I.J. Image, Graphics and Signal Processing, **2015, 7, 1-8**
Published Online June 2015 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijigsp.2015.07.01

Application-Oriented Farsi ALPD Using Deterministic Edge Clustering

M. M. Zeinali
Computer Engineering Department, Islamic Azad University, South Tehran Branch, Tehran, Iran.
Email: st_m_zeinali@azad.ac.ir

S. Ghofrani
Electrical and Electronic Engineering Department, Islamic Azad University, South Tehran Branch, Tehran, Iran.
Email: s_ghofrani@azad.ac.ir

A. Sengur
Electrical and Electronic Engineering Department, Firat University, Technology Faculty, Elazig, Turkey.
Email: ksengur@firat.edu.tr

Address all correspondence to: Sedigheh Ghofrani, Mail address: No. 9, 14th street, Pakistan street, Shahid Beheshty street, postal code 15317-64611, Tehran, Iran. Tel: +98-21-88731446, Fax: +98-21-88532683.

***Abstract*—**In this paper a new application-oriented method for automatic Farsi license plate detection (ALPD), based on morphology and a modified edge clustering algorithm is proposed. Access control (AC), law enforcement (LE), and road patrol (RP) are mainly three applications for ALPD. After image enhancement by preprocessing, the edge statistics analysis and the morphology filter are used to decrease the search regions and remove the unwanted edges. Then the expectation-maximization (E-M) algorithm is used to estimate the corresponding Gaussian components for edges of remained regions. In this way the results of edge clustering and Gaussian components estimation are deterministic whereas the processing time in comparison with similar approaches in literature, is decreased significantly. Candidate regions are obtained by applying application-oriented thresholds to the properties of estimated Gaussian components. Finally for each candidate region, the maximally stable extremal region (MSER) detector is used to detect character-like regions and then select the region(s) of interest containing license plates. The proposed method is evaluated by using a database which includes images for the three groups AC, LE and RP applications, whereas some images suffer of being low quality, low contrast and blur and some images have complex background through existing multiple license plates. The experimental results show that our proposed method is reliable for images of different quality and illumination condition and it is able to detect the rotated and skewed license plates even in images containing multiple license plates and complex backgrounds.

***Keywords*—**License plate detection, license plate character segmentation, application oriented, edge clustering.

I. Introduction

The automatic license plate recognition (ALPR) has an important role in some real-life applications such as automatic toll collection, traffic law enforcement, searching stolen cars and road traffic monitoring. In fact, using an ALPR system, a license plate is extracted from image captured by a color, black and white or infrared cameras. The scopes of license plate rotation, aspect ratio and size in an image, as variables, are different for anonymous applications. There are some challenges for ALPR systems arising from license plate types and environmental conditions diversity. For example, diversity of license plate background and character colors, fonts and languages, irregularity of number, type and location of characters in some countries and states, covering license plate by dust and pollution, rotation of license plate, different illumination conditions and existence of license plate-like textures in an image background, are some of these challenges.

In general ALPR systems include three modules named license plate detection (LPD), character segmentation and character recognition. The LPD module has the important role of system performance. The aim of LPD module is to decrease searching regions in a way that the system only processes regions having license plate features. License plate features are usually extracted from its format and constituting characters.

An ALPR method based on statistical analyzing and vertical edges was proposed in [1]. The algorithm named VEDA and it was 7 to 9 times faster than the Sobel operator which is known as the classic edge detector. The main challenge for implementing the VEDA, is finding an appropriate binarization algorithm. In addition, vertical edges may not be detected properly when the license plate region is small. The block based method was proposed in [2] where blocks with high edge density are assumed to be the license plate region. The main difficulty of block-based algorithms is finding many candidate regions especially for images with complex background. Although methods based on edge statistics, are simple and fast but the edge connectivity is necessary. However, the unwanted edges are removed and the license plate region is extracted by using the morphology filter. For example in [3]-[5], the connected component analysis and measuring features such as region area and aspect ratio were used.

Besides the edge statistical features, many literatures used the textural features. In [6], the license plate characters were detected by using scan-line technique and counting peaks in gray scale level changes. In [7]-[9], the edge density was computed and the regions with most density based on a weight allocation schema were segmented. In [10], a license plate roughly and then precisely was segmented by detecting the vertical edges and using the mathematical morphology filtering.

