB color space is transformed into HSI color space and only four colors are taken into account namely: white, black, red and green. Another region-based method was introduced in Jia et al. (2007). This method was modified into a new method in Mahmood et al. (2013) to detect Iranian license plates. Morphological operation was further applied to the candidate region of the Iranian LPs to remove unwanted edges. Finally, using the geometric features of the candidate region (e.g. aspect ratio, edge density and area), the LP was localized accurately. The aspect ratio of a vehicle plate number is the ratio of its width to its height. In the work of Yang et al. (2012), color based methods were classified into three main categories namely: Texture-based color methods (Wang et al., 2011), Background-color-based methods (Jia et al., 2005) and Color-texture-based methods (Youssef and Abaelrahaman, 2008). This categorization further helps to give a clearer understanding of color-based methods. Color is a distinct feature of a VPN and is most suitable for VPN detection when the lighting conditions are unchanged or stable (Lalimi et al., 2013).

In some works, a combination of two or more techniques which use boundary or region information have been proposed. This approach has been employed to provide a better detection rate or accuracy. In Abolghasemi and Ahmadyfard (2007), the authors proposed the use of edge statistics and color features to detect the LP. Similarly, Lalimi et al. (2013) employed the use of vertical edge detection and geometrical features to detect the LP region. In Wen et al. (2013), VEDA and morphological filter is employed to detect LPs. Ashtari et al. (2013) proposed a modified color template matching technique which uses support vector machine (SVM) to locate Iranian LPs. In order for LP recognition algorithms to be applied in real-time license plate recognition (LPR) systems, both high accuracy and quick response time are required (Wang et al., 2011). To calculate the accuracy of license plate localization techniques, Deshpande et al. (2015) have presented Eq. (1).

$$\%\,Accuracy\ of\ license\ plate\ localization = \frac{Successful\ Samples}{Total\ number\ of\ Samples} \times 100 \tag{1}$$

Similarly, segmentation techniques are also used for LP detection. They are categorized into those using either region or boundary features to extract license plates. They include techniques such as blob analysis, CCA, live-wire and color matching detection techniques. More interesting, is the use of the graph-cut segmentation technique which uses both region and boundary information to perform an efficient segmentation (Lv et al., 2003). The graph-cut algorithm is a typical algorithm for performing foreground and background segmentation (Boykov and Jolly, 2001). To reduce the number of manual interactions of graph-cut, an iterated graph-cut algorithm called GrabCut was developed by Rother et al. (2004). This new technique attempted to solve the problem of efficient, interactive extraction of a foreground object in complex environments whose background cannot be trivially subtracted.

It requires user interaction as the user need to draw a bounding box around the region of interest (ROI). As an interactive segmentation method, Jahangiri and Heesch (2009) presented an unsupervised GrabCut algorithm obtained with active contours which was initialized by a coarse segmentation. The algorithm was only able to segment the required foreground object from a plain background. Furthermore, Zhou et al. (2014) presented a semi-automatic GrabCut segmentation method which has the potential to become a fully automatic process by incorporating the object-detection result as a pre-processing step.

In Khattab et al. (2016), a new clustering technique using K-means and Fuzzy C-means was introduced to automate GrabCut and improve its segmentation accuracy. This technique was called the "SOFM clustering technique" for GrabCut automation as a replacement to the user interaction required by the traditional algorithm. To make GrabCut give a globally optimal solution, the cut is performed at the boundary region between the object and its background using Eq. (2).

$$|\mathbf{C}| = \sum_{e \in C} We \tag{2}$$

where **|C|** is the cost of the cut and W_e is the weight on the edge. To perform GrabCut, the image is converted to a graph which is made up of nodes and edges. The boundary and region information of the graph form an energy equation. The energy equation is minimized to give an accurate segmentation of the desired region. The energy equation is given as Eq. (3).

$$E(L) \ = \ \alpha R(L) \ + \ B(L) \tag{3}$$

where $E(L)$ is the energy function, $R(L)$ is the regional term which contains the region information. $B(L)$ is the boundary term which contains the boundary information. α is the relative importance factor between the regional and boundary term. When α is set to zero (0), only the boundary information is used, this means region information will not be considered. The regional term $R(L)$ and boundary term $B(L)$ are represented by Eqs. (4)–(9) respectively.

$$R(L) \ = \sum \mathrm{Rp}(\mathrm{lp}) \tag{4}$$

where Rp is the penalty of assigning the label lp to pixel p

$$Rp(1) = \ -log\ Pr\ (lp|'obj') \tag{5}$$

$$Rp(0) = \ -log\ Pr(lp|'bkg') \tag{6}$$

From Eqs. (5) and (6), it is observed that Rp(1) is smaller than Rp(0) when $Pr(lp|‘obj’|)$ is greater than $Pr(lp|‘bkg’|)$. This implies that when the pixel is more likely to be the object, the penalty for grouping the pixel into an object will be smaller. This will reduce the value of the energy function in Eq. (3). The boundary information is represented as:

$$B(L) = \Sigma[(B(p, q).\ \delta(lp,\ lq))] \tag{7}$$

$$\delta(\mathrm{lp},\ \mathrm{lq}) = \begin{cases} 1, & \text{if } \mathrm{lp} = \mathrm{lq} \\ 0, & \text{if } \mathrm{lp} \neq \mathrm{lq} \end{cases} \tag{8}$$

$$B(\mathrm{p}, \mathrm{q}) = \exp\left(-\frac{(\mathrm{lp} - \mathrm{lq})^2}{2\sigma^2}\right) \tag{9}$$

where *p, q* are neighboring pixels, σ is the camera noise, 'obj' represents Object and 'bkg' represents Background.

These equations are used for regions with similar or varying colors in a vehicle image. Gaussian Mixture Model (GMM) is used to cluster regions of similarities. It uses the Orchard and Bouman Clustering technique to further obtain the plate location by clustering regions of similar colors (Orchard and Bouman, 1991). Authors in Azam and Islam (2016), have stated that a good LP recognition system should provide accurate localization in good time. To this end, in this work we introduce shape prior information to give an initial trimap and to effectively locate the license plate region. GMM is used to cluster regions in the trimap that have the highest color densities or color clusters. These are used to identify the license plate location.

