IET Image Processing

Research Article

Locating multiple license plates using scale, rotation, and colour-independent clustering and filtering techniques

ISSN 1751-9659
Received on 21st September 2018
Revised 2nd July 2019
Accepted on 29th July 2019
E-First on 17th September 2019
doi: 10.1049/iet-ipr.2018.6237
www.ietdl.org

Rohit Bhargav[1] ✉, *Parag Deshpande*[1]
[1]*Department of Computer Science & Engineering, Visvesvaraya National Institute of Technology (VNIT), Nagpur, India*
✉ *E-mail: rohit.bhargav@students.vnit.ac.in*

Abstract: A license plate (LP) can help identify a motor vehicle. However, no common standard for LP exists across countries, and even within a country, significant LP variations are observed. In addition, environmental factors cause uncontrolled plate and character variations. It is, therefore, a challenge to design a robust and universal license plate recognition (LPR) system which works for multiple countries, for different types of vehicles, and for different styles of LPs. This study presents a novel approach for locating LP based on the use of multiple clustering and filtering techniques applied to the geometrical properties of LP characters. The proposed approach is independent of the size, rotation, and colour of the LP and can be used to locate single or multiple LP of different styles of different vehicles and of different countries. The approach has been validated using the standard Media-lab and application-oriented license plate (AOLP) datasets as well as on datasets of vehicles from other countries. The approach achieved an average success ratio of 93.42% for locating LPs from both the Media-lab and the AOLP dataset and is higher than the results of previously published methods which evaluated their performance over the same datasets.

1 Introduction

The ability to detect and identify a vehicle is a key requirement for any intelligent transport system (ITS). Once a vehicle is detected and identified, multiple actions can be performed by an ITS like counting the number of vehicles that have passed by, controlling access to areas, monitoring traffic etc. As vehicular traffic increases, there is a need for faster and more accurate identification of vehicles that an ITS must handle.

Across most countries, a license plate (LP) is unique to a motor vehicle and so license plate recognition (LPR) can be one way to identify a vehicle. Once the LP is accurately located and identified, additional information about the vehicle can be obtained, which can then be used for further processing by an ITS. However, there is no common standard for LPs across countries and even within countries and regions under common administrative control, significant license plate variations exist in the form of sizes, colours, fonts, and locations of the plate. In addition, environmental factors like light, dust, pollution and corrosion etc. cause additional uncontrolled plate and character variations. All these make it a challenge to design a robust and universal LPR system based on visual recognition.

Over the past few decades, a lot of research and development has gone into visual recognition of LPs and in many cases, superior performance is reported for such visual LPR systems. However, most of these evaluations have been performed in a controlled environment and with images from proprietary datasets. The absence of a common publicly available reference library makes it difficult to perform an accurate assessment of such LPR systems. In addition, the methods proposed have been depended on isolating and identifying the edge information or doing morphological operations, or by template matching, or by colour information of the LP. These methods tend to fail if any of the non-LP portions of a vehicle also matches the properties being isolated in these methods for LP detection.

In this paper, we propose a method for locating the LP from a source image using various clustering and filtering techniques applied to geometrical properties of the LP characters. The geometrical properties used are size, rotation, and colour invariant and therefore, unlike other published LPR methods, this method is not affected by non-LP elements in the rest of the source image.

The remaining sections of this paper are planned as follows. Section 2 lists existing research around detecting, locating and identifying LPs. Section 3 provides an overview of the proposed approach for locating the LP. Section 4 details each of the steps of the proposed method to locate the LP. Experimental results are discussed in Section 5 and conclusions are summarised in Section 6.

2 Literature review

As vehicle detection and identification is a key requirement for an ITS, much research has been carried out for LPR systems over the last decades. However, there exist considerable variations in LPs and thus LPR continues to remain a challenging area even today. In the published literature, there is no universal LPR method that works across many countries or across different vehicles types or for different styles of LPs. Further, most of the existing methods in the published literature depend on edge information, morphological operations, template matching, colour information of the LP, or on some variations of these. More recently, convolutional neural networks (CNN) have also been used for the identification of a vehicle LP.

Lee *et al.* [1] proposed a colour image processing (CIP) method to extract the LP based on the LP background and the LP character's colour using the colour histogram. Yang *et al.* [2] proposed a method based on fixed colour collocation (FCC) to locate the LP. Yuan *et al.* [3] proposed a method that uses a line density filter to extract candidate regions and then uses a cascaded classifier based on support vector machine (SVM) using colour saliency to identify possible LP regions. Colour-based LP extraction methods identify the LP using the background and sometimes foreground colour of the LP. These categories of LP detection methods fail to extract the LP from an image when any of the features they adopted to extract the LP also exists in other parts of the source image.

Hongliang and Changping [4] proposed a method based on edge statistics and morphology (ESM). Faradji *et al.* [5] proposed a real-time and robust (RTR) multi-stage method to find the LP location using a combination of Sobel mask, histogram analysis, and morphological operations. Huang *et al.* [6] proposed a method that uses a search window, a morphology-based dilation operation and horizontal and vertical projections to detect the LP region. Al-Ghaili *et al.* [7] proposed a method that works on greyscale images and aims to identify the LP using vertical edge detection. Hsu *et al.* [8] proposed an approach primarily designed for Taiwan and based on the expectation-maximisation clustering method using vertical edges of greyscale images. Yepez and Ko [9] used morphological operations for LP localisation. LPR methods that depend on edge information and morphological operations tend to look for areas with specific properties. Morphology-based approaches have to define a structuring element (SE) to perform morphological operations. Defining a particular SE to detect a probable LP is a non-generic approach and some LPs are not so easy to detect using edge information. Hence these proposed methods fail to detect the LP from the input images under various characteristics of LPs in the images.

Wen *et al.* [10] proposed a method to identify the LP but it is based on prior knowledge of the LP region. Zhou *et al.* [11] proposed an approach based on the principal visual word (PVW) discovery and visual word matching which compares extracted features with pre-existing LPs and locates the LP using matching. Mingdong [12] proposed a method focused on Chinese characters which used local binary pattern combined with horizontal and vertical projection algorithm for feature extraction, followed by backpropagation (BP) neural network (NN) to identify the LP characters. Haneda and Hanaizumi [13] proposed a method that performs a global search for the probable LP using multiple templates, a 3D cross-correlation function, and principal component expansion. The method uses corner detection to remove deformations of the LPs and uses spatial similarity with an LP template to detect the LP. Methods that depend on template matching require accurate templates to be used. The drawback of these proposed methods is that they depend on prior knowledge of the LP or the LP character which are used as templates and against which matching is performed, so these methods tend to fail if the LP region is not known in advance.

Samra and Khalefah [14] proposed a method using dynamic image processing techniques and genetic algorithms (DIP-GA) for detecting a single LP in the source. In a subsequent paper. Samra [15] proposed an improvement on the earlier method in [14] to handle rotated and multiple LPs by using a semi-hybrid-GA (SHGA) system for LP localisation. Khan *et al.* [16] proposed an approach that uses an entropy-based method for feature selection and SVM for classification. In the proposed method, segmentation is performed by identifying the luminance channel and then doing binary segmentation of that channel using Otsu thresholding. The drawback of these proposed methods is that they reported results considering only a subset of around 335 images out of the 741 images from the Media-lab benchmark LP dataset with the rest of the samples being from a proprietary dataset.