In this paper we proposed a new application-oriented method for Farsi LPD based on a modified edge clustering and morphology algorithm. The LPD applications are divided into three general categories named AC, LE and RP. A modified edge clustering algorithm with the help of edge statistical analysis and morphology is used for obtaining the candidate regions. The real license plate region is identified using MSER detector [11] and counting connected components. In order to evaluate our proposed algorithm, a Farsi license plate database including 390 images of AC, LE and RP applications is used.

The rest of this paper is organized as follows. In Section II, we give a short review on application-oriented license plate recognition method. In Section III, we explain our proposed method at first including preprocessing in order to improve the input image, removing unwanted regions or regions with low probability of being a license plate, estimating Gaussian components, obtaining candidate regions and determining the license plate region among candidates, then our experimental results on database containing 390 images are reported. Finally the conclusions are given in Section IV.

II. Review The Application-Oriented License Plate Recognition

The application-oriented license plate recognition method based on edge clustering was proposed for AC, LE and RP applications [12]. As an example, we can refer to the entrance gate monitoring for AC, road monitoring for LE and searching stolen cars for RP applications. From AC to RP applications, the variability scopes become wider. It means that many regions are candidate and therefore more time needs for processing. In [12], at first, the vertical Sobel edge detector was used for a gray scale image. Then assuming the Gaussian mixture model (GMM) for edge point's distribution, the E-M algorithm was used for estimating the Gaussian components. The GMM is expressed as:

$$p(x_i) = \sum_{j=1}^{N_c} \omega_j \phi(x_i \mid \mu_j, \mathbf{V}_j) \qquad (1)$$

where x_i refers to an edge point, $p(x_i)$ is the probability of the edge point, N_c is the pre-determined number of components, ω_j is the weighting factor and ϕ is the *j-th* Gaussian component with mean μ_j and covariance $\mathbf{V}_j$.

The E-M algorithm takes some mean, covariance matrix and weight values, as primary states of components and iteratively estimates new values for them until the convergence condition is satisfied. For preparing these values, first the edge points are clustered into the pre-determined clusters number (equal to components number) by using k-means algorithm. Then, the mean and the covariance of each created cluster are calculated as,

$$\begin{aligned} \mu_j &= \frac{1}{N}\sum_{i=1}^{N} x_{ij} \\ \mathbf{V}_j &= \frac{1}{N-1}\sum_{i=1}^{N} (x_{ij}-\mu_j)(x_{ij}-\mu_j)^T \end{aligned} \qquad (2)$$

where x_{ij} denotes the edge point *i* in cluster *j*, *N* is the number of edge points in cluster *j* and $(.)^T$ denotes the transpose. Also the weighting factor ω_j is obtained as the number of edge points in *j-th* cluster to total number of edge point ratio whereas,

$$\sum_{j=1}^{N_c} \omega_j = 1 \qquad (3)$$

For iteration *m* and for $j = 1,\ldots,N_c$, the E and M steps of E-M algorithm are as follows:

- In E-step compute $n_j^{(m)} = \sum_{i=1}^{N} \gamma_{ij}^{(m)}$ where

$$\gamma_{ij}^{(m)} = \frac{\omega_j^{(m)} \varphi(x_i \mid \mu_j^{(m)}, \mathbf{V}_j^{(m)})}{\sum_{l=1}^{N_c} \omega_l^{(m)} \varphi(x_i \mid \mu_l^{(m)}, \mathbf{V}_l^{(m)})}, \qquad i = 1,\ldots,N \qquad (4)$$

- In M-step compute the new estimates,

$$\begin{aligned} &\omega_j^{(m+1)} = n_j^{(m)} / N, \mu_j^{(m+1)} = \sum_{i=1}^{N} \gamma_{ij}^{(m)} x_i / n_j^{(m)} \\ &\mathbf{V}_j^{(m+1)} = \sum_{i=1}^{N} \gamma_{ij}^{(m)} (x_i - \mu_j^{(m+1)})(x_i - \mu_j^{(m+1)})^T / n_j^{(m)} \end{aligned} \qquad (5)$$

For a pre-determined threshold value α , the algorithm terminates if $\left| l^{(m+1)} - l^{(m)} \right| < \alpha$ where $l^{(m+1)}$ is obtained as follows:

$$l^{(m+1)} = \frac{1}{N}\sum_{i=1}^{N}\log(\sum_{j=1}^{N_c}\omega_j^{(m+1)}\phi(x_i \mid \mu_j^{(m+1)}, \mathbf{V}_j^{(m+1)})) \quad (6)$$