2.1. Challenges with existing techniques

Detection of LP is a difficult task for LP recognition systems (Tarabek, 2012). Algorithms which have been proposed in literature are designed based on techniques which use either boundary (edge-base features) or region features or both to locate LPs. Such techniques include: Blob detection (Kang, 2009), Color detection (Wang et al., 2008; Yang et al., 2012), Hough Transform (Duc et al., 2005; Angel et al., 2014), Histogram processing (Deshpande et al., 2015) and Edge detection (Musoromy et al., 2010; Liu et al., 2017) to list a few. These techniques can be broken down into two parts namely; those that use boundary features and those that use region features. Kang (2009) used blob detection to locate the LPs by employing a dynamic programming-based method. The distance between characters in the LP was used to determine the plate location which took 3 s to process. There findings showed that the distance between the characters in a LP are different for different states and different countries as shown in Fig. 1. The disadvantage of this technique is that the segmentation phase must precede the localization phase to extract the possible region of the plate. This makes it to have a high computational time. Wang et al. (2008) presented an algorithm based on fuzzy logic to tackle some of the challenges experienced in LPR systems by using color recognition (i.e. color features of LPs appear different in different scenes, poor illumination and camera characteristics). Furthermore, authors in Yang et al. (2012) combined color information with template matching algorithm for LP localization. The shortcoming of this method of localization is that it becomes difficult to locate the LP if the license plate number or its characters have the same color with the vehicle. A similar challenge is encountered when the algorithm is used to locate license plates (LPs) with different colors within a country. This technique cannot be used widely, since vehicles have varying colors and also varying color of LPs (Wang et al., 2011). In Li and Xie, (2007), an example of the variation between license plate colors in two different states within Tanzania was presented. This is shown in Fig. 2. A similar variation in color of three different countries LP within Africa namely: Nigeria, South Africa and Ghana is shown in Fig. 3. Sobel edge detection method was employed in Li and Xie (2007) to find the location of the LP from vehicle images. Noise removal and smoothing techniques using median filters was employed to improve the quality of the localized image. A comparison of license plates of Nigeria, South Africa and Ghana was presented in Danbatta et al. (2016).

The authors performed the experiment using blob detection to locate license plates of vehicles. The proposed method was implemented with Matlab R2015a and was tested with numerous vehicle images. An accuracy of 91.42% was achieved for localization with a processing time of 15 s. They concluded that Nigerian license plates are difficult to locate and also complex to process using image processing tools because of their complexity which include: the many inscriptions beside the license plate number, the image of the Nigerian map, Nigerian flag, State name and the inscription "Federal Government of Nigeria", etc. Similarly, in Owamoyo et al. (2013), Nigerian license plates extraction was performed using Sobel filter, morphological operations and CCA. A localization accuracy of 79.8% was achieved. In Li and Xie (2007) and Danbatta et al. (2016) color based techniques will not work efficiently because of the different variations in color of the foreground and the background of the LP. In Danbatta et al. (2016) a similar challenge was encountered as presented in Kang (2009). LP detection and localization is greatly affected because of the differences between blob (character) spaces and size. Furthermore, a different challenge is encountered in Kim (2010) as shown in Fig. 4.

The difference in LP size amongst other issues is also a challenge. In the light of this, various algorithms have been proposed to detect LPs irrespective of their differences in size, shape and color. In Duc et al. (2005), Hough Transform (HT) was employed for line and edge detection to locate license plates. Although HT is superior for finding lines and shapes out of edges in an image, it is computationally intensive, time consuming and demands a

Fig. 1. Blob detection using Dynamic Programming-Based Method (Kang, 2009).

Fig. 2. Color variations of license plates within Tanzania (Li and Xie, 2007).

Fig. 3. Color variations of license plates of Nigeria, South Africa and Ghana (Danbatta et al., 2016).

Fig. 4. Variations in license plate size and shape (Kim, 2010).

large memory space (Yanamura et al., 2003). To this end, numerous techniques/algorithms have not proved adequate to improve the localization accuracy coupled with their long processing time for LP localization.

2.2. Performance evaluation of existing techniques

License plate localization is affected by its recognition time (processing time) (Owamoyo et al., 2013). Finding a technique that provides a good accuracy with a high response time is difficult (Mobarhan et al., 2012). Various techniques and algorithms have been proposed. The experimental results obtained for their recognition accuracy and processing time is presented in the later part of this paper. The level of accuracy of recognition is vital for license plate localization (LPL) as it is the preliminary stage before character recognition. Failure of this stage will lead to subsequent failure in the proceeding stages (Tadic et al., 2016).

3. Methodology

In this section, we present our modified GrabCut algorithm for LP localization and also discuss some image processing stages performed in the process. The stages involved in the process are: preprocessing; which includes resizing the image to a certain size, the introduction of shape prior information into the traditional GrabCut algorithm to perform localization, converting the Graph-Cut image to grayscale to remove luminance and brightness, removing noise from the resultant grayscale image and converting the filtered image to a binary image so as to provide easier interpretation and computation with a computer system.

3.1. Data acquisition

The data collected for this work are vehicle images. Precisely, 500 vehicle images were acquired. The acquired images are colored Jpeg images and are stored in a folder on the system used for the development. 200 out of the 500 acquired vehicle images were captured via a digital camera (Nixon D7000), while the remaining 300 vehicle images with other countries LPs were obtained from online databases namely: CVonline: http://homepages.inf.ed.ac.uk/rbf/CVonline/Imagedbase.html, http://www.medialab.ntua.gr/research/LPRdatabase.html and CVpapers: http://www.cvpapers.com/datasets.html, to make up a total of 500 vehicle images used for the experiment. On the average, the captured images were taken seven feet away from the vehicle.

Fig. 9 shows some of the vehicle images out of the 500 that were used for the experiment. The selected vehicle images are a mixture of the acquired vehicle images that were captured and those acquired from online databases (vehicle plates of different countries). The proposed localization algorithm is designed using OpenCV with Python programming language and tested on 500 vehicle images to evaluate its performance. The Nixon D7000 digital camera has a resolution of 640 $\times$ 480.

3.2. Image pre-processing

Image pre-processing is a set of algorithms used to enhance the quality of an image. In this stage, the images are resized to the same size. The processing of images is done with a PC of Intel Core i5, 750 GB internal memory, 8 GB RAM and 2.6 GHz processor.

3.3. Proposed algorithm

In this section, we have succeeded in developing a vehicle plate number localization algorithm that can be used to locate vehicle plates of varying sizes, foreground and background colors, plate orientations and in addition, vehicles captured in complex scenes. The designed algorithm for vehicle plate localization is shown in Table 1 and the procedure of our proposed method is summarized in the flowchart in Fig. 5.

An initial trimap is required for the GrabCut process. In this work, aspect ratio (A_r) is introduced as a property for geometric feature extraction of rectangular features to create a trimap in the image. The aspect ratio is the ratio of the object weight to its height $\left(\frac{w}{h}\right)$ (Jia et al., 2007). The trimap is a combination of the three things; the foreground, background, and the unknown part of the image that can be either foreground or background. The introduction of A_r automates GrabCut and removes the user interaction required to obtain the initial trimap.

After the user interaction is removed, an initial segmentation is performed labeling the regions outside the rectangle as background and the region of the rectangle as foreground by a computer system. The resulting image is converted into a graph and

Table 1
Proposed localization algorithm.

