Journal of
Computer Science & Systems Biology

Safaei et al., J Comput Sci Syst Biol 2016, 9:3
http://dx.doi.org/10.4172/jcsb.1000225

Research Article Open Access

Robust Search-Free Car Number Plate Localization Incorporating Hierarchical Saliency

Safaei A*, Tang HL and Sanei S

Faculty of Engineering and Physical Sciences, University of Surrey, Guildford, UK

Abstract

There are two major shortcomings associated with presently implemented automatic license plate recognition (ALPR) systems: first, processing images with complex background is time-consuming and second, the results are not sufficiently accurate. To overcome these problems and also to achieve a robust recognition of multiple car number plates, saliency detection based on the ALPR system is used in this paper and also an improved and more effective definition of saliency is presented. In this new approach, the notion of the directionality of the edges using Gabor filtering and the detection of the patterns of numbers using L_1*-norm* have been added to the traditional saliency detection method. The proposed algorithm was tested on 660 images; some consisting of two or more cars.

A detection accuracy of 94.77% and an average execution time of 40 ms for 600 × 800 images are the marked outcomes. The proposed SB-ALPR method outperforms most of the state of the art techniques in terms of execution time and accuracy, and can be used in real-time applications. Also, unlike some recently introduced saliency-based ALPR methods, our two-stage saliency detection approach exploits smaller numbers of sample sizes to reduce the computation cost.

Keywords: Automatic license plate recognition; Gabor filter; L_1-*norm*; Saliency map detection; Search free

Introduction

Automatic license plate recognition (ALPR) has become increasingly popular in intelligent transportation systems. ALPR consists of three main stages: localization, segmentation, and optical character recognition (OCR). The quality of images taken by traffic cameras throughout the city is rapidly improving and therefore the size and the resolution of these images have been moderated. Most of the available localization methods, which have to deal with large-size images in real-time scenarios, become computationally intensive when cars are moving and therefore, very time-consuming. Developing more adequate algorithms for localization of car number plates for faster processing of such large images often becomes necessary.

Many researchers regard localization as the most important step of ALPR. It is also argued that the localization step, by itself, is strongly influenced by the exactness of the general ALPR system. Several methods have been developed for localizing the number plate on a car, to name a few: edge detection using Gabor and wavelet transforms, color properties of the number plates [1], discrete Fourier transform (DFT) [2], and analyzing the histogram [3]. In some approaches, the car number plates are localized through the application of one or more of the above methods.

According to Yu et al. [4], vertical edges can be found, and by using size and shape filters, the unnecessary edges can be eliminated. Yu et al. achieved the localization rate of 98.45% by finding the two vertical edges of the number plate. Although this method has some disadvantages, one of its advantages is its simplicity. Xu et al. [5] argue that this method only works for continuous edges and its accuracy is low for complex backgrounds.

Image transforms such as wavelet or Gabor have been popular for plate localization and recognition as they offer multi-resolution image analysis and represent important properties of the objects such as their edge orientations. Wang et al. [6] proposed a wavelet transform approach followed by hierarchical detection of the edges within four image sub bands of HH, HL, LH and LL. A reasonably high accuracy of 97.33% in detection has been reported for this scheme. However, according to Du et al. [7], these methods are sensitive to the noise level and the complexity of the background in images. In other approaches, the plate's color is used to locate it. Variability in lighting from both environment and the camera receptors often deteriorates the performance of such methods. Template-matching has also been proposed by Ashtari et al. [8] for detecting European and Iranian number plates, achieving up to 96% accuracy. However, the absence of color cameras in some control points as well as bearing an exhaustive search is amongst disadvantages of this method. There are many other algorithms which combine the above methods which, when applied individually, often fail in the localization process because of the aforementioned reasons. Such algorithms can be found in works carried out by Chang et al. [2] and Wen et al. [9].

License plate color is one of the features useful in the localization stage and the color difference between the character and the texture. Many researchers such as Wei-Gang et al. [10] and Badawy et al. [7] believe that the license plate color can be used to extract the plate region from the image. On the other hand, Badawy et al. [7] believe that the existence of characters in the license plate can be used as a feature in the localization stage and two or more features may be used in this stage for higher accuracy in localization.

One of the license plate extraction methods, that use global image information, has been recommended by Anagnostopoulos et al. [1], which uses connected component analysis (CCA) to scan the binary image and label its pixels according to the component base

***Corresponding author:** Amin Safaei, Faculty of Engineering and Physical Sciences, University of Surrey, Guildford, UK, Tel: 01483684615; E-mail: a.safaei@surrey.ac.uk

Received March 16, 2016; **Accepted** March 30, 2016; **Published** April 05, 2016

Citation: Safaei A, Tang HL, Sanei S (2016) Robust Search-Free Car Number Plate Localization Incorporating Hierarchical Saliency. J Comput Sci Syst Biol 9: 093-103. doi:10.4172/jcsb.1000225

Copyright: © 2016 Safaei A, et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

connectivity. It has been proved by Wu et al. [11] that if CCA is applied to a low resolution image, the correct detection rate would be still high. This method is easy and fast to run, but may generate disjoint regions.

Thanh-Tung [12] has presented a new framework for ALPR system based on boosting the learning technique carried out by Viola et al. [13], which uses the traditional Haar-like feature and the local binary pattern (LBP) algorithm proposed by Ojala et al. [14]. In order to improve the algorithm's performance, this framework presented a new mechanism for simultaneously detecting and classifying. The main purpose of using features instead of raw pixel values, as input to a learning algorithm, is to reduce the intra-class variability, while increasing the enter-class variability and adding insensitivity to certain image variations. This method performs the training step in three scenarios that produces three classifiers. After the training step, the trained classifiers will be applied to images or video for detection and localization of license plates, or will be combined at a later stage for improving the performance of the system. Thanh-Tung [12] claims that the system can perform real-time detection with high accuracy. However, not much information about the datasets and image sizes has been given and the work is still in progress. One of the advantages of this method is that it has a high precision during the learning process. However, in the feature extraction step, the calculation of Haar-like and LBP features are time-consuming and in the classification step, the process takes much time for learning and some pixels may be miss-classified.

Bai et al. [15] and Faradji et al. [16] argue that, since the shape of the license plate is usually rectangle with a known aspect ratio, the system can find all the existing rectangles with the same aspect ratio and extract the candidate plates using edge detection techniques such as Sobel algorithm. On the other hand, some researchers such as Zheng [17] argue that there is no need to use geometric attributes and this task can be done by only matching vertical edges. Wang et al. [18] state that using only horizontal edge for localization results in error due to car bumpers and that the magnitude of the vertical edges on the license plate is a robust extraction method in comparison to horizontal edge detection.

Soh et al. [19] and Xu et al. [20] both use a technique called scan line for this purpose. In this method, the changes of the gray-level correspond to the number of peaks in the scan line, which is equal to the number of characters. However, the algorithm does not work properly in a complex background, where many edges are found in the image. Some other authors such as Zunino et al. [21] state that vector quantization (VQ) can be used to locate the text in the image by giving it some hints about the regions of the image. Sliding concentric windows (SCW) has also been proposed by Anagnostopoulos et al. [1], where license plate numbers are viewed as abnormalities of the license plate texture so that every change in local characteristics has a potential to be a license plate number and can be extracted. However Deb et al. [22] believe that, if the histogram technique is added to SCW, the results would improve.