The squares of eigen values and eigen vectors of an estimated component covariance matrix, indicate the lengths and orientations of major and minor diagonals of an ellipse corresponding to the component, respectively. So corresponding ellipse of each component can demonstrate features such as width, aspect ratio and orientation and we can use these properties for measuring likeness to a license plate. By applying application-oriented thresholds as scopes of variables in each application to these properties, undesired components are removed. All the above are the following inequalities for measuring the ellipse major axis length to image width ratio, ellipse aspect ratio and ellipse orientation, respectively:

$$\begin{aligned} & s_m < \frac{\lambda_{i,M}}{W_I} < s_M \\ & r_\lambda < \frac{\lambda_{i,M}}{\lambda_{i,m}} < R_\lambda \\ & \theta_m < \tan^{-1}(\frac{\mathbf{v}_{i,M}}{\mathbf{v}_{i,m}}) < \theta_M \end{aligned} \quad (7)$$

where W_I is the image width , $\lambda_{i,m}$ and $\lambda_{i,M}$ are the smallest and the largest eigen values of covariance matrix $\mathbf{V}_j$ with associated eigen vectors $\mathbf{v}_{i,m}$ and $\mathbf{v}_{i,M}$. The minimum and maximum application-oriented thresholds are $[s_m, r_{\lambda,}\theta_m]$ and $[s_M, R_{\lambda,}\theta_M]$.

An iterative procedure in three steps containing k-means algorithm, E-M algorithm and removing undesired components, with different number of clusters and iterations for each application, were used in [12]. For example for AC, the iteration numbers considered 6 and the number of clusters varied from 6 to 10. It is shown that the license plate region is stably detected by Gaussian components in most iterations.

Due to non-deterministic behavior of k-means and E-M algorithms we may get different results for license plate region candidates whenever we run the algorithm. In this paper, in order to have a deterministic behavior we used the modified edge clustering and Gaussian component estimation by using edge statistics and morphology.

III. Application-Oriented Farsi ALPD

Our proposed method, in general consists of four steps including preprocessing for improving input image, removing unwanted regions (regions with low probability for being a license plate), estimating Gaussian components for remained edges and obtaining candidate regions and finally determining license plate region among candidates. The block diagram of the method is shown in Fig. 1 and explained in detail in following.

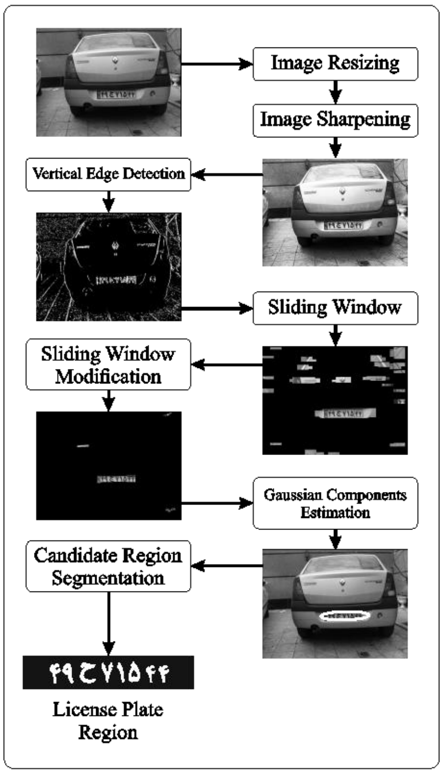

Fig.1. Block diagram of the proposed method.

A. Image Resizing

Although the proposed method is working perfect irrespective of an input image size, resizing image to 640×480 or 480×640 for portrait and landscape, is convenient because of computation cost. Obviously the aspect ratio is either 4:3 or 3:4. Now for any arbitrary input image we resize the image whereas the aspect ratio would be the same as the input image and the image size is the nearest to n= 640×480 pixels,

$$x = \text{round}(\text{sqrt}(\frac{n}{ar})) \quad (8)$$

$$y = \text{round}(\text{sqrt}(n \times ar)) \quad (9)$$

where $x \times y$ is the size or number of pixels of the resized or output image, and ar refers to the pre-determined aspect ratio; i.e. for 4:3 and 3:4 images $ar = 4/3$ and $ar = 3/4$ respectively.

B. Image Sharpening

Image sharpening as preprocessing is necessary due to low contrast input image, unclean license plate and sun light reflection. For this purpose we used a mask as a high pass filter:

$$\begin{bmatrix} C & C & C \\ C & W & C \\ C & C & C \end{bmatrix}$$

The mask coefficients are obtained as follows:

$$W = 9A - 1 \tag{10}$$

$$C = -\left(\frac{A}{2}\right) \tag{11}$$

where $A = 1 + (1 - \sigma)$ and σ is the standard deviation of the resized grayscale image. After applying the high pass filter, the image is rescaled into range [0, 1] and enhanced by histogram equalization. In Fig. 2, three original images of AC, LE and RP applications and their corresponding sharpened images are shown.