Plate Localization Algorithm
Requirement: vehicle image
Ensure the image is a vehicle image and it is a color image
1. Vehicle Image $\leftarrow$ (input Image)
2. Resize Image $\leftarrow$ (input Image)
3. Create a trimap $\leftarrow$ use $2 \leq$ aspect ratio ≤ 5, ((x, y, w, h), $\left(\frac{w}{h}\right)$)
4. Label foreground (fg) & background (bg) pixels (p) $\leftarrow$ (Computer hard-labels, p $\in$ (0, 1))
5. Model the fg and bg $\leftarrow$ (model using GMM)
6. Create new pixel distribution $\leftarrow$ (from learned GMM)
7. Graph is built from the pixel distribution $\leftarrow$ (Graph-Cut is performed)
8. Convert Graph-Cut image to grayscale $\leftarrow$ rgb2Gray (output Image from 8)
9. Remove noise from grayscale image $\leftarrow$ (filter out noise with laplacian filter from output Image from 8)
10. Convert grayscale image to binary Image $\leftarrow$ (thresholdImage; threshold = 0–127 as 0 & 128–255 as 1)
11. License Plate located $\leftarrow$ (yes candidate region located, return sub-Image of GrabCut (localized plate))
12. Save extracted license plate image to database
13. End if $\leftarrow$ stop program
14. Else return stage 7.
15. Continue until the classification converges.

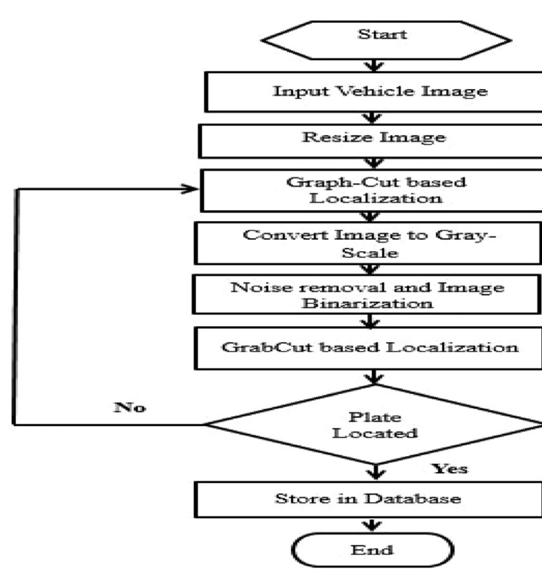

Fig. 5. Operational flowchart of the proposed algorithm.

graph-cut is performed. This leaves the required region on the graph and eventually, GrabCut grabs the region of interest out. In this work, because of the introduction of aspect ratio (A_r), the Eq. (3) becomes:

$$E(L, Ar) = \alpha R(L, Ar) + B(L, Ar) \quad (10)$$

where α or R(L, A_r) is zero (0), then Eq. (10) becomes Eq. (11).

$$E(L, Ar) = B(L, Ar) \quad (11)$$

Eq. (11) indicates that the introduction of A_r is sufficient to locate the plate boundary and extract the LP. The energy function, E (L, A_r) and the cost of the cut **|C|** is greatly minimized with Eq. (11).

If the region term in Eq. (3) is required, then GMM component is calculated with Eq. (12).

$$D_m = \log \sum \Pi i \frac{1}{\sqrt{|\sum|}} \exp^{(1/2[Zm-\mu]-\Sigma^{-1}[Zm-\mu])} \quad (12)$$

π_i is the weight of the image (a real), $|\sum|$ is the determinant of the covariance matrix (a real), μ is the mean (an RGB color), Z_m is the color of pixel m, T is the target region which is represented as 1 (foreground region) or 0 (background region) and Σ^{-1} is the inverse of the covariance matrix (a 3 × 3 matrix). Where m is a pixel which belongs to either the foreground or background GMM, D_{fore} or D_{back} respectively. D_{fore} and D_{back} are functions of the likelihood that the pixel m belongs to the foreground and background GMMs. These likelihoods are computed for pixel m using Eq. (12).

Table 2
Result of the accuracy of license plate localization.

Total number of vehicle images	500	Percentage 100%
Plates localized correctly	499	99.8%
Plates localized incorrectly	1	0.2%
Plates not found	0	0%

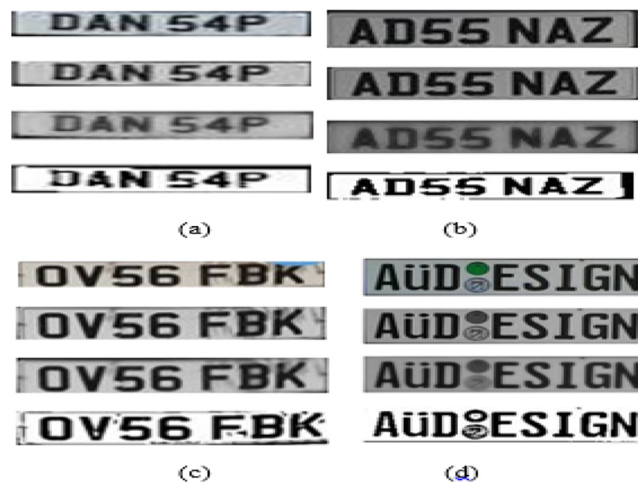

Fig. 7. GrabCut, gray-scale, noise removal and binarization of localized plates of vehicle images (a), (b), (c) and (d).