Other methods, such as by spatial frequency using DFT proposed by Parisi et al. [23], are used to detect plate and character positions. In their algorithms, characters are classified by a multi-layer neural network, which is trained by Block Recursive Least Squares algorithm.

Matas et al. [24] are a group of researchers that have worked on character features. In their algorithm, using the region based approach, all character-like regions are extracted. Then, the plate is assumed to be a liner combination of these small regions. Although this method is time-consuming, it is robust to rotation (since the character features are used instead of the edge patterns) and has a high rate of accuracy.

Abolghasemi et al. [25] has presented a new algorithm for license plate detection. In this method, two image enhancement features (edge density and intensity variance) are proposed to increase the contrast of plate-like regions and to avoid missing the plate location for poor quality images. Then a new fast filter is used to detect candidate regions as plate. This filter models the vertical edge density of the plate region regarding its neighborhood. The MNS method (Multi-modal neighborhood signature) is then used to find colored textures in the plate for detection purpose.

Kim et al. [26] also present a texture-based method for object detection in colored images. The SVM method (support vector machine) is used to analyze the color texture of the license plates. Then CAMShift (continuously adaptive mean- shift) algorithm is applied to detect the license plate regions. Although this method has a high rate of accuracy, it is time consuming.

Gou et al. [27] proposed a license plate recognition method based on ERs (character-specific extremal regions) and HDRBMs (Hybrid discriminative restricted Boltzmann machines). This method uses top-hat transformation, vertical edge detection and other morphological operations and also uses ERs to extract the character regions in license plate candidates. Then, an off-line trained pattern classifier is applied to recognize the characters.

The above methods for plate extraction in the ALPR process have their own pros and cons. Most of these methods are hard to be applied to a complex image. Since ALPR is used in intelligent transportation systems (ITS), it should be fast and highly accurate. Many of the current Persian ALPR algorithms do not meet these requirements. In our proposed method, we try to avoid any exhaustive searches and to present a novel search-free method for accurately locating the Persian LP number.

This paper introduces an innovative, robust and fast algorithm, suitable for localizing the plates of moving cars by first, developing a new saliency-based method by improving the traditional approach used in Itti et al. [28] and second, incorporating L_1-*norm* as a measure of sparsity into the edge detection process, to highlight the pattern of numbers and characters within the plate region. Finally, we show through experiments that, rather than just being an adequately fast algorithm in terms of execution time (time consumed to localize one plate number) and accuracy, the proposed saliency-based ALPR (SB-ALPR) method outperforms the existing benchmark techniques in ALPR for poor-conditioned colored images with simple or complex backgrounds, even in multiple-car scenarios.

Proposed Method

The block diagram of the SB-ALPR is shown in Figure 1. This method consists of five main stages, described in detail in the following sections.

Saliency map detection

Saliency refers to a discriminatory identity, based on some properties of an object or the regions of an image, to distinguish them from their background or surroundings. Saliency map detection (SMD) is derived from visual uniqueness or rarity and it is often dependent on certain features of the image such as changes in texture, gray-level, or color. SMD algorithms mainly fall into two categories; those which require low attention level (bottom-up) and those which

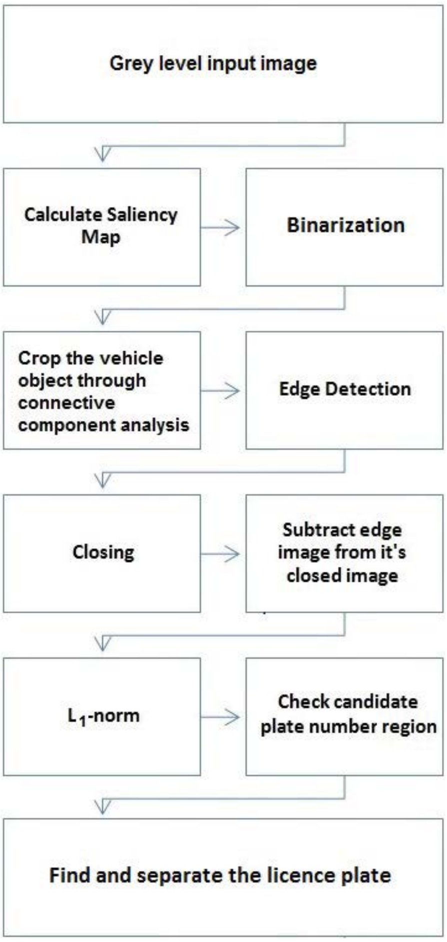
Grey level input image
Calculate Saliency Map
Binarization
Crop the vehicle object through connective component analysis
Edge Detection
Closing
Subtract edge image from it's closed image
L1-norm
Check candidate plate number region
Find and separate the licence plate

Figure 1: Block diagram of the proposed SB-ALPR method.

are described using high level (top-down) features mostly by using statistical or physical feature extraction methods. These maps are then integrated into a single top-down saliency map. However, according to Itti et al. [28], if the high level factors are identified, the performance level of SMD will be considerably higher. Figure 2 shows the Itti's SMD model [28] which is used as the basis for this work. The localization rate of this algorithm was high, when tested with our dataset. However, the speed of this algorithm is not sufficient for our purpose and some modifications were inevitable.

In our previous publication, Safaei et al. [29], we applied the saliency concept to ALPR for the first time. However, SMD method proposed by Itti et al. [28] was directly used and, in order to enhance the speed of this algorithm, some modifications were needed.

In order to locate the objects in the image, a new approach in SMD is proposed [28]. This is achieved by defining and extracting a number of features from the images. The first twelve features represent the color, the next six features are related to intensity variation and the remaining twenty four determine the center-surround differences.

First, the images are normalized. Then, by collecting normalized maps at each field, conspicuity maps are compiled and finally, the saliency map is obtained through normalized linear mixture of conspicuity maps. The saliency-based approach augments the saliency models by introducing a high-level algorithm which locates the regions indicating potential objects in the image, and a task-relevancy module that uses the expected locations of the candidate objects to further restrict the regions. The result of using a proto-object mask, as well as an expected location map, is that the low-level features of the image are simultaneously inhibited if they are unimportant for the task under execution, and enhanced if they are important for the observer's goal. The resulting map is called conspicuity map because it indicates which regions of an image are perceptually conspicuous as well as visually salient. Producing twelve features related to color and intensity requires high computational complexity. This problem is highly mitigated in our approach.

In addition, calculation of twenty four features for each of the $0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}$ directions is very time-consuming. Therefore in our SB-ALPR, we rely on the fact that car plates often have edges in 0 and $\frac{\pi}{2}$ directions. Therefore, in order to reduce the amount of calculations, other directions are ignored.