Kim *et al.* [17] proposed a method that uses region based CNN for vehicle region detections and then uses hierarchical sampling and deep convolutional neural network to eliminate non-LP regions. Xie *et al.* [18] used a CNN-based MD-YOLO framework for multi-directional car license plate detection. The method used rotation angle prediction and a fast intersection-over-union evaluation strategy to overcome rotational problems in captured images. Li *et al.* [19] also used a CNN to simultaneously localise license plates and recognise the letters in a single forward pass through a single network. Methods relying on CNN (or any other method using machine learning) require very a large dataset to be accurate and so when used with a smaller sized publicly available dataset, these methods have to expand the dataset with additional images created by performing operations on the images of the original dataset to increase accuracy. This means the methods are tested on a different dataset than the original public dataset and accuracy is dependent on the quality of the expanded dataset used for training the methods.

Anagnostopoulos *et al.* [20] proposed a new image segmentation method called sliding concentric windows (SCW) for LP detection. The method uses statistics such as the standard deviations and the mean values in local window regions for detecting the LP location. The statistical measurement threshold values used by the method are set by the user after a trial-and-error procedure which is not a generic solution in the real-world context as the LPs can have various deformations such as tilt, rotation, and pan from various viewpoints. Salazar *et al.* [21] used agglomerative clustering for grouping images based on their content similarities and segmentation of regions of an image in similar areas. The method while not specific to LP detection classified similar objects together using a clustering based approach. Soora and Deshpande [22] proposed a method based on a clustering technique applied to geometrical properties of the connected components in an image. The method used a thinning and resizing operation to differentiate between the LP and non-LP components and then detects the LP. However, the thinning and resizing operation as described fails to sufficiently differentiate between the LP and non-LP components and so the method has low accuracy of locating the LP.

In this paper, we are proposing a method, which applies clustering techniques on geometrical properties of the LP characters to locate the LP. As this method is based on clustering of geometrical properties, it is independent of scale, rotation, colour, tilt, and the orientation of the LP and thus can be used to locate LPs from different countries, LPs of different vehicles, multiple LPs, and LPs in different styles. The proposed method also significantly reduces the number of components that are retained and which then need to be tested for LP character match. There are very few techniques in the published literature which talk about LP detection methods under various environmental conditions and plate variations and which will work for different types of vehicle and for LPs in different styles.

3 Overview of proposed approach for locating LP using clustering techniques

Our proposed method uses clustering and filtering techniques based on the geometrical properties of the LP characters to locate the LP. As this method is based on clustering of geometrical properties, it is independent of scale, rotation, colour, tilt, and the orientation of the LP. We do not use any type of edge detection, template matching, morphological operations, or colour information of the LPs, which have been used by earlier works around LPR. The proposed method is also independent of plate variations, and environmental conditions. The plate variations include the location of the plate on the vehicle, multiple plates in a single image, different combination of vehicles with different plate orientations, different sizes of plates, different background colours of plates, plates with dirt, rotated plates, plates having two lines of characters with each line of characters of different sizes, tilted LPs etc. The environmental conditions include different illuminations, background conditions and weather conditions. The proposed method is therefore a generalised method for identifying the LP that can be used for multiple countries, for different types of vehicles and is independent of plate variations under different environmental and weather conditions.

To locate the LP we use the knowledge, that in general, the LP characters are adjacent to each other, they are positioned typically at the same horizontal level along one or multiple lines, they are of similar height and of similar width, and have a similar thickness. LP characters also tend to have similar space in between the characters and have other local similarities. Most of these properties are followed by many countries while designing their LPs and so the approach can be used to identify and locate LP for many countries.

The proposed approach uses the following steps to first create a group of clusters which may contain the LP.

- Apply pre-processing to the input source. This may include converting a video stream to a frame which is then used as the input image.
- After pre-processing is completed, connected component analysis (CCA) is used to identify and label components of the images and extract geometrical properties of each component like the location of the top-left corner, height, width, area, number of perimeters, length of the perimeter, the ratio of area to perimeter etc.

- After the connected components are extracted, we cluster them into a distance, line, and height-based groups.

The resultant clustered components have properties such as, being near to each other, positioned along a line, and of similar height but these clusters can contain non-LP components. The proposed approach next uses the following filtering operations as described below to remove the non-LP components:

- Each component in the remaining clusters is examined and a thinning operation is performed on the component. The thinning operation reduces the component to a single pixel width. LP characters tend to thin to line like segments while non-LP characters tend to retain their shapes. We utilise this to eliminate non-LP components
- Each component in the remaining clusters is examined and the ratio of its area to the square of its perimeter length is calculated. LP characters tend to have a larger perimeter length enclosing a smaller area. We utilise this to eliminate non-LP components
- Each component in the remaining clusters is examined and the thickness of the character stroke is determined. LP characters tend to be of a similar character stroke thickness. We utilise this to eliminate non-LP components
- Each component in the remaining clusters is examined and the end-to-end width of the components of the cluster is determined. LP characters tend to be of a similar end-to-end width. We utilise this to eliminate non-LP components
- Each component in the remaining clusters is examined, and the height of the component is determined. LP characters tend to be of a similar height. We utilise this to eliminate non-LP components.
- After each of the operations above, the size of the cluster which is the number of components in the cluster is checked and if the cluster size is less than a specified minimum number of components, the cluster is removed.
- Finally, we check for the presence of a boundary region to differentiate between the cluster of LP characters and other clusters which may have text like characters as components.

Thus based on the geometrical properties of LP characters, we identify connected components in the source images and cluster these components as per the above properties to locate the LP. The next section explains each of the steps in the proposed method in detail.

4 Detailed description of the proposed approach for locating LP

This section describes in detail each of the steps performed as part of the proposed approach for locating the LP in the source image.

4.1 Pre-processing stage

The starting point for the proposed approach is an input image. If the input is a video, it is converted into frames and then individual frames are considered as the input image. Next, the following pre-processing steps are performed:-

- The input image is first converted to a greyscale image and then to a binary image using thresholding. A source image may have uneven illumination and shadows present, both of which can create difficulties during binarisation. Bernsen's algorithm with configurable window size is therefore used to overcome uneven illumination and shadows and the greyscale image is converted to a binary image. Sometimes, the source image can contain dark shadows on the LP. Very dark shadows across a LP can cause incorrect segmentation in binarised image even if Bernsen's algorithm is used. We manually identified such images with dark shadows in the dataset, performed shadow removal as a separate pre-processing step on these images and then these shadow free images were taken up for binarisation.
- CCA is then performed on the binary image and connected components are extracted. The components are internally stored as a binary matrix with '1' representing a foreground (lit) pixel and '0' representing a background (unlit pixel). As components are extracted, we also determine their geometrical properties like left-top coordinates, width, height etc.
- We ignore those components whose pixel height and pixel width are less than a configurable minimum number of pixels count, e.g. three (3), as LP character with height and pixel width less than this number, would be unrecognisable

So, at end of the pre-processing stage, we have a collection of components with their geometrical properties like left-top coordinates, width, height etc. determined. Fig. 1 illustrates the original source and the resultant image after binarisation. Fig. 2 illustrates the results after small components are removed from the binarised image in the pre-processing stage.