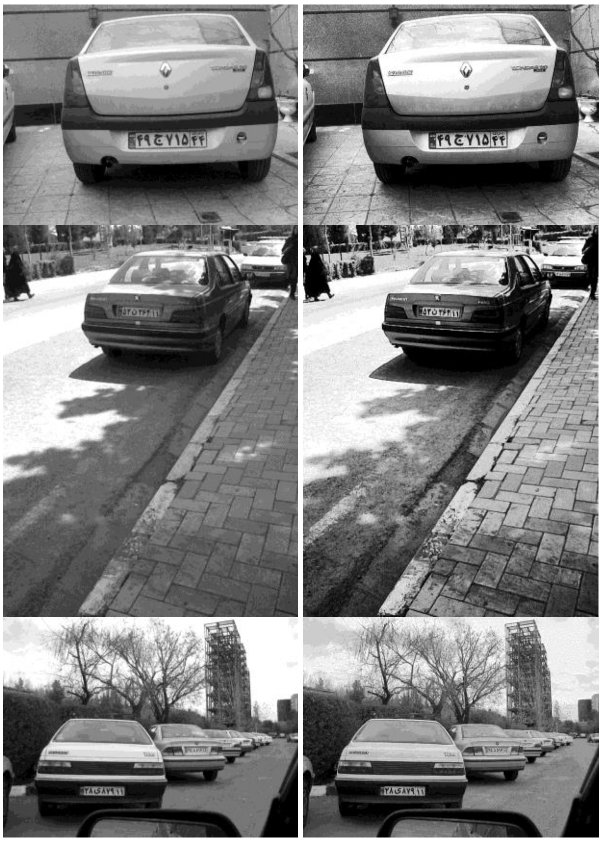

Fig.2. Three sample original (left) and sharpened (right) images for AC, LE and RP applications (top to bottom).

C. Vertical Edge Detection

In this stage, the Sobel mask, $[-1\ 0\ 1;\ -2\ 0\ 2;\ -1\ 0\ 1]$, is used as a vertical edge detector. The obtained values of applying Sobel operator for different pixels may be positive or negative. The positive values indicate moving from a dark intensity level to a light one whereas the negative values indicate a reverse moving. Since both edge types are important, we consider the absolute value of output. The result of applying Sobel operator is enhanced by using histogram equalization. Although many edges are detected, the strong edge points should be kept. For this purpose, a global threshold:

$$Th = \text{Max}(\mathbf{E}) - \left(\frac{m_{\mathbf{E}}\sigma_{\mathbf{E}}}{2}\right) \tag{12}$$

is used where $\mathbf{E}$ refers to an edge image, $m_{\mathbf{E}}$ and $\sigma_{\mathbf{E}}$ are the mean and the standard deviation. Obtaining the threshold value, we have:

$$\hat{\mathbf{E}}(i,j) = \begin{cases} \mathbf{E}(i,j) & ; \quad \mathbf{E}(i,j) \geq Th \\ 0 & ; \quad \mathbf{E}(i,j) < Th \end{cases} \tag{13}$$

Fig. 3 shows the edge images, $\mathbf{E}$ and $\hat{\mathbf{E}}$.