In order to produce the conspicuity maps, Greenspan et al. [30] proposed "Gaussian pyramid" representation of images, which can be obtained by hierarchical Gaussian filtering and down-sampling. This involves creating a series of images using Gaussian filtering and down sampling. By applying this technique, a stack of successively smaller images is created, with each pixel containing a local average that corresponds to a pixel neighborhood on a lower level of the pyramid. Then, by taking the derivative of the Gaussian pyramids, we form a "Laplacian pyramid" [30]. Next, unlike the work in Ref. [30] which uses sinusoids for the orientation identification, here Gabor filter is applied to the Laplacian pyramid leading to "orientation pyramid", which also acts as "direction map" [28]. In Itti et al. [28] method, six features for each direction are obtained; nevertheless, in the SB-ALPR, only 0 and $\frac{\pi}{2}$ directions are considered for the region of interest. Hence, just two other sets of features are enough and the orientation map is concluded by normalizing these two maps as shown below. Consider G_0 representing the input image. Then, based on Itti et al. [28] algorithm, by applying the Gaussian filter W_{Gau}

$$H_n = W_{Gau} * G_n \qquad n = 0, \ldots, 5 \qquad (1)$$

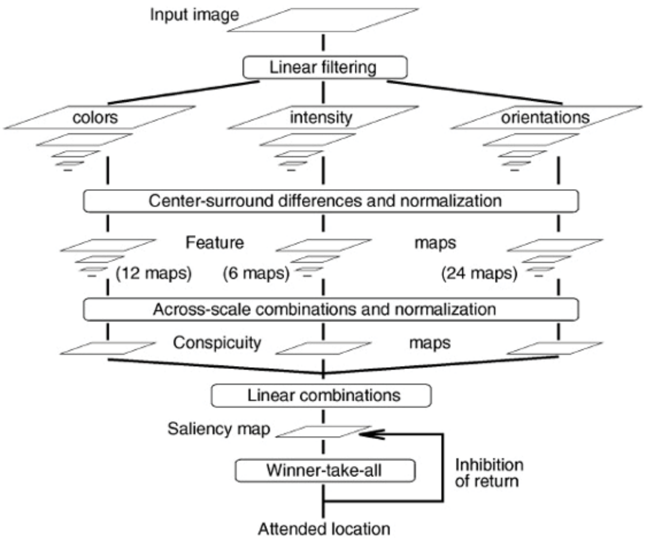

Figure 2: Itti's saliency map detection model [28].

where * represent the convolution operator. W_{Gau} acts as antialiasing 2-D filter.

$$G_{n+1} = downsampled\ H_n \qquad n = 0, ..., 4 \qquad (2)$$

$$L_n = G_n - H_n \qquad n = 2, ..., 5 \qquad (3)$$

$$O_n = W_{Gab} * L_n \qquad n = 2, ..., 5 \qquad (4)$$

$$OM1 = |O_2 - O_4| \qquad (5)$$

$$OM2 = |O_3 - O_5| \qquad (6)$$

where G_n is the image after the n^{th} stage of Gaussian filtering and down-sampling, that represents the Gaussian pyramid.

W_{Gau} shows the Gaussian filter, W_{Gab} shows the Gabor filter, L_n represents the Laplacian pyramid, O_n shows the direction pyramid, and $OM1$, $OM2$ represent the two "direction maps" that should be normalized. According to equation (3) L_n is equivalent to the differentiation of Gaussian filters and each of the differentiations affects $OM1$ and $OM2$ features. The normalized maps are obtained through the following three stages:

- Elimination of amplitude differences by adjusting the values in the "feature map" (gray-scale levels) to the range 0 to 1 proposed by Ref. [28].
- Finding the highest five local peaks and calculating their average ($\bar{m}$)
- Normalization of the values by scaling them to interval [0, 1] for obtaining a normalized map $(\widehat{OM1})$, where $\hat{\alpha}$ denotes normalized α.

Upon normalizing $OM1$ and $OM2$, the final "direction map" is obtained as:

$$OM = \widehat{OM1} + \widehat{OM2} \qquad (7)$$

As shown in Figure 3, In order to increase the contrast of the number plates, given that their backgrounds are white, normalization of map O_2, shown as $\widehat{O_2}$, is added to normalization of OM to make the final saliency map as:

$$OM_f = \widehat{OM} + \widehat{O_2} \qquad (8)$$

Figure 3 shows the enhanced saliency map detection algorithm. In this Figure, ↓1/2 refers to down-sampling by 2 and |.| stands for absolute value. Figure 4 shows the results of its implementation on a data sample.

Table 1 shows the results of the evaluation of the algorithm by Itti et al. [28] in comparison to our new and improved SMD method, in terms of recognition accuracy and processing time. Both algorithms were tested using a dataset of 100 infra-red images which had complex background as shown in Figure 4. The accuracy or the localization rate is defined as the percent- age of the correctly-detected out of the total number of plates in the dataset.

As shown in Figure 1, the gray image is applied to the saliency detection block. In order to increase the speed, first the $M \times N$ gray image is down-sampled by 2. Then the proposed improved algorithm is examined. Upon finding each connected component (CC) in the object map, in compliance with the size of the object map in the image and the gray-scale image, the corresponding pixels are cropped from the gray-scale image.

The reason why the proposed method is over 30 times faster than Itti's SMD model is that, in our proposed SMD algorithm, we down-sampled the images and also we eliminated the intensity and the orientation maps which seem to be unnecessary for this application. Next section discusses this issue and evaluates the proposed SMD

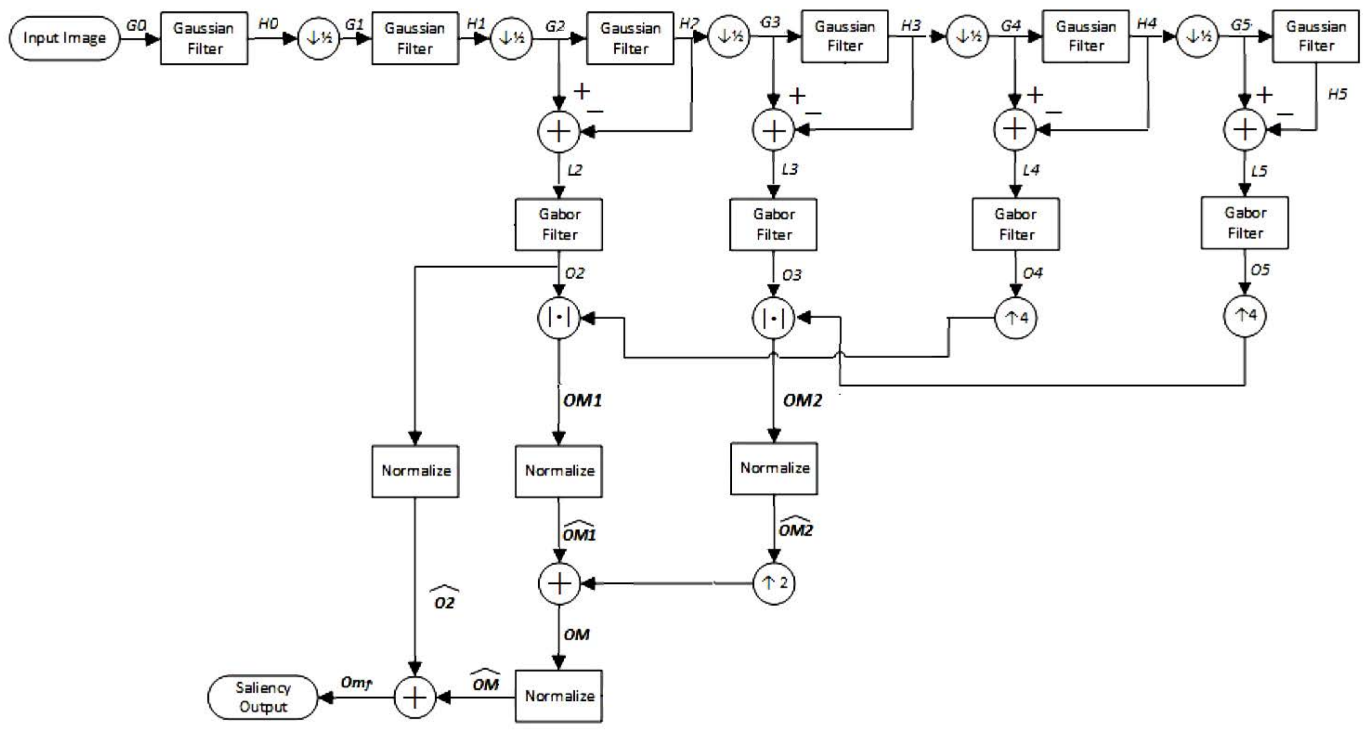

Figure 3: Enhanced saliency map detection.