4.2 Clustering stage

The clustering stage is used to group the probable LP components from the components remaining at the end of the pre-processing stage. In this stage, three types of clustering operation are performed on the remaining components one after the other. The first clustering is based on the Euclidean distance between the components and it groups together components which are near to each other. Next, line-based clustering is used to divide each distance-based cluster into groups that are in a line. After this, height-based clustering is performed so that components of similar height are retained in the clusters. The steps used for clustering the components are similar to the clustering steps described by Soora and Deshpande [22]. After these distance, line, and height-based clustering steps are completed, the resultant clustered components have the properties such as, being near to each other, being positioned along a line, and being of similar height. These properties are exhibited by the characters comprising the LP of many countries.

Fig. 3 illustrates distance-based clustering in which the components C1 to C13 are examined and then clustered into two distance-based clusters C1–C9 and C10–C13. The adjacent image shows the binarised image after distance-based clustering is completed on the components. Fig. 4 illustrates line-based clustering on a distance-based cluster having components C1–C9, via which two line-based clusters, C1–C6 and C7–C9 are created, and a line cluster of [C10, C12, C13] is created from the distance-based cluster of C10–C13 by removal of C11.

The adjacent image shows the binarised image after line-based clustering is completed on the components. Fig. 5 illustrates height-based clustering applied on a line-based cluster of C1 to C6, and how C3 is removed as it has a height much different from the other components of the cluster. The adjacent image shows the

Fig. 1 *Source and binarised image*

Fig. 2 *Image after removing small components*

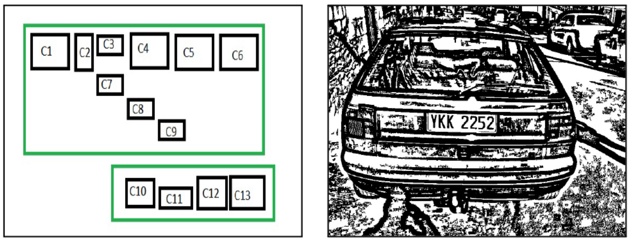

Fig. 3 *Distance-based clustering*

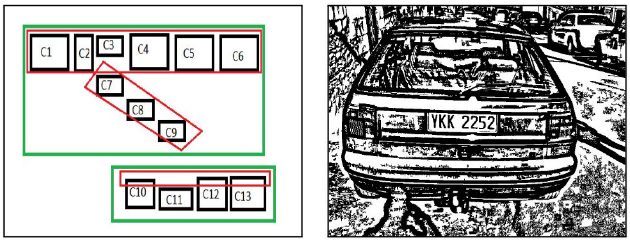

Fig. 4 *Line-based clustering*

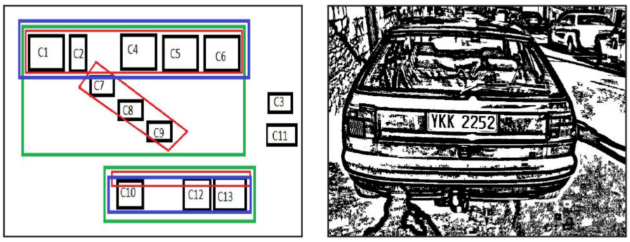

Fig. 5 *Height-based clustering*

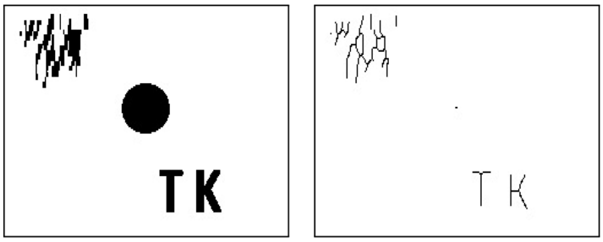

Fig. 6 *Thinning operation on regular and irregular-shaped components*

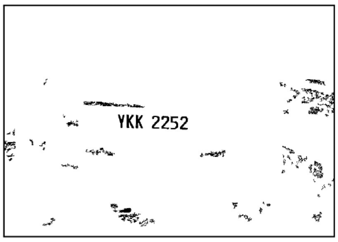

Fig. 7 *Binarised image after thinning operation*

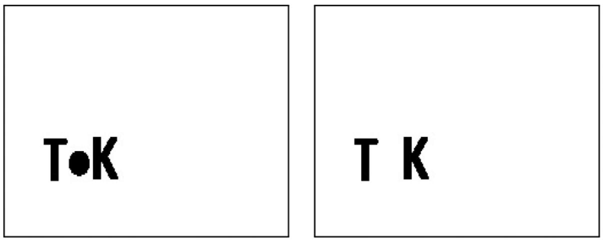

Fig. 8 *Area/perimeter^2 filtering operation*

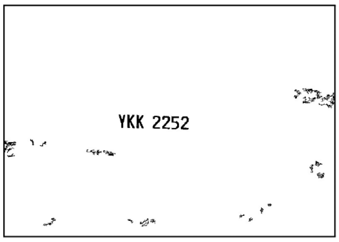

Fig. 9 *Binarised image after area/perimeter^2 filtering operation*

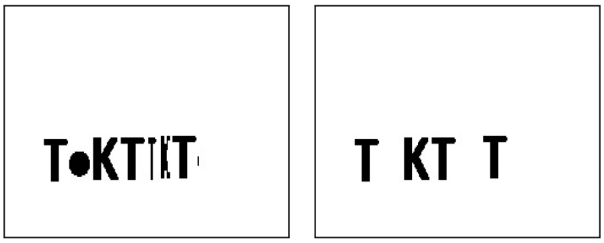

Fig. 10 *Scan-width filtering operation*

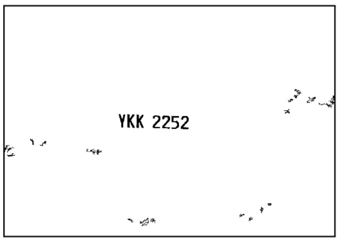

Fig. 11 *Resultant image after scan-width filtering operation*

binarised image after height-based clustering is completed on the components.

After the clustering steps are completed, the components retained in each remaining clusters satisfy all the characteristics based on distance, line, and height-based clustering. However, as seen in Fig. 5, the resultant image still contains a lot of non-LP components. These non-LP components are next removed through additional filtering operations described next.

4.3 Filtering stage

The filtering stage is used to remove components which exhibit all the characteristics checked in distance, line, and height-based clustering but are not part of the LP. The filtering stage applies multiple filters one after the other to remove components in the remaining clusters which do not meet the filtering criteria. The filters are based on the similarity of geometrical properties of the LP characters to each other. The filtering steps used are detailed below:

- Each component in the remaining clusters is examined and a thinning operation is performed on the component. Thinning results in the skeletonisation of the binary image. LP characters being of regular shape tend to thin to a single line like segments, while non-LP components being of irregular shape tend to almost retain their shapes. We compare the component before and after the thinning operation by checking for matching ‘1’ in their binary matrix representation and remove those components which do not show much change after the thinning operation. The thinning operation, however, retains components which are of regular shape but are not LP components. Fig. 6 shows the