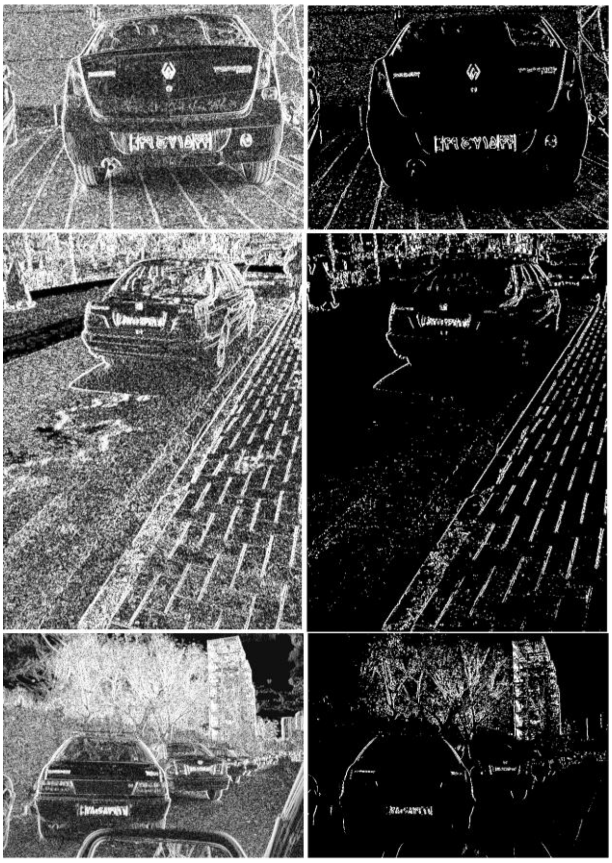

Fig. 3. Shown the output of Sobel vertical edge detector without using a threshold (left-column) and with using the threshold value based on Eqs. (12)-(13) (right-column).

D. Sliding Window

Sliding window procedures of an edge output image based on the edge density is explained in following:

1. Consider a primary window size.
2. Move the window horizontally and then vertically on the edge output image and compute the edge density value function for every location according to edge point values of window.
3. Keep the window content if its edge density value satisfies the pre-defined range.
4. Increase the window size and then repeat steps 2-4 if the window width is smaller than image width named *'y'* in Eq. (9).

For each window size, for reducing the processing, we move the window in horizontal and vertical directions with steps equal to 33% of horizontal and vertical widow dimensions respectively. As the standard Farsi license plate is 10×50 centimeters, the window size for every iteration, $t = 1, 2, \ldots \lfloor y/50 \rfloor$ is considered to be:

$$window - size_t = 10t \times 50t \qquad (14)$$

In addition, at each iteration and for every window size equal to $p \times q$, the edge density is obtained as:

$$edge - density = \frac{1}{p \times q} \sum_i \sum_j \hat{\mathbf{E}}(i, j) \qquad (15)$$

During window moving, the desired regions are identified if the edge density value be in range [0.22, 0.35] according to [13].

E. Sliding Window Modification

The aim of this step is partially extracting some parts of remained regions in which there are character-like textures. As it is seen in Fig. 4, the output of sliding window procedure contains some connected components. We apply the sliding window modification to each connected component separately. The procedures of modification for every connected component are as follows:

1. Count active edge points in $\hat{\mathbf{E}}$ for every row and column of connected component and obtain two vectors named $\mathbf{sum_r}$ and $\mathbf{sum_c}$.
2. Apply closing and opening operators by using a rectangular structure of values 1 and size,

$$\begin{aligned} s_x &= \text{round}(0.3 \times \overline{\mathbf{sum_c}}) \\ s_y &= \text{round}(0.6 \times \overline{\mathbf{sum_r}}) \end{aligned} \qquad (16)$$

   where $\overline{\mathbf{sum_c}}$ and $\overline{\mathbf{sum_r}}$ refers to the mean value of vectors $\mathbf{sum_r}$ and $\mathbf{sum_c}$, to grayscale image (before sharpening), and then subtract the output images of opening and closing morphology filters.
3. Binarize the subtraction image using its mean value as a global threshold. A point with a value greater than threshold is considered as a point of a character-like region.

It is seen in Fig. 4 that like sliding window procedure, at the end of sliding window modification procedure, some connected components are found. By analyzing the connected components we discard some candidate regions based on the standard plate size and aspect ratio. Supposing the worst case that the license plate occupies 10×50 pixels (one pixel per square centimeter) in an image and because the characters occupy about 55% of license plate total area, we choose minimum allowed area for a connected component, equal to 220 (44% of 10×50) pixels and discard components with smaller area. Also we discard components with an area greater than one sixth of total image area. In addition, connected components with aspect ratio out of a specific range, is not a license plate and must be removed. According to [13], the aspect ratio range is [1.8, 13.7]. The results of sliding window and sliding window modification steps on sample images are shown in Fig. 4.