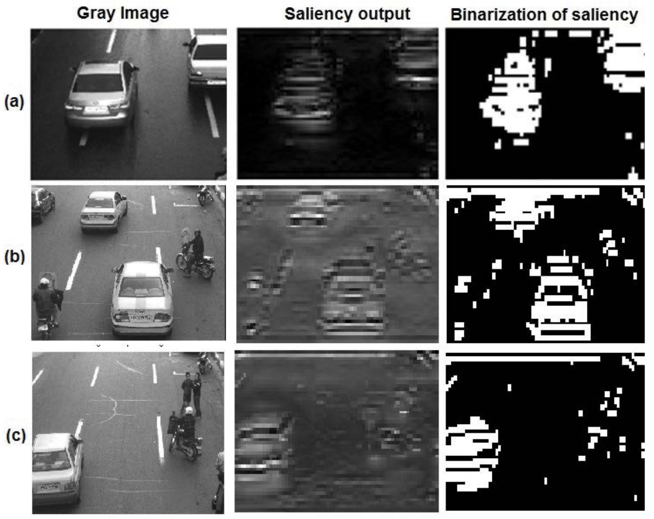

Figure 4: The result of applying the enhanced saliency map detection algorithm to gray-scale images for simple and complex backgrounds.

Saliency Method	Time (ms)	Accuracy (%)	Image Size
Proposed Method	2.9	96	960 × 1280
Itti's Method	104.5	95	960 × 1280

Table 1: Comparing the time and accuracy of the SB-ALPR algorithms to those of the benchmark Itti'method. Here MATLAB 8.2.0 (2013b), Core i7, and clock frequency of 2.70 GHz have been used.

algorithm.

Evaluation of proposed SMD algorithm

In order to evaluate this method under different conditions, we tested the algorithm in three ways. In the first experiment, we investigated the effects of adding the "intensity maps" to "orientation maps" (with the direction of ninety degree). The Gaussian filter parameters (μ_i, σ_i^2) for an image of $N_i \times M_i$ and are mean, $\mu_i = (\frac{N_i}{2}, \frac{M_i}{2})$ and variance, $\sigma_i^2 = \frac{\sigma_0^2}{2^i}$ (where σ_0^2 is the initial variance), $i = 1, ..., 5$.

We made use of G_n previously calculated in the orientation map:

$$IM1 = |G_2 - G_4| \tag{9}$$

$$IM2 = |G_3 - G_5| \tag{10}$$

where G_2, G_4 and G_3, G_5 are the down-sampled versions of the Gaussian filtered images. Then we normalized and added them together:

$$IM = \widehat{IM1} + \widehat{IM2} \tag{11}$$

To calculate the saliency map based on intensity and orientation, we normalized and added IM and OM_f calculated previously in (8):

$$FinalSal_1 = \widehat{IM} + \widehat{OM_f} \tag{12}$$

Figure 5 shows the results of this operation. Based on this, the final saliency map ($FinalSal_1$) is similar to the orientation map and changing the angle does not affect the map.

- Pros: Due to normalization of the intensity map, the values of this map become very small (local maxima have approximately the same values), so there are no advantages in adding this map to the orientation map.
- Cons: Increased computation time.

In the second experiment, we added the directions of 0, 45 and 135 to the orientation map at the 90 degree direction. We used equations (1) to (9) for calculating the orientation maps in directions 0, 45 and 135 using W_{Gab} for each direction.

Next, we calculated $OM_f^{\angle 0}$, $OM_f^{\angle 45}$, and $OM_f^{\angle 135}$, which were then normalized and added together:

$$FinalSal_2 = \widehat{OM_f^{\angle 90}} + \widehat{OM_f^{\angle 0}} + \widehat{OM_f^{\angle 45}} + \widehat{OM_f^{\angle 135}} \tag{13}$$

where $\widehat{OM_f^{\angle \theta}}$ refers to rotated $\widehat{OM_f}$ by angle θ. From the saliency map based on four directions, we observe that other directions increase salient pixels that are not related to the cars. Figure 6 shows the procedure and that the final saliency map is identical to the orientation map 90. This empirical conclusion is then taken into account to further reduce the complexity.

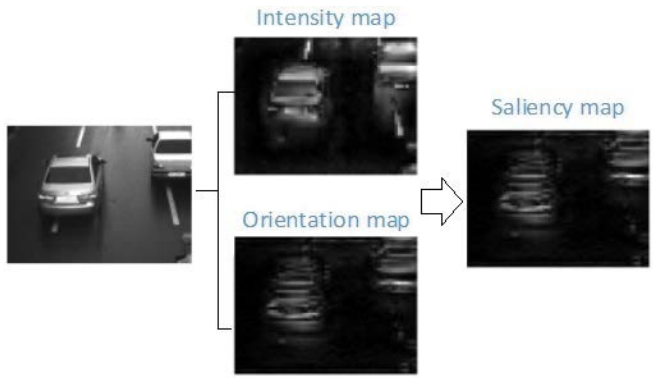

Figure 5: Saliency map obtained by adding orientation (angle=90°) and intensity maps together.

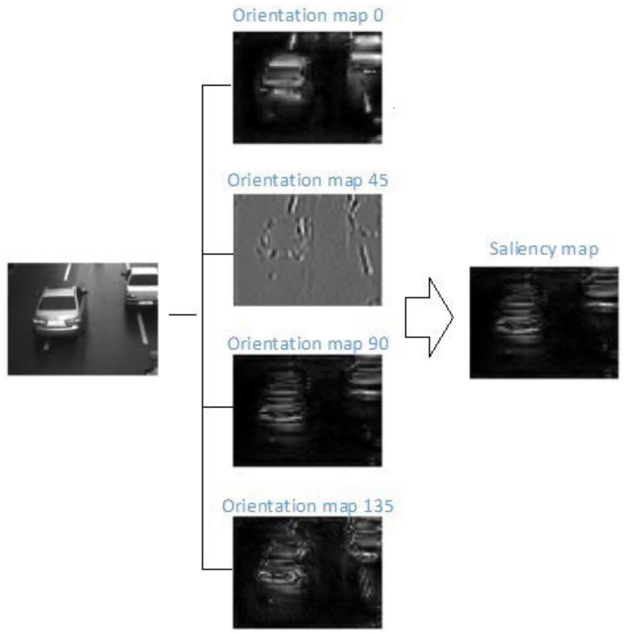

Figure 6: Saliency map by adding four orientation maps together.

- Pros: High detection rate for salient pixels which may be missed if only the orientation method is used.
- Cons: Increase in the computation time and the number of salient pixels that are not related to the cars (increased false positive).