- result of the thinning operation on some example components in a cluster. Fig. 7 shows the binarised image after thinning operation.
- Each component in the remaining clusters is again examined and its filled area (i.e. number of '1's in the binary matrix representation) and the total length of the external and internal perimeters are calculated. LP characters being of regular shape tend to have a longer perimeter length enclosing a smaller filled area while non-LP characters of regular shape (e.g. spots or globular patterns) will have a smaller perimeter enclosing a larger filled area. For each component, we compare the ratio of its area and the square of its total perimeter length and remove those components where the ratio is high; thus retaining those components that have a longer perimeter length enclosing a smaller filled area. Fig. 8 illustrates the area/perimeter^2 filter on some example components in a cluster. Fig. 9 shows the binarised image after the filtering operation.
- Each component in the remaining clusters is again examined and the scan width of the component is determined. The scan width of the component is calculated by performing a row-by-row line scan of the component to determine the width of a continuous segment of a row. Next, we discard the larger and smaller values to remove widths of the wider and the thinner portions of the component; thus retaining width of the portions which are of similar scan width. The scan width approximately matches the character stroke thickness. As LP characters tend to have a similar thickness of character stroke, we eliminate those components whose scan width differs much from that of the other components in the cluster. In this way, we retain those components in a cluster which have some consistency of their character stroke thickness and hence are more likely be part of a LP. Fig. 10 illustrates the scan-width filter on some example components in a cluster. Fig. 11 shows the binarised image after the filtering operation.
- For symmetry, LP generally tends to have a similar height of LP characters and a similar width of LP characters. We, therefore, examine the components in the remaining clusters and remove those components whose height and width are at much difference to other components in the clusters. Fig. 12 illustrates the height and width filter on some example components in a cluster. Fig. 13 shows the resultant image after the filtering operation.

4.4 LP border region filtering

The clustering and filtering stage results in a group of components which are, close to each other, are in a line, and have similar properties like a higher percentage of change when thinned, a higher ratio of area to perimeter length, similar component thickness, height similarity and width similarity. These properties are seen in LP characters. However, these properties are also possessed by components in other regions of the source image which have textual content. These other regions may however not be part of the LP. We now filter the clusters that form part of the LP from clusters in other textual regions but which are not part of the LP.

Typically, the LP is enclosed in a boundary. Further, the boundary is usually not too far from the actual LP characters and the LP character tends to occupy a sizeable portion of the area between the top and the bottom boundary. The boundary is formed due to the LP being of a different colour than the colour of the rest of the vehicle or the LP being in a frame. Due to this when the source image is converted to a binary image, the LP tends to be enclosed in a nearby boundary-like region while other regions with text characters do not have a boundary enclosing them. This boundary may not be continuous and may be broken due to thresholding, but the presence of a boundary-like enclosure is helpful in differentiating between clusters that belong to the LP and clusters that belong to other textual regions of the image. The steps used to check for boundary enclosure are illustrated below:

- We examine each remaining cluster. For each of the remaining clusters, we examine each of the components in the cluster. For each of the components, we start at the top-left corner of the component and traverse in a vertical direction the initial binary image from which the component was extracted. A boundary in the binary image will be the first lit pixel encountered while traversing in the vertical direction. The distance to be traversed in pixel count can be configured as a multiple of a typical LP component pixel height.
- We first traverse the binary image from the top-left corner of the component in an upward direction and look for the presence of a boundary or until we traverse a pre-specified distance. If a top side boundary pixel is found we store the coordinate of this pixel and go to the next step.
- We next traverse the binary image from the bottom-left corner of the component in a downward direction and look for the presence of a boundary or until we traverse a pre-specified distance. If a bottom side boundary is found we store the coordinate of this pixel and go to the next step.
- If no top or bottom border pixel is found we continue looking for them using the next pixel in the top-left corner of the component until we have covered the entire component width. If the component width is crossed, we move on to the next component in the cluster.
- We retain those clusters in which a top and a bottom border pixel are found and discard other clusters for which no top or bottom boundary pixel was found.

These steps thus retain those clusters which have a boundary-like region enclosing them. Fig. 14 illustrates the border detection operation to filter an example cluster that is enclosed by a border frame. Fig. 15 shows the cluster remaining in the binarised image after border region filtering.

Thus, using the additional filtering operations on clusters created using distance, line, and height-based clustering technique, we can remove many non-LP components and clusters with non-LP components, retaining only probable LP regions in the resultant image. For some images, the above steps eliminate all the non-LP components retaining just the LP components, while for other images some non-LP components continue to be retained. These

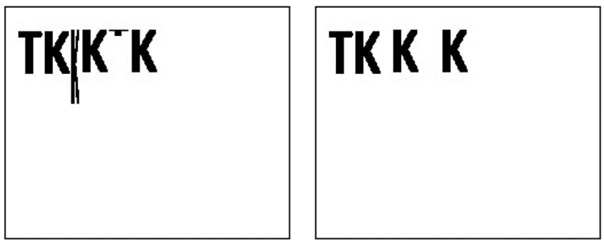

Fig. 12 *Height and width filtering operation*

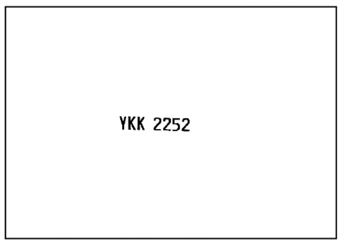

Fig. 13 *Resultant image after height and width filtering operation*

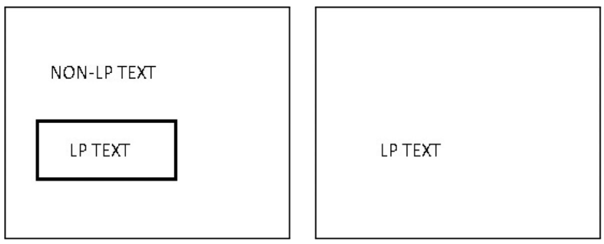

Fig. 14 *Border region filter operation*

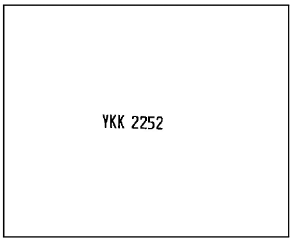

Fig. 15 *Binarised image after border region filtering*

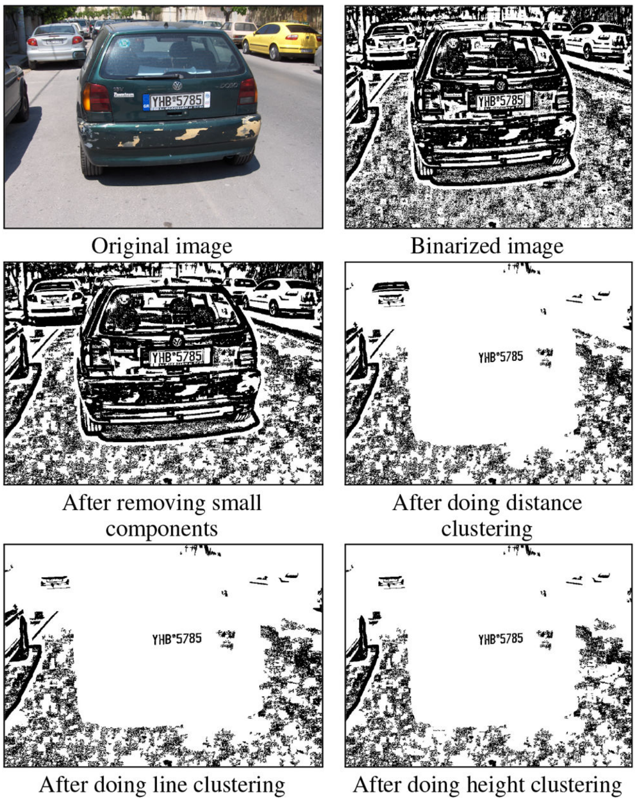

Fig. 16 *Clustering operations performed on source image*

remaining clusters and components can then be taken up for LP character identification to locate and identify the actual LP.