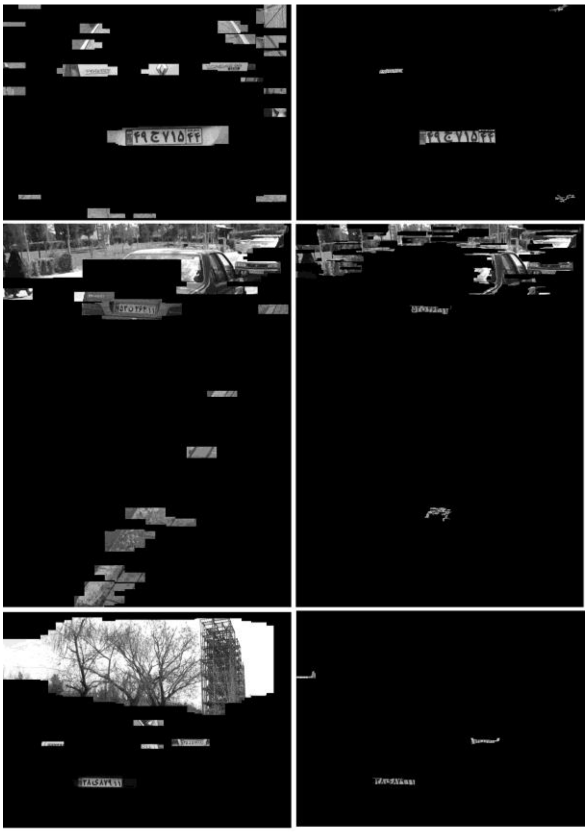

Fig.4. Remained regions after using sliding window (left-column) and sliding window modification (right-column).

F. Gaussian Components Estimation

In this section, we estimate a Gaussian component for edge points of each connected component. In this way, a candidate region is kept if it satisfies the properties of being a license plate, otherwise is would be removed. We used the E-M algorithm for Gaussian component estimation in a different way than the method in [12]. Therefore, instead of using k-means algorithm for clustering the edge points, we consider the edge points in each connected component as a cluster. So the result of E-M algorithm is deterministic whereas no iteration is needed for different number of clusters and the processing time is decreased significantly. Also the effect of non-deterministic behavior of k-means algorithm in edge points clustering is eliminated. For discarding unlikely components, we apply some application-oriented thresholds according to Eq. (7), for features of Gaussian components, listed:

1. Ellipse major axis length to image width ratio
2. Ellipse aspect ratio
3. Ellipse orientation (major axis angel relative to horizontal axis)

If one of these properties of component is out of the chosen range, the component is omitted. The experimentally chosen ranges for each application are:

$$\text{AC: } [s_m, s_M] = [0.085, 0.25]\ [r_\lambda, R_\lambda] = [4, 9]$$
$$[\theta_m, \theta_M] = [-13^\circ, 13^\circ] \tag{17}$$

$$\text{LE: } [s_m, s_M] = [0.05, 0.22]\ [r_\lambda, R_\lambda] = [3.3, 9.25]$$
$$[\theta_m, \theta_M] = [-22^\circ, 22^\circ] \tag{18}$$

$$\text{RP: } [s_m, s_M] = [0.035, 0.27]\ [r_\lambda, R_\lambda] = [3.34, 11.1]$$
$$[\theta_m, \theta_M] = [-30^\circ, 30^\circ] \tag{19}$$

where [s_m, r_λ, θ_m] and [s_M, R_λ, θ_M] are application-oriented minimum and maximum allowed threshold values for three above mentioned properties. In Fig. 5, results of estimating Gaussian components after discarding undesired ones are shown. We will consider the horizontal surrounding rectangle of each ellipse on original grayscale image (the image before sharpening) as a candidate region.

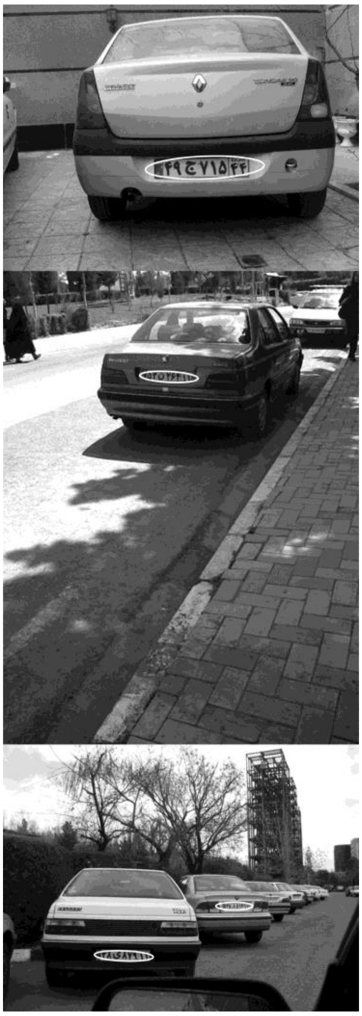

Fig.5. Shown the estimated Gaussian components (white ellipses) after discarding undesired ones for AC (top), LE (middle) and RP (bottom) applications.