To investigate the robustness of the algorithm, we added Gaussian and salt-pepper noises to the gray-scale images and calculated the accuracy of the saliency map. As shown in Figure 7, as the signal to noise ratio (*SNR*) increases, the detection rate grows, and the false alarm is decreased. However, since our dataset to evaluate *SNR* is small, detection probability rate has some fluctuations. *SNR* is defined as:

$$SNR = 10log10(\frac{p_{signal}}{p_{noise}}) \qquad (14)$$

where p_{signal} and p_{noise} are respectively the signal and noise powers. Our saliency detection method is adequate while *SNR*>2 *dB* when Gaussian noise is added. As shown in Figure 7, when *SNR* changes from 0 to 10 dB, the detection probability varies between 0.7 and 0.62. However, the false alarm increases when *SNR* is reduces.

As shown in Figure 8, our method for saliency is adequate if the noise level<0.4 when salt and pepper noise is added. However, the detection rate is highly sensitive to this noise level.

Detecting the car region

According to the block diagram in Figure 1, with the help of saliency, the region of the car can be identified. The steps for identifying a car region are as follows:

- Setting a counter to 0.
- Finding the pixels with maximum gray-level in saliency feature map.
- Determining the pixels in the CC which would have the highest

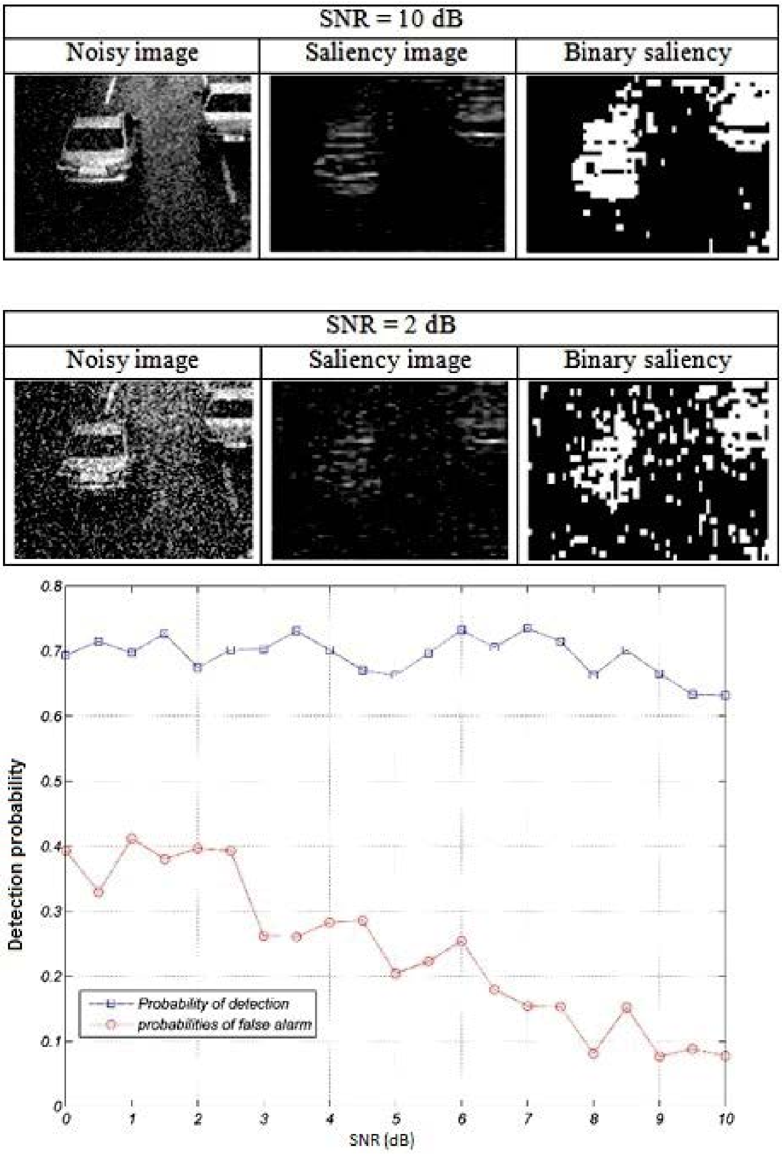

Figure 7: Detection probability versus SNR when Gaussian noise is added.

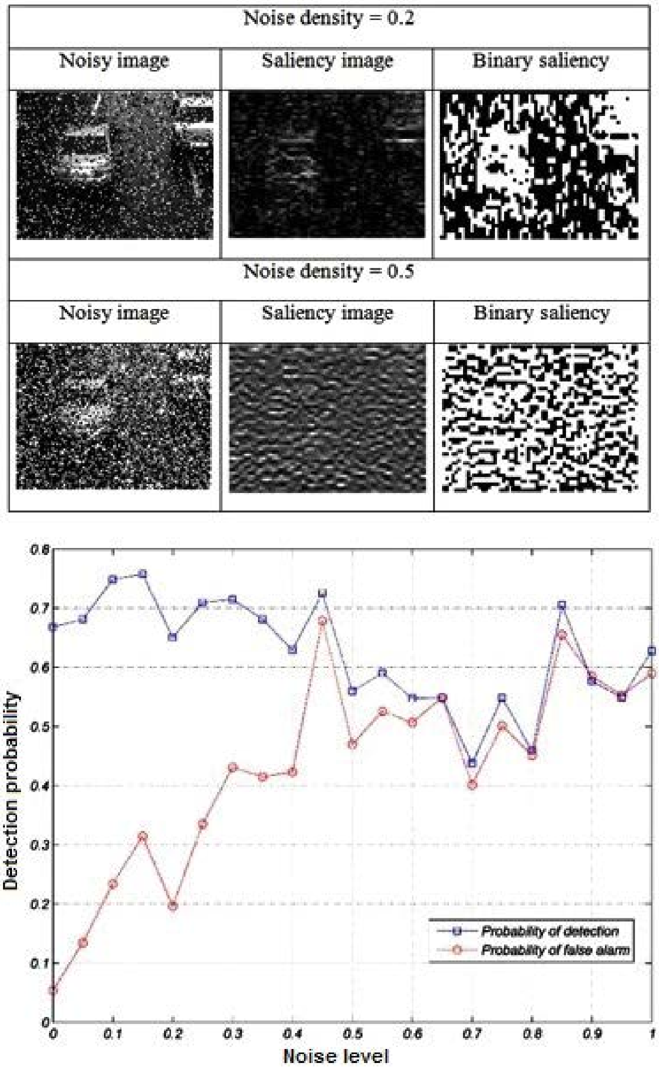

Figure 8: Adding salt & pepper noise with density=0.2 and 0.5.

chance of bearing the number plate (Using L_1-*norm* and cropping these corresponding pixels out of this CC region).

- Setting the above-selected pixels in the saliency map to zero (black) and increment the counter by 1.
- Returning to step 1 only if the number of black pixels is not higher than a predetermined limit (default is 3), otherwise the process terminates.

Figure 9 shows this algorithm for the image in Figure 4(a) in which, the maximum expected number of plates is set to 3. In Figure 9(a), the saliency map shows that the pixels around coordinate (17,59) have the most gray-surface and the CC corresponding to these pixels makes up the "object map." Figure 9(c) shows the cropped image from the CC (by observing a safe margin e.g., 10 pixels) of gray image and the pixels corresponding to this CC are set to zero in the saliency object map. In Figures 9(d-i), the repetition of this process is shown. It is interesting that there is the possibility of observing the same number plate in two cropped images which is enabled when the safe margin is wide. This, in turn, raises the detection probability. In addition, by increasing the authorized number of searches, the probability of losing the number plate will decrease while the processing time per image will increase.