The computational complexity of the proposed approach is $O(N^2)$. The clustering step creates a group of clusters from the connected component in the image and the filtering steps then applies different filters to components in a cluster. The clustering step has a computational order of $O(N^2)$, where N is the number of components remaining at the end of each clustering step. The filtering step has a computational order of $O(M)$, where M is the number of components in each cluster on which the filter is applied.

Fig. 16 uses another sample image to illustrate each step of the clustering operation and Fig. 17 illustrates each step of the filtering operation. For the last image in Fig. 17, where the LP is shown identified, template matching was used to identify the LP characters in the remaining components.

The clustering operations mentioned in this section require a few threshold values like the maximum distance between the components, the maximum angle and the maximum height differences between the components etc. to be specified. The filtering operation also requires threshold values based on which components will be retained or removed. These threshold values can be specified as pre-configured context-dependent value and can be further tuned to satisfy country-specific LP and LP character constraints.

The next section presents the results obtained using the proposed approach on various publicly available LP datasets.

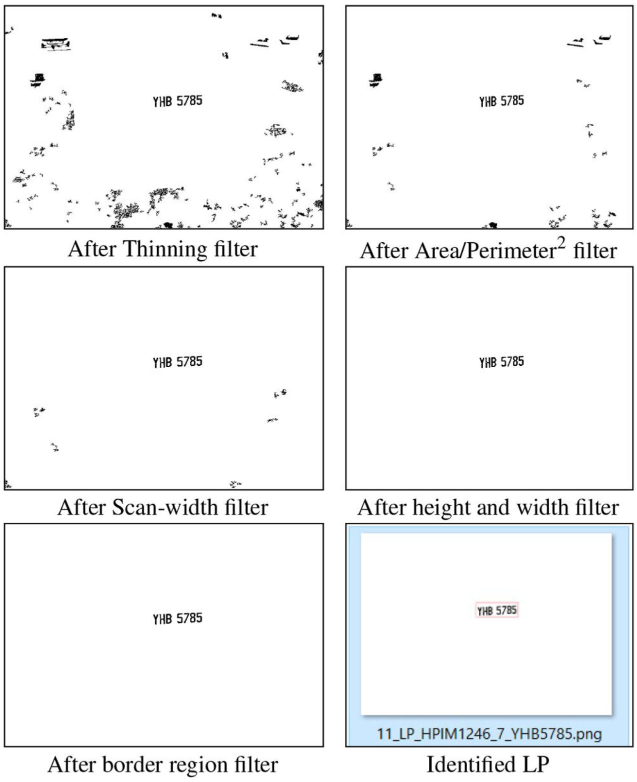

Fig. 17 *Isolating LP using filtering operations*

5 Experimental results

The clustering and filtering techniques presented in this paper were implemented in Java and the resulting Java application was run on an Intel Core 5 processor with 8 GB of RAM. The Java application took as input the set of images which need to be processed for LP detection and a text file with a set of values that specified the threshold values to be used for the clustering and filtering operation. The output was a set of binary images with the LP highlighted. The time taken to locate the LP for an input image of size 1024×768 pixel was ~1 s. We tested the method using images having different plate variations and weather conditions in an open environment from countries like Greece, Taiwan, Israel, Hungary and USA which were available on the web, with images of proprietary Indian LPs, and with images from publicly available LP datasets like Media-lab LP dataset (Greece), AOLP dataset (Taiwan). The results of our proposed method were checked against a total of 3600 sample source images. Of these 3600 images, 3425 were from publicly available sources and the rest were proprietary LP images. We checked the performance of 22 published LP detection methods and have compared the performance of our proposed method with published LP detection method which have shared their results across two or more publicly available datasets.

Table 1 summarises the various types of LPs against which we tested the proposed approach. The corresponding images and test results for images used in Table 1 are shown in Figs. 18–24. Figs. 18–24 show the results obtained for different categories of the LPs, which include images of Greek, Taiwanese, Israeli, American, Hungarian and proprietary Indian LPs with different plate variations, and in various environmental, and weather conditions. The odd columns of Figs. 18–24 show the input image and the even columns show the filtered output image having the LP.

In an open environment, there are many ways in which a camera can capture an image. The proposed geometry-based clustering and filtering techniques are invariant to size, tilt, pan, and rotation. The method is able to locate the LP captured from different angles, at different distances, and under different environmental conditions, and with multiple LPs in the source image. The method is also able to locate the LP that have LP characters in multiple lines. The proposed method can, therefore, work for different types of vehicles, motorcycles, vans, and trucks

Table 1 Summary of different types of LPs shown in Figs. 18–24

Sr. no.	Fig. no.	Country of LP	Image details	Located by method, Y/N
1	18a1	Greece	image in an open environment [23]	yes
2	18a2	Greece	image with shadow on plate [23]	yes
3	18a3	Greece	image with dirt and screw in the LP [23]	yes
4	19a1	Greece	night image with reflectance [23]	yes
5	19a2	Greece	image with scattered lights at night [23]	yes
6	20a1	Greece	image with extreme pan [23]	yes
7	20a2	Greece	image with LP at a different place [23]	yes
8	20a3	Israel	image with a close view of LP (Web)	yes
9	21a1	Israel	image with different tilt (Web)	yes
10	21a2	Israel	image with different tilt (Web)	yes
11	21a3	Hungary	car images from Hungary [24]	yes
12	22a1	Hungary	car images from Hungary [24]	yes
13	22a2	America	car images from America [25]	yes
14	22a3	America	car images from America [25]	yes
15	23a1	Taiwan	image from AC Set [8]	yes
16	23a2	Taiwan	image from LE Set [8]	yes
17	23a3	Taiwan	image from RP Set [8]	yes
18	24a1	India	image with multiple LPs (proprietary)	yes
19	24a2	India	image with multiple LPs (proprietary)	yes
20	24a3	India	image with uneven illumination (proprietary)	yes