G. Candidate Regions Segmentation

For determining license plate region(s) among candidate regions, we use MSER detector [11] to segment candidate regions. The MSER detector considers all possible threshold values for binarizing an image. In general, a MSER region is a connected region in an image with little changes in its size or area for a range of thresholds.

Although there are 11 types of Farsi license plates, in this paper, we considered 3 types including private, public and state as samples shown in Fig. 6. There are seven numeric and one alphabetic Farsi characters in these license plates. In addition as it is seen in Fig. 6, both private and public license plates have dark characters and light background, whereas it is vice versa for the state license plate. We use the number of numeric characters as a criterion for indentifying license plate region(s).

Before processing by MSER detector, the intensity of candidate regions is rescaled for having a range [0, 1]. The output of MSER detector is some connected components in candidate region as MSER regions. According to our experiments, in each candidate region, the ratio of character areas to corresponding surrounding rectangle of Gaussian component area is in a specific range, it is [0.0075, 0.1]. So the connected components out of the range ration are discarded. With calculating mean and covariance for points of a connected component according to Eq. (2), we obtain an ellipse in the same way that was used for Gaussian components. In a standard Farsi license plate, the ellipses of numeric characters have some similar properties. We use these properties for indentifying numeric characters in a candidate region, by applying some thresholds that describe them. For each ellipse of region, the thresholds are applied for properties listed below:

1. The ratio of ellipse major axis to Gaussian component ellipse minor axis length
2. Ellipse aspect ratio (the ratio of major axis to minor axis length)
3. The ellipse orientation (major axis angle relative to horizontal axis)

According to our experiments, the chosen threshold ranges for the above first, second and third defined items, in order are:

$$\begin{aligned} &1:\ [s_e, s_E] = [0.445, 1] \\ &2:\ [r_e, r_E] = [1.43, 9] \\ &3:\ [\theta_e, \theta_E] = [50^\circ, 140^\circ] \end{aligned} \tag{20}$$

The license plate region of AC sample image with white ellipses MSER regions is shown in Fig. 7 where regions with areas out of above mentioned range, have been discarded. Also for classification, we show the AC sample image and the guide indicators in Fig. 8.

After discarding regions with out of range properties, a binary image is constructed in which pixels corresponding to all remained regions are set to 1 and others have value 0. The aim of applying threshold ranges to MSER region ellipses originally is to identify numeric character regions, so we consider the number of numeric characters (seven) in a license plate as a criterion for identification. We count the number of connected components in the constructed binary image and if the number of them is equal or greater than 7, the candidate region is considered as a license plate region. The results for sample images are shown in Fig. 9. It is seen in Fig. 9 that in two binary images, the alphabetic character is extracted by MSER detector because its properties are like numeric characters. Also it is obvious that we can use the results of MSER detector as results of a character segmentation module for next usage.

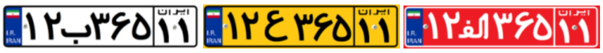

Fig.6. Three types of Farsi license plates. They are private, public and state from left to right.

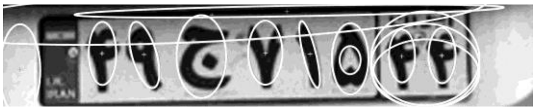

Fig.7. MSER region ellipses after discarding regions with out of range areas and before applying thresholds in Eq. 20.

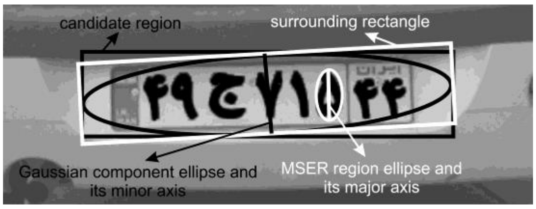
candidate region
surrounding rectangle
Gaussian component ellipse and its minor axis
MSER region ellipse and its major axis

Fig.8. Guide indicators on the AC sample image for better understanding of subjects.

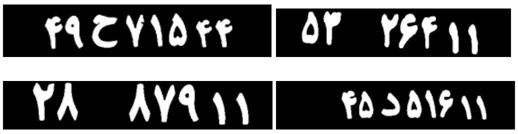

Fig.9. Binary license plate segmented region(s) of AC (top-left), LE (top-right) and RP (bottom) applications.