Estimation of low-resolution number plates

Upon finding each car (or part of it) by the above process, the following algorithm is executed for finding a low- resolution version of the number plate:

1) Applying Sobel filter for detecting the edges.

2) Applying closing morphology operation to the edges of image with rectangular structural elements within the (2,10) size, this image is called closed image.

3) Obtaining a differentiated (edge) image by finding the difference between the "edge images" and "closed images". This step eliminates most of the non-number plate edges. The detection rate is increased with reduced false alarms.

4) Identifying the most populated region in terms of edges. For each zone L_1-*norm* is calculated.

$$L_p\text{ - }norm : \| X \|_p = \left(\sum_{i=1}^{N} \sum_{j=1}^{M} |x(i,j)|^p \right)^{\frac{1}{p}} \qquad (15)$$

where $X = x(i, j)$ is the input $N \times M$ matrix and L_1-*norm* can be achieved by setting p=1.

Note that L_1-*norm* calculates the sum of neighborhood pixels and it is used to exploit the sparse variation of edges in the LP to highlight the pattern of numbers and characters within the plate region. Figure 10 shows the comparison of using L_1-*norm* and closing morphology operation. As shown in this figure, L_1-*norm* leads to sparsity of edge image and we can find the aggregation of edges and LP region. Comparing to closing technique, when L_1-*norm* method is used, LP is fairly extracted.

5) Changing the output of L_1-*norm* to binary, using a threshold defined by Otsus method [2] (0.6× peak values). This binary image is called binaryL1.

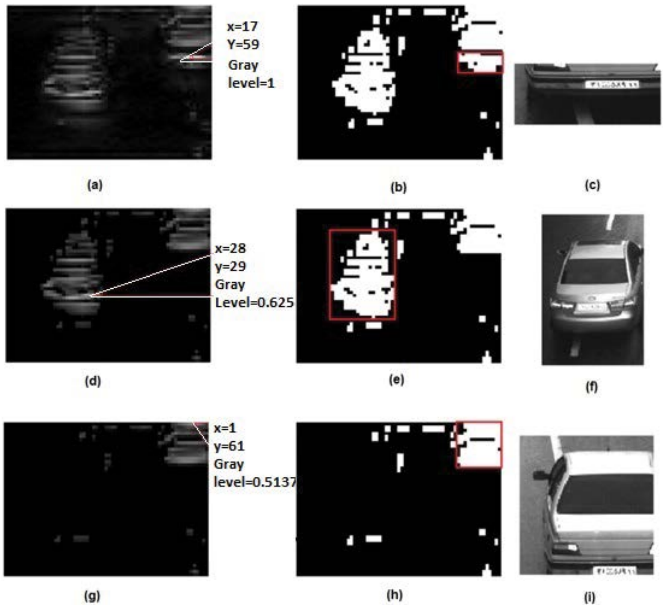

Figure 9: The outcomes of applying SB-ALPR for localization of the car in Figure 4(a).

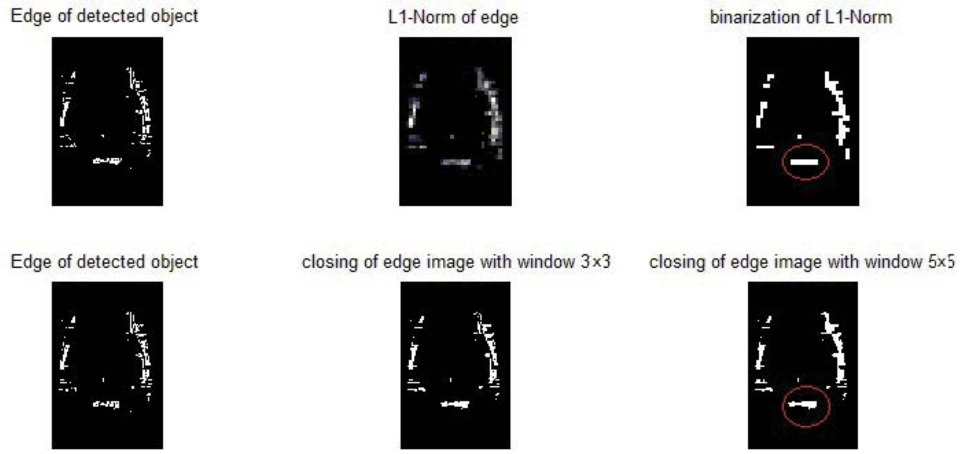

Figure 10: Comparison of using L_1- *norm* and closing morphology operation.

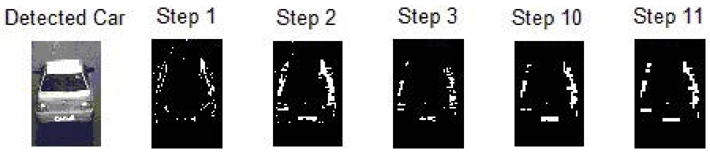

Figure 11: Output example of step 1, 2, 3, 10 and 11.

6) Applying closing operation to binaryL1 image containing line structural elements of size (1, 4). This image is called closedL1.

7) Processing the CC by beginning from the largest CC. If the width to height ratio is above 2, consider this region as the new candidate for the CC. In order to process this CC, the corresponding pixels are cropped from the gray image and then, steps 8 to 15 are applied. Otherwise, stop.

8) Obtaining the "edge image" by applying a 3 × 3 Sobel mask to the image and changing it to binary.

9) Identifying the zone boundaries by calculating L_1-*norm* to highlight the region of interest.

10) Changing the result to binary. This is called BinaryCCL1.

11) Configuring a "closedCCL1 image" by closing the binaryCCL1 image with line structural element of size (1,4).

12) Processing the largest CC and measuring its height to width ratio. If it is not larger than two, proceed to the next CC. In order to process these CCs, the corresponding pixels of gray-scale image are cropped and called: GrayCC

13) Converting the GrayCC image into binary and projecting it onto a column vector ProjCC.

14) Converting "ProjCC" vector to binary using a threshold of 0.75 × *max* (ProjCC). This is called: "BinaryProj". The value of the threshold 0.75 × *max* was chosen experimentally.

15) Obtaining the number plate image by the number of transitions between 0 and 1, which for Persian plates this should be equal or greater than 12. The corresponding pixels are cropped from gray-scale image into what is regarded as the number plate image.

Figure 11 shows some output example of estimation of this stage and Figure 12 shows the algorithm output for low- resolution number plate in combination with Figure 9 (i, c, f).

Accurate estimation of the plate number area

After localization of the low-resolution number plate, it is necessary to deal with the overlapping pixels to avoid the duplication of the number plates. For instance, in cases where the pixels of a number plate are overlapping with those of another number plate, the two plates are selected and combined into a new number plate with a larger coordination system. For obtaining the positions of the merged (tight) number plate, a low-resolution localization procedure should be applied in the following order:

- Apply 3 × 3 Sobel filter to detect the image edges.
- Highlight the detected edge pattern and calculate the level of L_1-*norm* for each zone.
- Binarize the output map of L_1-*norm* with a threshold level of 0.1. This binary image is called: BinaryTightL. Note that, in order to increase the sensitivity of the binarization, the threshold level was selected as a small value.
- Introduce the largest CC covering the tight plate number and crop the corresponding pixels from the gray-scale image.

Completing the above steps provides the location of car number plates and completes the algorithm. Figure 13 shows the results of the SB-ALPR algorithm for a low-resolution output image. As a result, the output of Figures 12 (a) and (b) are refined as (c) and (d).