Fig. 18 *Locating LPs from Greece*

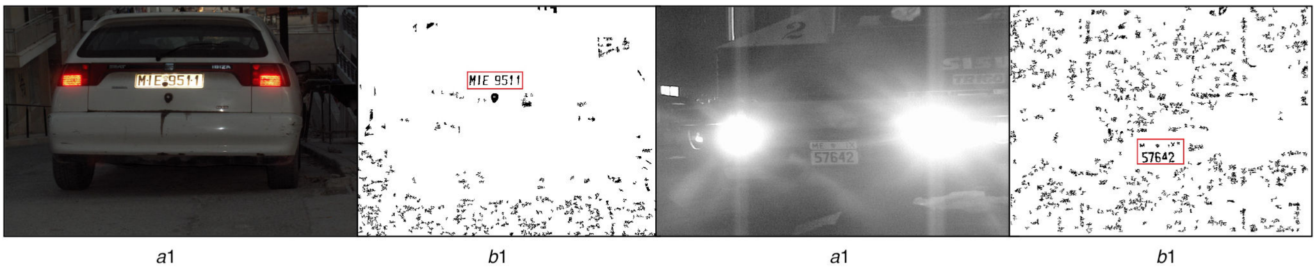

Fig. 19 *Locating LPs in low light*

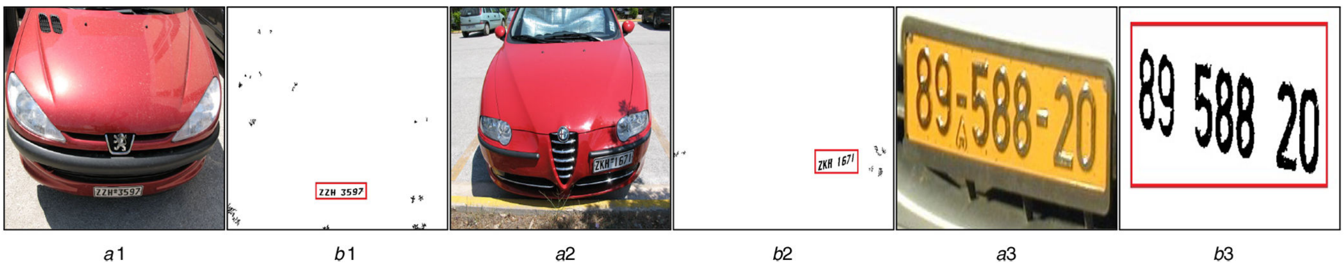

Fig. 20 *Locating LPs in different locations*

having multiple lines of LP characters and where each line of characters may have different sizes.

The method proposed by Soora and Deshpande [22] also uses clustering to detect the presence of LP in a source image. However in that method and as shown in Fig. 16, a large number of non-LP components from the input image are retained as-is in the output, and these high number of non-LP components reduces the accuracy of the method in locating the LP. In the proposed method and as shown in Fig. 17, we eliminate a significant number of non-LP components which helps reduce the number of components which need to be checked for LP character matches. Table 2 provides a comparison between the method in [22] and the proposed approach in removing non-LP components from the input image.

In Table 3, we list the performance of some prominent and published LP detection methods which used either proprietary datasets or used common publicly available benchmark datasets for reporting their results.

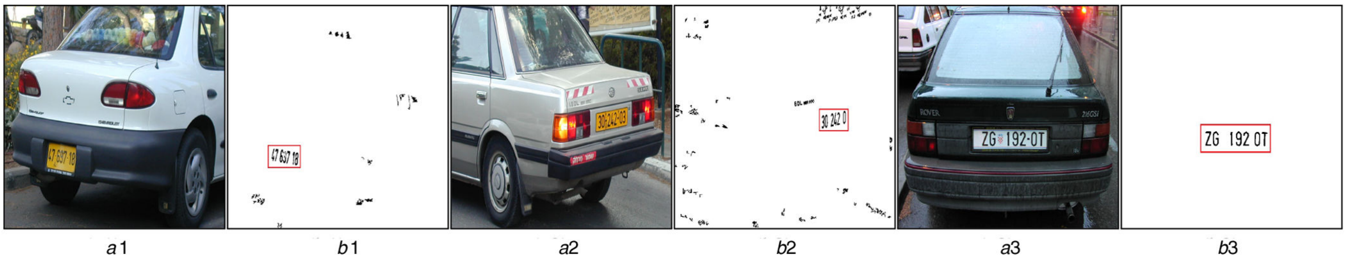

Fig. 21 *Locating LPs from Israel and Hungary*

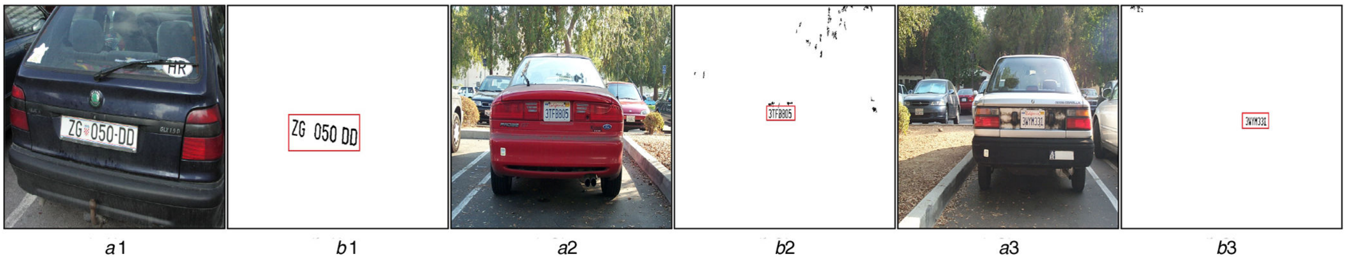

Fig. 22 *Locating LPs from Hungary and America*

Fig. 23 *Locating LPs from Taiwan*

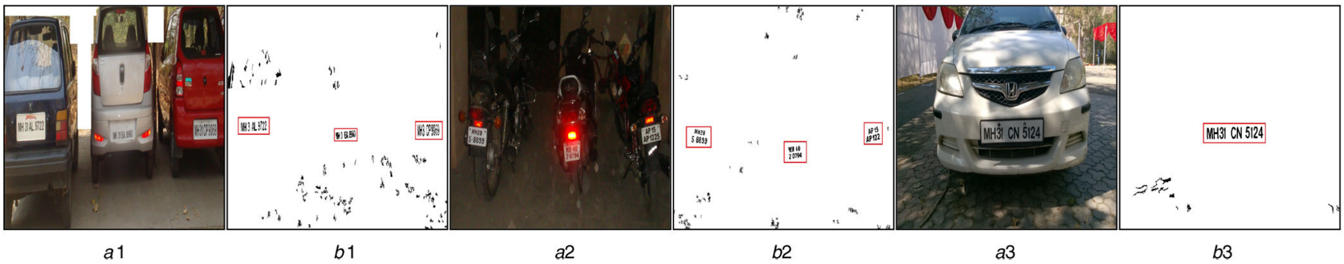

Fig. 24 *Locating LPs from India*

Table 2 Comparison between the proposed method and another clustering method for non-LP component removal using Media-lab and AOLP benchmark LP datasets

Benchmark LP dataset	LP dataset details	Environmental conditions	NON-LP component retained in output image	
			Clustering method [22], %	Proposed method, %
AOLP benchmark LP dataset [8]	2049 images having vans, trucks and cars	open environment, different plate variations	>90	<5
media-lab benchmark LP dataset [23]	741 images having vans, trucks and cars	open environment, different plate variations	>90	<5
AVERAGE of removal rate			>90	<5

At 99.6% the method proposed by Hongliang and Changpingin [4] has the highest success rate and outperforms all the listed methods, but this result has been reported using a proprietary dataset which is not publicly available. The methods proposed in [14–16] report high results but have reported these using only around half of the total images in the dataset. The methods proposed in [3, 6, 10, 12, 13] report high success rates but these have also used proprietary datasets which are not publicly available for verification and comparison. The methods used in [1, 2, 5, 7, 11] reported low success rates even with proprietary datasets. The method from [20] reports a 96.5% success result but uses a mixed set of proprietary images and images from the Media-lab dataset. The method in [9] reports a high result for the Media-lab dataset but does this using a dataset-specific SE and so it not a universal method.