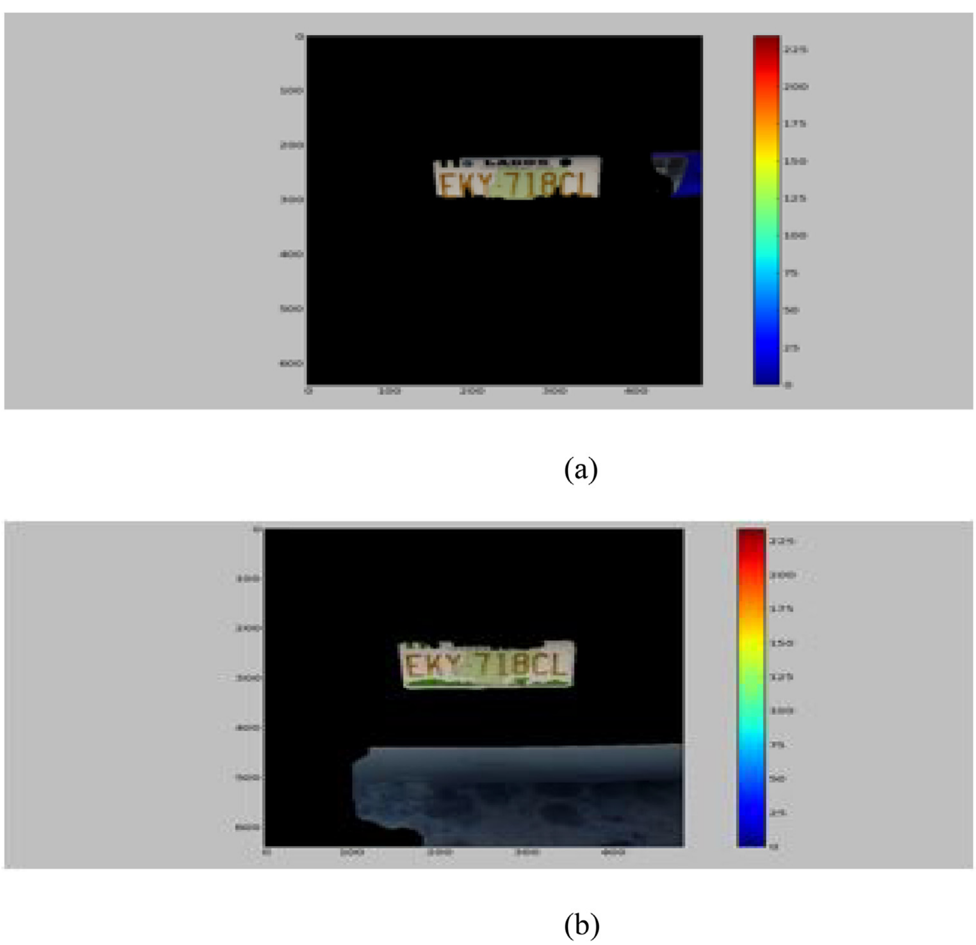

Fig. 6. (a) Inaccurate results obtained when α is varied above 0 but less than or equal to 0.5. (b) Inaccurate results obtained when α is varied above 0.5 but less than 1.

3.4. Conversion of RGB image into grayscale

A color image is a 24 bit RGB image. It is converted to an 8 bit grayscale image to reduce its complexity and for easy manipulation by a computer system. The RGB image is converted to a grayscale image using Eq. (13).

$$Gray\,Conversion = 0.299R + 0.587G + 0.114B \tag{13}$$

3.5. Noise removal and image binarization

A digital image contains noise as a result of the condition in which the image is captured or from the capturing device during transmission. In this work, we use the laplacian filter which serves as an edge detector and a filter to remove noise from the image. The output of the grayscale images is further converted into digital form by a computer system. Binarization is the process of converting an image to black and white. A threshold (t*) is set between 0 and 255. The image pixels that fall within 0–127 are labelled 0 (black) and those greater than 128 but less or equal to 255 are labelled 1 (white) as given by Eq. (14).

$$\mathrm{I_b}(\mathrm{x},\mathrm{y}) = \begin{cases} 0I\ (\mathrm{x},\mathrm{y}) \le t*, & 0 \\ 1I\ (\mathrm{x},\mathrm{y}) > t*, & 1 \end{cases} \tag{14}$$

where I is the image, Ib (X,Y) is the binarized image and t_* is the threshold. When $I \le 127$, the image is labelled 0 (0I) and when I < 128, the image is labelled 1 (1I).

3.6. Localization of license plate

The localization of LPs is performed using the designed algorithm. The algorithm is implemented on OpenCV with Python programming language. The Python edition of OpenCV used is JetBrains Pycharm Community Edition 2016.3.2.

Open computer vision is an open source platform and it is usually referred to as OpenCV. It contains tools for image processing and is expensively being used in various fields namely: computer vision, artificial intelligence, image processing, medicine and engineering. The algorithm for the various steps of vehicle plate recognition are written here and executed to perform localization. Localization is performed by minimizing the energy function as shown in Eq. (11), which when minimized gives the accurate location and extraction of the LP.

4. Results and discussion

4.1. Experimental results obtained for the LP localization process

In this section, we present the experimental results of the evaluation of the license plate localization process and in addition present a comparison of our method with existing methods. In the experiments, we evaluate our proposed method on an Intel core i5 PC with 2.6 GHz CPU, 8 GB RAM, and Windows 8 operating system. The image set includes 500 different images with a size of 480×640 each.

The experimental results of the modified GrabCut gave accurate localization of LPs for both Graph-cut localization and GrabCut localization as shown in Fig. 9. An accuracy of 99.8% was achieved for LP localization as presented in Table 2. This accuracy was achieved from calculations using Eq. (1).

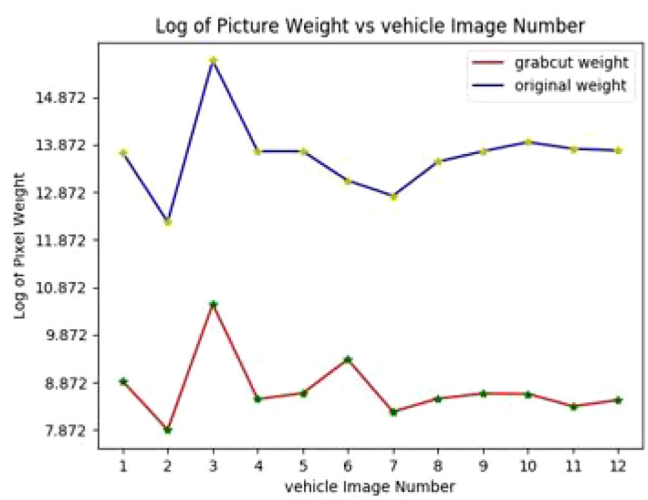

Fig. 8. Graph of original image weight and GrabCut image weight against vehicle image numbers (1–12).

Table 3
Results of processing time and aspect ratio of processed vehicle images.

S/N	Vehicle Image	Vehicle Number	Aspect Ratio (A_r)	Processing Time (s)
1	a	DAN54P	3.837	0.262
2	b	AD55NAZ	4.500	0.191
3	c	OV56FBK	4.896	0.104
4	d	AUDESIGN	2.027	0.278
5	e	KW3MXR	3.215	0.123
6	f	VJ88671	4.564	0.195
7	g	WBX360	3.470	0.275
8	h	RX61GDU	3.333	0.262
9	i	KA555ZG	4.314	0.202
10	j	BITCH852	4.500	0.275
11	k	MH01AV8866	3.634	0.122
12	l	ZG6803Z	3.634	0.226

Table 4
Results of the original image weight and the Grabcut weight of processed vehicle images.