Comparison of the proposed method with other methods

Very recently, a small number of works have been published in using saliency detection for ALPR research, such as the works carried out sby Gu et al. [31] and Lin et al. [32]. Using the algorithm in Ref.

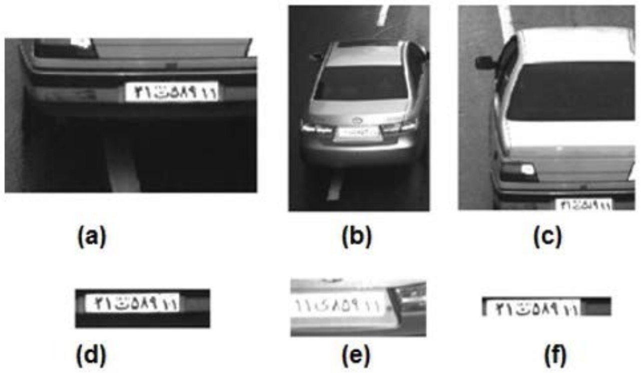

Figure 12: The results after each step of the estimation of low resolution number plates algorithm applied to Figure 9(a).

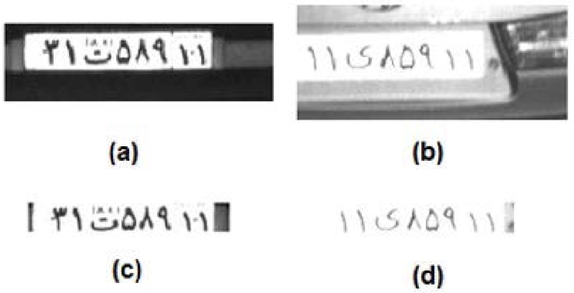

Figure 13: Output of the refined localization algorithm.

[31], the images are sub-sampled to build a 9 level image pyramid which incorporates 6 intensity, 12 color and 12 texture features by Gabor filtering and discrete cosine transform (DCT) [28]. In the binarization step, an adaptive threshold is used. Then, the regions that match the shape of a license plate are detected. SVM classifier is then utilized to eliminate the false license plates. The number of features in Ref. [31] is thirty, which makes their estimation time-consuming, and the classifier is very likely to be over-fitted. A similar approach Lin et al. [32] filters the image using difference of Gaussian (DoG). An on-off and an off-on saliency detection procedure is applied to detect white characters in darker and also dark characters in whiter backgrounds. This method uses a more robust technique based on statistical properties of the plate patterns such as entropy and likelihood. Unfortunately, convolving the image with the DoG and estimating the probabilistic features make it too slow for real-time applications. On the other hand, in both above algorithms, the saliency map is estimated for the detection of the plate directly. Therefore, for the cases of multiple plates and complex backgrounds, the detection error might be large. In our proposed algorithm, we use two orientation features and calculate the saliency based on down-sampled images which enhance the speed. More importantly, in our proposed two-stage method, we first find the car object and then the plate number. Table 2 shows the comparison of SB-ALPR method with Ref. [31] and Ref. [32].

Experimental Results

In order to study the accuracy of the SB-ALPR algorithm, since this method is based on still-image, we applied it to 660 various quality images. As shown in Figure 14, this database consists of IR (Inrfa-red) and color images both in simple and complex backgrounds.

Table 3 shows the descriptions of four datasets for the evaluation. All images are of 600 × 800 pixels in size. In IR images in the first dataset (Set 1), each image contains one car. Applying the SB-ALPR to this dataset, the accuracy rate of localization for this dataset was 98:95% and the average execution time was 40 ms.

The second dataset (Set 2) is from the IR images dataset which has complex background. Most of these images include more than one car. The complex backgrounds often consist of trees, pedestrians and sidewalk patterns. The localization accuracy for this dataset was 97.06% and the average execution time was 65 ms. Each color image with simple background in our third dataset (Set 3) contains one car. This dataset was also experimentally tested with the localization rate of 94.50% and the average execution time of 45 ms.

The fourth dataset (Set 4) consists of the most difficult- to-process and challenging images selected to illustrate the performance of the algorithm presented in this paper. This dataset contains color images with complex background with reflecting lights. While applying the SB-ALPR to these im- ages, the localization accuracy was 92.33% with an average execution time of 80 ms. In general, the average localization rate (accuracy) for the above four datasets is 94.77% and the average processing time is 61.95 ms. Figure 15 shows the experimental results after each step of SB-ALPR using different datasets.

Approach	Detection Rate	Computation Rate (ms)
Gu et al. [11]	92%	103
Lin et al. [15]	87%	112
SB-ALPR algorithm	97.06%	65

Table 2: Comparison of SB-ALPR method with other methods using images in dataset 2 (see Table 3).

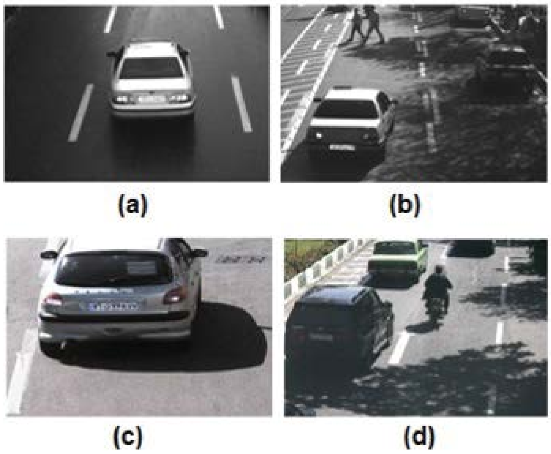

Figure 14: Examples of IR and color images in sample datasets. (a) IR image in simple background, (b) IR image in complex background, (c) color image in simple background, and (d) color image in complex background.

Images	Description	Ave. size (pixel)	No. of Images	No. of cars
Set 1	Infrared (IR) with sample background	600 × 800	90	95
Set 2	Infrared (IR) with complex background	600 × 800	150	170
Set 3	Color with sample background	600 × 800	180	200
Set 4	Color with Complex background	600 × 800	240	300

Table 3: Descriptions of four sample datasets.

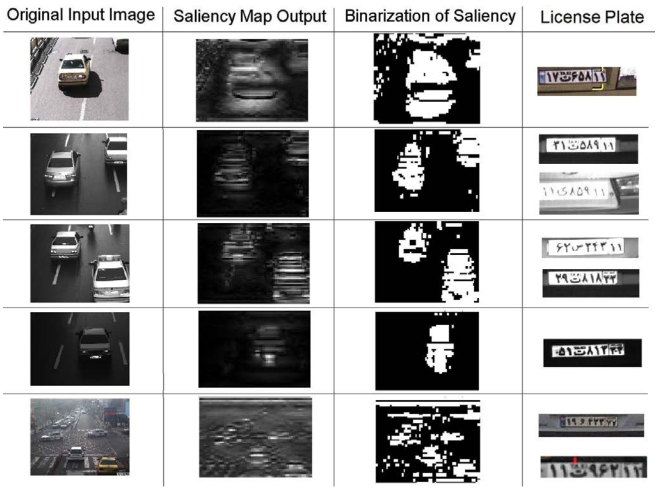

Figure 15: The results after each step of SB-ALPR using simple and complex datasets.