Recent CNN-based methods proposed in [17–19] also reported high success rates, but achieve it with proprietary additions to the dataset to improve accuracy. These methods reported processing times of 400–700 ms for processing images from the AOLP [8] dataset in which the image sizes varied from dimensions of 320 ×

Table 3 Performance of different LP detection methods

Algorithm and references	LP dataset type	Number of images in dataset	Types of vehicles in the dataset	LP detection rate, %
CIP [1]	proprietary	80	cars	91.25
FCC [2]	proprietary	150	not reported	95.3
Yuan [3]	Caltech + proprietary	3977	cars	96.62
ESM [4]	proprietary	9825	vans, trucks, cars	99.6
RTR [5]	Caltech	1999	cars	84.8
SWHVP [6]	proprietary	522	motorcycles	97.55
VEDA [7]	proprietary	664	cars	91.65
Yepez and Ko [9]	Media-lab [23]	741	cars	98.45
two pass [10]	proprietary	9026	cars, trucks	97.16
PVW [11]	proprietary	410	not reported	93.2
Mingdong [12]	proprietary	Not reported	cars	95.55
RELIP [13]	proprietary	100	not reported	97
DIP-GA [14]	Media-lab [23]	335	vans, trucks,cars	97.61
SHGA [15]	Media-lab [23]	336	cars	98.5
Khan *et al*. [16]	Media-lab [23]	324	cars	99.3
CNN [17]	Caltech + proprietary	Not reported	cars	98.39
CNN [18]	AOLP [8] + proprietary	2049 + Custom	cars	99.5
CNN [19]	AOLP [8] + proprietary	2049 + Custom	cars	99.25
Soora [22]	Media-lab [23]	741	cars	97.3
Soora [22]	AOLP [8]	2049	cars	93.7
Baza [24]	Baza [24]	509	cars, vans	90
proposed method	Cars [25]	126	cars, vans	80.76
proposed method	Baza [24]	509	cars, vans	87.54
proposed method	proprietary	175	cars, trucks, motorcycles	98.23
AOLP [8]	Media-lab [23]	741	vans, trucks, cars	92.1
AOLP [8]	AOLP [8]	2049	car, vans	93.33
SCW [20]	Media-lab [23] and proprietary	1334	vans, trucks, cars	96.5
SCW [20]	AOLP [8] and proprietary	1334	vans, trucks, cars	81.67
proposed method	Media-lab [23]	741	vans, trucks, cars	96.52
proposed method	AOLP [8]	2049	cars, vans	90.32

240 pixels to 640×480 pixels. The processing time of these methods increases with increasing image sizes. In contrast, our method was able to process each image from the AOLP dataset in <300 ms.

It is not fair to perform a comparison between the results of methods which used proprietary dataset or dataset specific turning as we do not then have a common base for comparison of a universal method. In Table 4, we list the performance of a few prominent LP detection method and compare the performance of the proposed approach with a few of them using some common publicly available benchmarks. The method from [20] reports a result of 96.5% on the Media-lab dataset and reports a result of 81.67% on the AOLP data. The method from [8] reports a result of 92.1% on the Media-lab dataset and reports a result of 93.33% on the AOLP data. Using the Media-lab dataset, the proposed method's success rate of 96.52% is comparable to the results obtained by Anagnostopoulos *et al*. [20] and higher than that of the method used by Hsu *et al*. [8]. Using the AOLP dataset, the proposed method's success rate of 90.32% is higher to the result obtained by Anagnostopoulos *et al*. [20] but lower than that of [8]. However, with additional tunings of the threshold values, higher results can be obtained by the proposed method for a specific dataset. The clustering method from [22] reports a LP detection success rate of 97.3% for the Media-lab and 93.7% for the AOLP dataset. However due to the large number of non-LP components that are retained [22] will have a lower rate of accurately locating the LP, unfortunately the authors have not made the location rate available. Further the method from [22] takes more than 1 s to process an input image of size 1024×768 pixel and is slower than our proposed method.

In our approach, we have not assumed any fixed number of characters, size, or number of lines as being required for the LP. We have, however, used some threshold values to be used for the clustering and filtering operations. The threshold values used to present the results in this section are listed in Table 5. The values were obtained by considering a few sample source images from the Media-lab [23] dataset. The value of area/perimeter^2 listed was obtained by running a check against different alpha-numeric characters from multiple font styles and computing the ratio. These values can be tuned further for a specific dataset, to improve on the removal of non-LP components and locate the LP faster or to remove a lesser number of components and have higher LP detection results. However, for the result presented here we have used the same values, as listed, across all the tested datasets. Also, while testing against the Media-lab [23] dataset we found that around 40 images had dark shadows on the plates. These images were manually identified and separately pre-processed for shadow removal and then the modified images were used as inputs. Similarly, about 20 images had unreadable LPs and an output of no LP detected was considered as a success. The AOLP [8] dataset did not have any images with dark shadow and so shadow removal pre-processing was not required.

The methods proposed in the literature have used different techniques around edge information, morphological operations, template matching, and colour information of the LP. More recently CNN-based methods have also been used to identify the LP. These methods are restricted in LP detection as explained in Section 2. The proposed method is independent of colour, scale, rotation, number of lines, number of characters etc. in the LP and is thus, as illustrated in Table 1, able to identify single or multiple LPs across multiple regions, across different environments and with different plates and characters variations.

The proposed approach will fail to detect the LP from an input image if the LP characters touch the border of the LP or there are less than the minimum cluster size characters in the LP or there are very small unreadable characters present. It also fails if there is a

Table 4 Performance comparison of the proposed method with SCW, AOLP methods using Media-lab and AOLP benchmark LP datasets

Benchmark LP dataset	LP dataset Type	Types of Images in dataset	SCW method [20], %	AOLP method [8], %	Proposed method
AOLP benchmark LP dataset	2049 images having vans, trucks and cars	open environment, different plate variations	81.67	93.33	90.32
Media-lab benchmark LP dataset	741 images having vans, trucks and cars	open environment, different plate variations	96.5	92.1	96.52
average success rate	—	—	89.09	92.72	93.42

Table 5 Threshold values used for clustering and filtering operations

Parameter	Value	Description
M_p	3	minimum pixel size for a component to be retained.
B_p	8	window size in pixels for Bernsen local thresholding algorithm
C_d	30	maximum pixel distance between components in a distance cluster.
C_a	10	maximum difference in angle (degree) between components in a line cluster.
C_h	30	maximum pixel difference in height between components in a height cluster.
C_s	2	minimum size of a cluster to be retained.
T_p	50	minimum change percentage on thinning for a probable LP components.
AP_r	0.010	minimum area/perimeter^2 ratio value for probable LP components.
SW_m	20	percentage of top and bottom scan-width values to be ignored for a component.
Bo_p	3	multiple of the component height to be used for checking border presence
D_m	25	maximum percentage difference between any other values (width or height) of probable LP components

dark shadow across the LP. For such images, as mentioned earlier, a pre-processing step to remove the dark shadow will be required.