S/N	Vehicle Image	Vehicle Number	Original Image weight	GrabCut Image weight	$\left(\frac{\text{Total Weight of CrabCut Image}}{\text{Total Weight of Original Image}}\right)$
1	a	DAN54P	900,240	7345	0.815
2	b	AD55NAZ	208,608	2622	1.256
3	c	OV56FBK	6,220,800	37,062	0.595
4	d	AUDESIGN	921,600	5016	0.544
5	e	KW3MXR	921,600	5698	0.618
6	f	VJ88671	498,000	11,520	2.313
7	g	WBX360	360,000	3840	1.067
8	h	RX61GDU	740,775	5070	0.684
9	i	KA555ZG	921,600	5655	0.613
10	j	BITCH852	1,121,280	5609	0.500
11	k	MH01AV8866	972,902	4306	0.442
12	l	ZG6803Z	937,890	4914	3.634

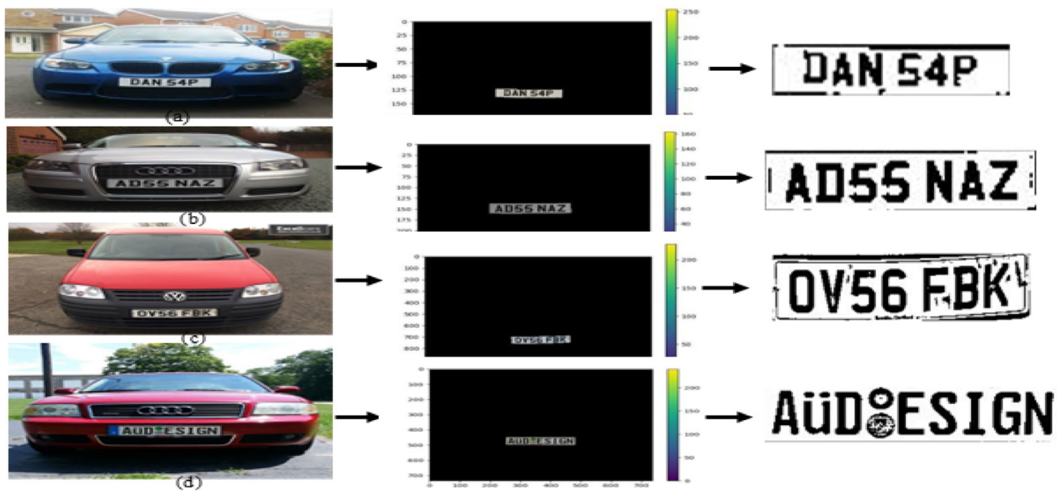

Fig. 9. Some of the vehicle images used for the test 1. Vehicle images (a), (b), (c) and (d) (Source: Computer Vision datasets: http://homepages.inf.ed.ac.uk/rbf/CVonline/Imagedbase.html). 2. Result of Graph-Cut images (a), (b), (c) and (d). 3. Result of localized vehicle images (a), (b), (c) and (d).

Table 5
Comparison of results of the different LPL algorithms and their localization accuracies.

Author	License plate localization technique	% Accuracy	Processing time (s)	System configuration
Anagnostopoulos et al. (2006)	Sliding concentric window technique	Not specified	0.27	Pentium IV with 3.0 GHz CPU and 512-MB RAM
Wang et al. (2017)	HSV colour space	75.8%	Not specified	Intel Core i7-3537 U with 2.50 GHz CPU
Parisi et al. (1998)	Neural networks	76.0%	Not specified	Pentium Pro PC 200 MHz processor
Faradji et al. (2007)	Morphological operations	84.3%	Not specified	Pentium 4 with 2.4 GHz CPU and 256 MB RAM
Owamoyo et al. (2013)	Sobel filter, morphological operations and CCA	85.0%	Not specified	Not specified
Azam and Islam (2016)	Statistical binarization and radon transform	86.2%	0.73	Intel Core 2 Duo CPU T6600 with 2.2 GHz processor and 2 GB RAM
Saha et al. (2009)	Horizontal and vertical edge detection	89.2%	Not specified	Intel Core 2 Duo, 2.2 GHz, S: 120X40
Shi et al. (2005)	Color image processing	90.0%	0.30	PIII 1G PC
Danbatta et al. (2016)	Blob detection	91.4%	15.00	Not specified
Roy and Ghoshal (2011)	Component labeling and region growing on LP	91.5%	Not specified	Not specified
Salahshoor et al. (2013)	Color	91.6%	2.30	Pentium 4 processor, 512 MB RAM
Caner et al. (2008)	Gabor filter and connected component labeling (CCL)	91.7%	0.50	Xilinx Virtex IV FPGA
Al-Ghaili et al. (2013)	Vertical edge detection algorithm (VEDA)	91.7%	Not specified	Core 2 CPUs with 1.83 GHz and 1 GB of memory
Mahmood et al. (2013)	Region-based filtering	92.0%	Not specified	Not specified
Lalimi et al. (2013)	Sobel operator and morphological filtering	92.0%	Not specified	2.26 GHz CPU
Hsieh et al. (2005)	Region-based filtering Wavelet based method	92.4%	Not specified	Intel Pentium 4–1.6 GHz CPU/256 M RAM
Duan et al. (2005)	Hough transform	92.8%	0.65	Pentium 4 X2000 with 1.4 GHz CPU and 512 MB RAM
Zhou et al. (2012)	Principal visual word and local feature matching	93.2%	0.22	PC with 4-G memory and 2.53-GHz CPU
Zhang et al. (2006a)	Global and local features detection	93.5%	Not specified	PC with Pentium 2.8 GHz CPU
Al-Hmouz, and Challa (2010)	Feature fusion and Bayes' rule	93.7%	0.82	Not specified
Yang et al. (2012)	Color edge extraction	94.7%	Not specified	Intel Pentium 4 with 2.4 GHz CPU and 1-GB RAM
Wang et al. (2008)	Color-FCRA	95.1%	0.43	TI DM642 600 MHz/32 MB RAM
Hemayat et al. (2014)	Dynamic thresholding	95.3%	Not specified	Intel Core CPU 2.0 GHz RAM
Gou et al. (2015)	Extremal regions (ERs) and Boltzmann machines	95.9%	0.40	PC with 3.1-GHz Intel Core 2 Quad CPU and 4-GB RAM
Rashid (2013)	Color	96.0%	0.26	Pentium 3 CPU 800 MHz
Wang et al. (2015)	SIFT feature	96.0%	0.26	Pentium 4 with 2.4 GHz CPU and 1 GB memory
Li and Xie (2007)	Color and texture approach	96.4%	3.00	AMD 1.75 GHz CPU and DDR400 512 MB
Zhang et al. (2006b)	Vertical edge map and Haar cascade	96.4%	Not specified	PC with Pentium 2.8 GHz CPU
Wang et al. (2010)	Horizontal scan of repeating contrast changes	96.8%	Not specified	Not specified
Zimic et al. (1997)	Fuzzy logic approach	97.0%	2.00	Workstation PC SG-INDIGO 2
Guo and Liu (2008)	Textures, color information and aspect ratio	97.1%	0.53	Intel Pentium 4, 3.2 GHz CPU and 512-MB RAM
Hung et al. (2007)	Haar wavelet transform	97.3%	Not specified	Intel Pentium 4–2.66 GHz CPU/512 MB RAM
Chen et al. (2009)	Feature salience	97.3%	0.22	Pentium 4 with 2.26-GHz CPU
Comelli et al. (1995)	Gradient analysis	Not specified	1.10	IBM RS/6000 running AIX (a version of UNIX)
Ter Brugge et al. (1998)	Color	97.6%	6.00	Not specified
Naito et al. (2000)	Color and texture	Not specified	1.00	Not specified