Images	Localisation rate	Ave. time/image (ms)	False detection
Set 1	94/95 (98.95)%	40	0.5%
Set 2	165/170 (97.06)%	65	1%
Set 3	189/200 (94.50)%	45	2%
Set 4	277/300 (92.33)%	80	2.5%
Overall	725/765 (94.77)%	61.89	1.5%

Table 4: Evaluation of the proposed method using the four datasets. Here MATLAB 8.2.0 (2013b), Core i7, and clock frequency of 2.70 GHz have been used.

Based on the results in Table 4, it seems that the execution time for complex backgrounds is longer in comparison to simple backgrounds. This can also be because of having more than one car in most of those images. Considering this factor, the processing time per number plate is actually shorter and more consistent in overall.

Summary and Conclusion

Looking at the evolution of ALPR systems over decades, there are two pronounced problems currently associated with these methods; the execution time associated with exhaustive search particularly for large high-resolution images and the accuracy of ALPR due to various technical and surrounding problems. In this paper, we highly mitigated these problems by improving our saliency-based technique [29] which was introduced for the first time in the area of ALPR, by introducing the directionality detection within the definition of saliency.

In addition, we benefited from the power of L_1*-norm* together with edge detection to better highlight the region of characters/ numbers. The SB-ALPR algorithm was tested using a reasonably large datasets with different complexities. The average accuracy of 94.77%, average execution time of 61.52 ms, and average processing time of 40.2 ms per plate show the significant improvement over traditional benchmark techniques. Nevertheless, the problem of separating the cars in close proximity will be among other possible problems to be solved in our future researches. For future work, we will investigate the impact of different environment and lighting conditions on the image acquisition, dealing with complex image background (tree, pedestrian, motorcycle existing in the image), the affects of the angle of rotation on the localization accuracy, and the challenges of dealing with multi-frame (video) and more complicated datasets to overcome the problem of tuning the system parameters.

References

1. Anagnostopoulos CNE, Anagnostopoulos IE, Psoroulas ID, Loumos V, Kayafas E, et al. (2008) License plate recognition from still images and video sequences: A survey. Intelligent Transportation Systems. IEEE Transactions 9: 377-391.
2. Chang SL, Chen LS, Chung YC, Chen SW (2004) Automatic license plate recognition. Intelligent Transportation Systems. IEEE Transactions 5: 42-53.
3. Barroso J, Dagless E, Rafael A, Bulas-Cruz J (1997) Number plate reading using computer vision. IEEE International Symposium 3: 761-766.
4. Yu M, Kim YD (2000) An approach to korean license plate recognition based on vertical edge matching. 2000 IEEE International Conference 4: 975-2980.
5. Xu J, Li S, Chen Z (2003) Color analysis for chinese car plate recognition. Intelligent Systems, and Signal Processing.
6. Wang YR, Lin WH, Horng SJ (2011) A sliding window technique for efficient license plate localization based on discrete wavelet transform. Expert Systems with Applications 38: 3142-3146.
7. Du S, Ibrahim M, Shehata M, Badawy W (2013) Automatic license plate recognition (alpr): A state-of-the-art review. Circuits and Systems for Video Technology. IEEE Transactions 23: 311-325.
8. Ashtari H, Nordin MJ, Fathy M (2014) An iranian license plate recognition system based on color features. Intelligent Transportation Systems. IEEE Transactions 15: 1690-1705.
9. Wen Y, Lu Y, Yan J, Zhou Z, Von Deneen KM, et al. (2011) An algorithm for license plate recognition applied to intelligent transporta- tion system. Intelligent Transportation Systems. IEEE Transactions 12: 830-845.
10. Wei-gang Z, Guo-Jiang H, Xing J (2002) A study of locating vehicle license plate based on color feature and mathematical morphology. In Signal Processing, 6th International Conference 1: 748-751.
11. Wu HHP, Chen HH, Wu RJ, Shen DF (2006) License plate extraction in low resolution video, In Pattern Recognition. 18th International Conference 1: 824-827.
12. Nguyen TT, Nguyen TT (2012) A real time license plate detection system based on boosting learning algorithm. 5th International Congress, pp: 819-823.
13. Viola P, Jones M (2001) Rapid object detection using a boosted cascade of simple features. In Computer Vision and Pattern Recognition. Proceedings of the 2001 IEEE Computer Society Conference 1: 1-511.
14. Ojala T, Pietikainen M, Maenpa T (2002) Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions 24: 971-987.
15. Hongliang B, Changping L (2004) A hybrid license plate extraction method based on edge statistics and morphology. 17th International Conference 2: 831-834.
16. Faradji F, Rezaie AH, Ziaratban M (2007) A morphological-based license plate location. IEEE International Conference 1: I-57.
17. Zheng D, Zhao Y, Wang J (2005) An efficient method of license plate location. Pattern Recognition Letters 26: 2431-2438.
18. Wang S, Lee HJ (2003) Detection and recognition of license plate characters with different appearances. Intelligent Transportation Systems 2: 979-984.
19. Soh YS, Chun BT, Yoon HS (1994) Design of real time vehicle identification system, Humans, Information and Technology. International Conference 3: 2147-2152.
20. Xu HK, Yu FH, Jiao JH, Song HS (2005) A new approach of the vehicle license plate location. 6th International Conference, pp: 1055-1057.
21. Zunino R, Rovetta S (2000) Vector quantization for license-plate location and image coding. Industrial Electronics. IEEE Transactions 47: 159-167.
22. Deb K, Chae HU, Jo KH (2009) Vehicle license plate detection method based on sliding concentric windows and histogram. Journal of computers 4: 771-777.
23. Parisi R, Di Claudio E, Lucarelli G, Orlandi G (1998) Car plate recognition by neural networks and image processing. Proceedings of the 1998 IEEE International Symposium 3: 195-198.
24. Matas J, Zimmermann K (2005) Unconstrained licence plate and text localization and recognition. Intelligent Transportation Systems Proceedings, pp: 225-230.

25. Abolghasemi V, Ahmayfard A (2009) An edge-based color-aided method for license plate detection. Image and Vision Computing 27: 1134-1142.

26. Kim KI, Jung K, Kim JH (2002) Color texture-based object detection: an application to license plate localization. pp: 293-309.

27. Gou C, Wang K, Yao Y, Li Z (2015) Vehicle license plate recognition based on extremal regions and restricted boltzmann machines. Intelligent Transportation Systems. IEEE Transactions 17: 1096-1107.

28. Itti L, Koch C, Niebur E (1998) A model of saliency-based visual attention for rapid scene analysis. IEEE Transactions on Pattern Analysis & Machine Intelligence 20:1254–1259.

29. Safaei Tang H, Sanei S (2015) Search-free license plate localization based on saliency and local variance estimation. International Society for Optics and Photonics, pp: 94451U.

30. Greenspan H, Belongie S, Goodman R, Perona P, Rakshit S, et al. (1994) Overcomplete steerable pyramid filters and rotation invariance. IEEE Computer Society Conference, pp: 222-228.

31. Gu B (2015) Visual attention saliency model for license plate location. International Journal of Signal Processing, Image Processing and Pattern Recognition 8: 267-274.

32. Lin KH, Tang H, Huang TS (2010) Robust license plate detection using image saliency. 17th IEEE International Conference, pp: 3945-3948.

Citation: Safaei A, Tang HL, Sanei S (2016) Robust Search-Free Car Number Plate Localization Incorporating Hierarchical Saliency. J Comput Sci Syst Biol 9: 093-103. doi:10.4172/jcsb.1000225