6 Conclusion

In this paper, we have proposed a new method for LP detection which uses multiple filtering operations applied to clusters created using clustering techniques based on geometrical properties of LP characters. The proposed approach is not specific to any country, has no restriction on the size or number of characters or number of lines in LP. The method can detect multiple LPs in an image as illustrated in Table 1 and Figs. 18–24. Thus, the approach is less restrictive as compared with most of the previously published work and works for many countries and for different variations in the plate and under different environmental conditions.

The Media-lab [23], AOLP [8] benchmark LP datasets were used to evaluate the proposed method's performance and the reported success rates was 96.52 and 90.32%, respectively, as shown in the performance comparison in Table 3. As shown in Table 4, the proposed method achieved a higher average success rate of 93.42% as compared to the method in SCW [20] which had an average success rate of 89.09% and the method in AOLP [8] which had an average success rate of 92.72%, when the three methods were tested over both the benchmark LP datasets.

The proposed approach fails to identify the LP, if components cannot be extracted due to the presence of dark shadows across plates or if the LP characters are missed due to the presence of junk like dirt or when the image is blurred, or the LP characters touch the LP border.

7 Acknowledgments

This work was done by us at VNIT Nagpur and is the outcome of the R&D work undertaken as a project under the 'Visvesvaraya PhD Scheme' of Ministry of Electronics & Information Technology (MEITY), Government of India, being implemented by Digital India Corporation.

8 References

[1] Lee, E.R., Kim, P.K., Kim, H.J.: 'Automatic recognition of a car license plate using color image processing'. Proc. of 1st Int. Conf. on Image Processing, Austin, USA, 1994, vol. 2, pp. 301–305

[2] Yang, Y.Q., Bai, J., Tian, R.L., *et al.*: 'A vehicle license plate recognition system based on fixed color collocation'. 2005 Int. Conf. on Machine Learning and Cybernetics, Guangzhou, China, 2005, vol. 9, pp. 5394–5397

[3] Yuan, Y., Zou, W., Zhao, Y., *et al.*: 'A robust and efficient approach to license plate detection', *IEEE Trans. Image Process.*, 2017, **26**, (3), pp. 1102–1114

[4] Hongliang, B., Changping, L.: 'A hybrid license plate extraction method based on edge statistics and morphology'. Proc. of the Pattern Recognition, 17th Int. Conf. on (ICPR'04) Volume 2. ICPR '04, Washington, DC, USA, IEEE Computer Society, 2004, vol. 2, pp. 831–834. Available at http://dx.doi.org/10.1109/ICPR.2004.44

[5] Faradji, F., Rezaie, A.H., Ziaratban, M.: 'A morphological-based license plate location'. 2007 IEEE Int. Conf. on Image Processing, San Antonio, USA, 2007, vol. 1, pp. I-57–I-60

[6] Huang, Y.P., Chen, C.H., Chang, Y.T., *et al.*: 'An intelligent strategy for checking the annual inspection status of motorcycles based on license plate recognition', *Expert Syst. Appl.*, 2009, **36**, (5), pp. 9260–9267. Available at http://dx.doi.org/10.1016/j.eswa.2008.12.006

[7] Al.Ghaili, A.M., Mashohor, S., Ramli, A.R., *et al.*: 'Vertical-edge-based car-license-plate detection method', *IEEE Trans. Veh. Technol.*, 2013, **62**, (1), pp. 26–38

[8] Hsu, G., Chen, J., Chung, Y.: 'Application-oriented license plate recognition', *IEEE Trans. Veh. Technol.*, 2013, **62**, (2), pp. 552–561

[9] Yepez, J., Ko, S.: 'Improved license plate localisation algorithm based on morphological operations', *IET Intell. Transp. Syst.*, 2018, **12**, (6), pp. 542–549

[10] Wen, Y., Lu, Y., Yan, J., *et al.*: 'An algorithm for license plate recognition applied to intelligent transportation system', *IEEE Trans. Intell. Transp. Syst.*, 2011, **12**, (3), pp. 830–845

[11] Zhou, W., Li, H., Lu, Y., *et al.*: 'Principal visual word discovery for automatic license plate detection', *IEEE Trans. Image Process.*, 2012, **21**, (9), pp. 4269–4279

[12] Mingdong, L.: 'License plate character recognition method based on combination feature and BP network'. IET Conf. Proc., Shanghai, China, 2017, vol. 1, pp. 3(4)–3(4). Available at http://digital-library.theiet.org/content/conferences/10.1049/cp.2017.0105

[13] Haneda, K., Hanaizumi, H.: 'A flexible method for recognizing four-digit numbers on a license-plate in a video scene'. 2012 IEEE Int. Conf. on Industrial Technology, Athens, Greece, 2012, pp. 112–116

[14] Samra, G.A., Khalefah, F.: 'Localisation of license plate number using dynamic image processing techniques and genetic algorithms', *IEEE Trans. Evol. Comput.*, 2014, **18**, (2), pp. 244–257

[15] Abo.Samra, G.: 'Application independent localisation of vehicle plate number using multi-window-size binarisation and semi-hybrid genetic algorithm', *J. Eng.*, 2018, **2018**, (2), pp. 104–116

[16] Khan, M.A., Sharif, M., Javed, M.Y., *et al.*: 'License number plate recognition system using entropy-based features selection approach with SVM', *IET Image Process.*, 2018, **12**, (2), pp. 200–209

[17] Kim, S.G., Jeon, H.G., Koo, H.I.: 'Deep-learning-based license plate detection method using vehicle region extraction', *Electron. Lett.*, 2017, **53**, (15), pp. 1034–1036
[18] Xie, L., Ahmad, T., Jin, L.*, et al.*: 'A new CNNN-based method for multi-directional car license plate detection', *IEEE Trans. Intell. Transp. Syst.*, 2018, **19**, (2), pp. 507–517
[19] Li, H., Wang, P., Shen, C.: 'Toward end-to-end car license plate detection and recognition with deep neural networks', *IEEE Trans. Intell. Transp. Syst.*, 2018, **20**, (3), pp. 1–11
[20] Anagnostopoulos, C.N.E., Anagnostopoulos, I.E., Loumos, V.*, et al.*: 'A license plate-recognition algorithm for intelligent transportation system applications', *IEEE Trans. Intell. Transp. Syst.*, 2006, **7**, (3), pp. 377–392
[21] Salazar, A., Igual, J., Safont, G.*, et al.*: 'Image applications of agglomerative clustering using mixtures of non-Gaussian distributions'. 2015 Int. Conf. on Computational Science and Computational Intelligence (CSCI), Las Vegas, USA, 2015, pp. 459–463
[22] Soora, N.R., Deshpande, P.S.: 'Color, scale, and rotation independent multiple license plates detection in videos and still images', *Math. Probl. Eng.*, 2016, **2016**, p. 14. Available at 10.1155/2016/9306282
[23] Anagnostopoulos, C.E., Anagnostopoulos, I.E., Psoroulas, I.D.*, et al.*: 'Medialab LPR database, multimedia technology laboratory, National Technical University of Athens'. Available at http://www.medialab.ntua.gr/research/LPRdatabase.html
[24] Kalafatíc, Z.: 'Baza car dataset, university of Zagreb, 500 images of the rear views of various vehicles'. Available at http://www.zemris.fer.hr/projects/LicensePlates/english/results.shtm
[25] Weber, M.: 'Car dataset taken by Markus Weber. California Institute of Technology, 126 images of cars from the rear'. Available at http://www.vision.caltech.edu/html-files/archive.html