(continued on next page)

Table 5 (*continued*)

Author	License plate localization technique	% Accuracy	Processing time (s)	System configuration
Jia et al. (2007)	Mean shift region-based method	97.6%	Not specified	Not specified
Chang et al. (2004)	Fuzzy logic approach and neural networks	97.9%	0.40	Pentium 4 with 1.6 GHz CPU
Deb and Jo (2009)	Color and fuzzy aggregation	97.9%	0.40	Pentium-IV 2.4 GHz 1024 MB RAM
Hsieh et al. (2002)	Morphology-based method	98.0%	Not specified	Not specified
Saini and Saini (2017)	Multiwavelet Transform	98.3%	Not specified	Intel Core i7 with 3.60 GHz CPU
Liu et al. (2017)	Color edge algorithm	98.95%	0.24	PC with Intel Core i5-3210 M processor (2.5 GHz), 8 GB RAM
Cho et al. (2011)	Gray-level values	99.5%	Not specified	2.6-GHz Pentium 4 PC with 512 MB of memory
Song and Shi (2011)	Level set transform and Voronoi diagram	99.6%	Not specified	Thinkpad X61 PC with 2.1 GHz CPU, 0.97 GB RAM
Zheng et al. (2005)	Edge detection analysis	99.7%	1.20	Pentium 4 with 2.4 GHz CPU and 256 MB RAM PC
**	Proposed technique (Modified GrabCut)	99.8%	0.21	Intel Core i5 PC with 2.6 GHz CPU and 8 GB RAM

Experiments have been performed to show the results when the foreground and background pixel labelling (α) is varied between 0 and 1. The results presented in Fig. 6(a) show when α is varied above 0 but less than or equal to 0.5, inaccurate results are obtained. This is similar to Fig. 6(b) which shows that when α is varied above 0.5 but less than 1, also inaccurate results are obtained. In our proposed algorithm, we set the computer to assign 0 or 1 to α, to differentiate the background pixels from the foreground pixels respectively. When the values of α are set to 0 and 1, and A_r is set as $2 \leq A_r \leq 5$, the foreground will be accurately separated from the background as shown in Fig. 9.

4.2. Results obtained for gray-scale conversion, noise removal and binarization

Some samples of the selected RGB images which were tested with the designed algorithm are shown in Figs. 7 and 9. The results obtained for the gray-scale conversion, noise removal and image binarization of the localized LPs is shown in Fig. 7.

4.3. Foreground extraction

To reduce the computational time and the space occupancy of the localized images in the database, GrabCut extracts only the foreground region and save it in the database. This is shown in Fig. 7.

4.4. Calculation of the accuracy of license plate localization

The accuracy of the proposed algorithm is calculated using Eq. (1). The experimental results obtained for the LP localization accuracy test using the Eq. (1) is shown in Table 2. It is important to note that the accuracy of LP localization and how fast the LPs are localized are important for a VPN recognition system.

An average processing time of 0.21 s was achieved with an accuracy of 99.8%. The results of the accuracy of localized LPs and the processing times achieved are shown in Tables 2 and 3 respectively.

4.5. Variation in pixel weight of acquired image to that of the GrabCut image

GrabCut technique performs the cut when the cost function $|C|$ is minimized. This causes the weight of the edges of the resultant GrabCut image to be less than that of the original image as given by Eq. (11). The results of the reduction in vehicle image weight is presented in Table 4. A plot of the weight of the original image to that of the GrabCut image for vehicle numbers (1–12) is shown in Fig. 8. The weight on the vertical axis is converted to logarithm to enable the values obtained to be plotted together in a single graph as shown in Fig. 8. The pixel weight of the GrabCut image is less than 3% of the original image for all images tested. This is shown in Table 4. In addition, the experimental results obtained for the A_r and processing time are shown in Table 3. In Table 4, it is observed that there is a large variation between the original image weight and the obtained GrabCut weight. This shows that the cost of the cut was greatly minimized with the modified GrabCut algorithm. The comparison of our technique with other existing techniques is shown in Table 5.

5. Conclusion

In this paper, an efficient and robust algorithm for vehicle plate number localization has been presented. The proposed algorithm is not country specific and can be used to detect LPs of different sizes, orientations, complex backgrounds, varying illumination and weather conditions. The algorithm is extensively tested with 500 vehicle image samples from different countries. The experimental result shows that this algorithm has high accuracy, reaching 99.8% and a corresponding fast processing time of 0.21 s. In future works, this approach will be further extended to localize license plate of motorcycles.

References

Abolghasemi, V., Ahmadyfard, A., 2007. A fast algorithm for license plate detection. Visual, LNCS 4781, 468–477.

Al-Ghaili, A.M., Mashohor, S., Ramli, A., Ismail, A., 2013. Vertical-edge-based car-license-plate detection method. IEEE Trans. Veh. Technol. 62 (1), 26–38.

Al-Hmouz, R., Aboura, K., 2014. License plate localization using a statistical analysis of discrete fourier transform signal. Comput. Electr. Eng. 40, 982–992.

Al-Hmouz, R., Challa, S., 2010. License plate localization based on a probabilistic model. Mach. Vis. Appl. 21, 319–330.

Anagnostopoulos, C.N., Anagnostopoulos, I., Kayafas, E., Loumos, V.A., 2006. License plate recognition algorithm for intelligent transportation system applications. IEEE Intell. Transp. Syst. 17, 377–392.

Angel, G., Jeny, R.R., Simon, J.A., 2014. Automatic license plate recognition using image enhancement technique with hidden markov model. Int. J. Sci. Eng. Res. 5 (5), 602–605.

Ashtari, A., Nordin, M., Fathy, M., 2013. An Iranian license plate recognition system based on color features. IEEE Trans. Intell. Transp. Syst. 12 (3), 830–845.

Azam, S., Islam, M.M., 2016. Automatic license plate detection in hazardous condition. J. Vis. Commun. Image Represent. 36, 172–186.

Badr, A., Abdelwahab, M.M., Thabet, A.M., Abdelsadek, A.M., 2011. Automatic number plate recognition system. Ann. Univ. Craiova, Math. Comput. Sci. Ser. 38 (1), 62–71.

Boykov, Y., Jolly, M.P., 2001. Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images. International Conference on Computer Vision. 1, 105–112.

Caner, H., Gecim, H.S., Alkar, A.Z., 2008. Efficient embedded neural-network-based license plate recognition system. IEEE Trans. Veh. Technol. 57 (5), 2675–2683.

Chang, S., Chen, L., Chung, Y., 2004. Automatic license plate recognition. IEEE Trans. Intell. Transp. Syst. 1, 42–53.

Chen, Z.X., Liu, C.Y., Chang, F.L., Wang, G.Y., 2009. Automatic license-plate location and recognition based on feature salience. IEEE Trans. Veh. Technol. 58 (7), 3781–3785.

Cho, B.K., Ryu, S.H., Shin, D.R., Jung, J.I., 2011. License plate extraction method for identification of vehicle violations at a railway level crossing. Int. J. Automot. Technol. 12 (2), 281–289.

Comelli, P., Ferragina, P., Granieri, M.N., Stabile, F., 1995. Optical recognition of motor vehicle license plates. IEEE Trans. Veh. Technol. 44 (4), 790–799.

Danbatta, S.J., Mohammed, M.D., Mohammed, N.D., Abdullahi, A.S., 2016. Segmentation and recognition of Nigerian license plate using bounding box and template matching. Dutse J. Pure Appl. Sci. 2 (2), 239–244.
Deb, K., Jo, K.H., 2009. A vehicle license plate detection method for intelligent transportation system applications. Cybern. Syst. 40 (8), 689–705.
Deshpande, S., Kamat, S., Patil, V., Patil, M., Patil, P., 2015. Use of horizontal and vertical edge processing technique to improve number plate detection. Int. J. Res. Eng. Technol. 4 (12), 286–292.
Duan, T.D., Hong Du, T.L., Phuoc, T.V., Hoang, N.V., 2005. Building an automatic vehicle license-plate recognition system. Proceedings of International Conference in Computer Science, Can Tho, Vietnam. 21–24.
Duc, D.T., Hong, T.L., Vinh, P.T., Viet, H.N., 2005. Building an automatic vehicle license plate recognition system. Proceedings of the International Confererence on Computer Science, Can Tho. 59–63.
Faradji, F., Rezaie, A.H., Ziaratban, M., 2007. A morphological-based license plate location. Proceedings of IEEE International Conference on Image Processing. 1, I–57-I–60.
Fu, Y.P., Li, Z.N., Yuan, D., 2004. Improving license plate recognition system by color algorithms based on HIS space. J. Comput. Eng. Des. 25 (5), 703–707.
Ghahnavieh, A.E., Amirkhani-Shahraki, A., Raie, A.A., 2014. Enhancing the license plate character recognition methods by means of SVM. 22nd Iranian Conference on Electrical Engineering. 220–225.
Gou, C., Wang, K., Yao, Y., Li, Z., 2015. Vehicle license plate recognition based on extremal regions and restricted boltzmann machines. IEEE Trans. Intell. Transp. Syst., 1–12
Guo, J.M., Liu, Y.F., 2008. License plate localization and character segmentation with feedback self-learning and hybrid binarization techniques. IEEE Trans. Veh. Technol. 57 (3), 1417–1424.
Halin, A.A., Sharef, N., HazriJantan, A., Abdullah, L., 2013. License plate localization using a Naive Bayes Classifier. IEEE International Conference on Signal and Image Processing Applications, Melaka, Malaysia. 20–24.
Hemayat, S., Saberkari, A., Baharlou, S.M., 2014. A fast and adaptive license plate localization algorithm with pattern-checking capabilities. Proceedings of the 7th International Symposium on Telecommunications, Tehran. 100–105.
Hongliang, B., Changping, L., 2004. A hybrid license plate extraction method based on edge statistics and morphology. Proc. Pattern Recognit. 2, 831–834.
Hsieh, J.W., Yu, S.H., Chen, Y.S., 2002. Morphology-based license plate detection from complex scenes. Proceedings of the 16th International conference on pattern recognition. 3, 176–179.
Hsieh, C.T., Juan, Y.S., Hung, K.M., 2005. Multiple license plate detection for complex background. Proceedings of IEEE 19th International Conference, Aina. 389–392.
Hung, K.M., Chuang, H.L., Hsieh, C.T., 2007. License plate detection based on expanded Haar wavelet transform. Proceedings of 4th IEEE International Conference on Fuzzy Systems and Knowledge Discovery, Haikou. 415–419.
Jahangiri, M., Heesch, D., 2009. Modified GrabCut for unsupervised object segmentation. Proceedings of the 16th IEEE International Conference on Image Processing. 2389–2392.
Jia, W.J., Zhang, H.F., He, X.J., Piccardi, M., 2005. Mean shift for accurate license plate localization. Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems. 566–571.
Jia, W., Zhang, H., He, X., 2007. Region-based license plate detection. J. Network Comput. Appl. 30, 1324–1333.
Kang, D.J., 2009. Dynamic programming-based method for extraction of license plate numbers of speeding vehicles on the Highway. Int. J. Automot. Technol. 10 (2), 205–210.
Khattab, D., Ebeid, H.M., Tolba, F.M., Hussein, A.S., 2016. Clustering-based image segmentation using automatic GrabCut. Proceedings of the 10th International Conference on Informatics and Systems. 95–100.
Kim, M.K., 2010. License plate location method unaffected by variation in size and aspect ratio. Int. J. Automot. Technol. 11 (5), 751–758.
Lalimi, M.A., Ghofrani, S., McLernon, D., 2013. A vehicle license plate detection method using region and edge based methods. Comput. Electr. Eng. 39, 834–845.
Li, J., Xie, M., 2007. A color and texture feature based approach to license plate location. In: Proceedings of the International Conference on Computational Intelligence and Security, Harbin. 376–380. https://doi.org/10.1109/CIS.2007.71.
Liu, Y., Huang, H., Cao, J., Huang, T., 2017. Convolutional neural networks-based intelligent recognition of Chinese license plates. Soft Comput. 1–17. https://doi.org/10.1007/s00500-017-2503-0.
Lv, X., Wang, M., Wang, G., Peng, G., 2003. Study on license plate segmentation. Comput. Eng. Appl. 15, 226–229.
Mahmood, A.L., Sedigheh, G., Desmond, C.M., 2013. A vehicle license plate detection method using region and edge based methods. Comput. Electr. Eng. 39 (3), 834–845.
Mobarhan, M.A., Shahbahrami, A., Parva, S., Asadi, M.N., Khajekini, A.A., 2012. A license plate detection algorithm using edge features. Adv. Comput Sci., Eng. Appl. 166, 413–421.
Musoromy, Z., Soodamani, R., Nico, B., 2010. Edge detection comparison for license plate detection. Intelligent Control, Automation, Robotics and Vision. 1–6.
Naito, T., Tsukada, T., Yamada, K., Kozuka, K., Yamamoto, S., 2000. Robust license plate recognition method for passing vehicles under outside environment. IEEE Trans. Veh. Technol. 49 (6), 2309–2319.
Orchard, M.T., Bouman, C.A., 1991. Color quantization of images. IEEE Trans. Signal Process. 39 (12), 2677–2690.
Owamoyo, N., Fadele, A.A., Abudu, A., 2013. Number plate recognition for Nigerian vehicles. Acad. Res. Int. 4 (3), 48–55.
Parisi, R.G., Di-Claudio, E.D., Orlandi, G., 1998. Car plate recognition by neural networks and image processing. Proceedings of IEEE International Symposium on Circuits and Systems. 3, 195–198.
Rashid, A.E., 2013. A fast algorithm for license plate detection. Proceedings of the IEEE International Conference on Signal Processing, Image Processing and Pattern Recognition, Coimbatore, India. 44–48.
Rother, C., Kolmogorov, V., Blake, A., 2004. GrabCut interactive foreground extraction using iterated Graph Cuts. ACM Trans. Graphics, 309–314.
Roy, A., Ghoshal, D.P., 2011. Number plate recognition for use in different countries using an improved segmentation. IEEE, 1–3.
Saha, S., Basu, S., Nasipuri, M., Basu, D.K., 2009. A novel scheme for binarization of vehicle images using hierarchical histogram equalization technique. International Conference on Computer, Communication, Control and Information Technology. 270–274.
Saini, M.K., Saini, S., 2017. Multiwavelet transform based number plate detection. J. Vis. Commun. Image Represent. 44 (C), 128–138.
Salahshoor, M., Broumandnia, A., Rastgarpour, M., 2013. An intelligent and real-time system for plate recognition under complicated conditions. Proceedings of the 8th Iranian Conference on Machine Vision and Image Processing, Zanjan. 245–250.
Salau, A.O., 2018. Development of a Vehicle Plate Number Localization Technique using Computer Vision PhD Thesis. Obafemi Awolowo University, Ile-Ife, Nigeria, p. 200.
Shi, X., Zhao, W., Shen, Y., 2005. Automatic License Plate Recognition System based on Color Image Processing. LNCS Springer-Verlag, Berlin Heidelberg, pp. 1159–1168.
Song, Y., Shi, L., 2011. License plate location by level set transform and voronoi diagram. In: Proceedings of the 4th International Conference on Intelligent Networks and Intelligent Systems. 37–40. https://doi.org/10.1109/ICINIS.2011.25.
Tadic, V., Popovic, M., Odry, P., 2016. Fuzzified Gabor filter for license plate detection. Eng. Appl. Artif. Intell. 48, 40–58.
Tarabek, P., 2012. A real-time license plate localization method based on vertical edge analysis. Proceedings of the IEEE Federated Conference on Computer Science and Information Systems. 149–154.
Ter Brugge, M.H., Stevens, J.H., Nijhuis, J.A.G., Spaanenburg, L., 1998. License plate recognition using DTCNNs. Proceedings of the 5th IEEE International Workshop on Cellular Neural Networks and their Applications, London. 212–217.
Ullah, I., Lee, H.J., 2016. An approach of locating Korean vehicle license plate based on mathematical morphology and geometrical features. Proceedings of the International Conference on Computational Science and Computational Intelligence. 836–840.
Wang, J., Bacic, B., Yan, W.Q., 2017. An effective method for plate number recognition. Multimedia Tools Appl. 1–14. https://doi.org/10.1007/s11042-017-4356-z.
Wang, Y., Lin, W., Horng, S., 2011. A sliding window technique for efficient license plate localization based on discrete wavelet transform. Expert Syst. Appl. 38, 3142–3146.
Wang, Y., Chen, J., Hu, B., Yang, X., Ban, X., 2015. License plate recognition based on SIFT feature. Optik Int. J. Light Electron Opt., 1–24
Wang, M., Liu, Y., Liao, B., Lin, Y., Horng, M., 2010. A vehicle license plate recognition system based on spatial/frequency domain filtering and neural networks. Comput. Collective Intell. Technol. Appl. 6423, 63–70.
Wang, F., Man, L.C., Wang, B.P., Xiao, Y.J., Pan, W., Lu, X.C., 2008. Fuzzy-based algorithm for color recognition of license plates. Pattern Recognit. Lett. 29, 1007–1020.
Wen, Y., Lu, Y., Zhou, Y.J., von Deneen, K.M., Shi, P., 2013. An algorithm for license plate recognition applied to intelligent transportation system. IEEE Trans. Intell. Transp. Syst. 12 (3), 830–845.
Yanamura, Y., Nishiyama, D., Soga, M., Nakatani, H., Saji, H., 2003. Extraction and tracking of the license plate using Hough transform and voted block matching. Proceedings of IEEE intelligent vehicles symposium. 243–246.
Yang, X., Hao, X., Zhao, G., 2012. License plate location based on trichromatic imaging and color-discrete characteristic. Int. J. Light Electron Opt. 123, 1486–1491.
Youssef, S.M., Abaelrahaman, S.B., 2008. A smart access control using an efficient license plate location and recognition approach. Expert Syst. Appl. 34, 256–265.
Zhang, H., Jia, W., He, X., Wu, Q., 2006a. Learning based license plate detection using global and local features. Proceedings of the International Conference on Pattern Recognition, Hong Kong, China. 1102−1105.
Zhang, H., Jia, W., He, X., Wu, Q., 2006b. A fast algorithm for license plate detection in various conditions. Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, Taipei, Taiwan. 2420–2425.
Zheng, D., Zhao, Y., Wang, J., 2005. An efficient method of license plate location. Pattern Recognit. Lett., 2431–2438
Zhou, X., Ito, T., Zhou, X., Chen, H., Hara, T., Yokoyama, R., Kanematsu, M., Hoshi, H., Fujita, H., 2014. Universal approach for automatic organ segmentations on 3D CT images based on organ localization and 3D Grabcut. Proc. SPIE 9035, 1605–7422.
Zhou, W., Li, H., Yijuan, L., Qi, T., 2012. Principal visual word discovery for automatic license plate detection. IEEE Trans. Image Process. 21 (9), 4269–4279.
Zimic, N., Ficzko, J., Mraz, M., Virant, J., 1997. The fuzzy logic approach to the car number plate locating problem. Proceedings on the Intelligent Information Systems, Grand Bahama Island. 227–230.