H. *Experimental Results*

In order to evaluate our proposed method and show some sample output of different blocks, we used an Iranian license plate database including 390 images. We mention that the database includes some low contrast, low quality, and blurred images and furthermore some images with complex background contain multiple license plates. The images are grouped into AC, LE and RP applications where in order, they consist 81, 211 and 390 images. All images in AC and LE are duplicated in RP, because as previously mentioned, from AC to RP application, the variability scopes become wider. So RP covers other applications. The method is implemented using MathLab2012 on a PC with Core i7 CPU and 8GB of RAM. The obtained license plate detection accuracy of applying AC, LE and RP settings for three grouped images are shown in Fig. 10. As it is seen, RP setting, Eq. (19), works properly for the three group images; i.e. AC, LE and RP applications. In contrast, the average consumed time are 0.53, 0.49, 0.61 seconds in order for AC, LE and RP applications. Although the detection accuracy of RP setting is appropriate, it is time consuming in comparison with AC and LE settings. In addition, although according to [12], the average consumed time of our method is greater, note that the method in [12] was implemented in C++ environment. The experimental results for each application, based on image resolution are reported in Table 1. We have also compared our method with proposed methods in [12] (with totally 2049 images) and [13] (with totally 300 images) in Table 2. Note that [13] is not application-oriented. Although the accuracy of our proposed method is lower than [13] and [12] (only for AC), it is faster in processing because of more simplicity and no use iterative procedures. In addition, we saw the used database for [12] and we believe that our Farsi license plate database including more low contrast images.

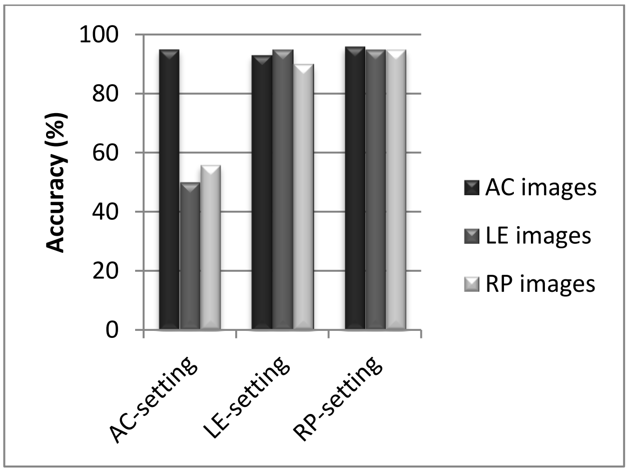

Fig.10. The accuracy of the proposed method with AC (left columns), LE (middle columns) and RP (right columns) settings on images of AC, LE and RP applications.

Table 1. Detection accuracy based on image resolution for AC, LE and RP applications.

	High resolution		Medium resolution		Low resolution	
	Images	Accuracy (%)	Images	Accuracy (%)	Images	Accuracy (%)
AC	68	97	11	91	2	50
LE	185	95	23	95	3	100
RP	339	96	50	92	7	85

Table 2. Comparing the achieved detection accuracy with others.

	Accuracy (%)		
Application	The proposed method	[12]	[13]
AC	95	96	97
LE	95	95	
RP	95	94	

IV. Conclusion

In this paper a reliable and robust method for application-oriented Farsi license plate detection was proposed. The proposed method uses morphology for reducing search regions and a modified edge clustering algorithm for estimating Gaussian components for edge points detected by Sobel vertical edge operator. Using modified edge clustering and Gaussian component estimation algorithms, we eliminate the effect of non-deterministic behavior of k-means and E-M algorithms. Also compared to similar methods in [12] and [13] our method is faster because of more simplicity and no use iterative procedures. It is shown that the combination of morphology and modified edge clustering algorithm can improve the accuracy of determining candidate regions while the processing time reduces significantly. The method is robust against license plate rotation, skew and multiplicity. The proposed method has been tested on an application-oriented classified image database and the experimental results show that the average accuracy is appropriate. Although the proposed method is designed for Farsi license plates, it can be modified and used for other license plates as well. Also, implementing the proposed method using more efficient programming techniques and environments capable the algorithm for real time applications.

References

[1] A.M. Al-Ghaili, S. Mashohor, A. Ramli, and A. Ismail, "Vertical edges-based car license plate detection method," IEEE Transactions on Vehicular Technology, vol.62, pp.26-38, 2013.

[2] H.J. Lee, S.Y. Chen, and W.S. Zheng, "Extraction and recognition of license plates of motorcycles and vehicles on highways," IEEE International Conference on Pattern Recognition, vol.4, pp.356-359, 2004.

[3] B. Hongliang, and L. Changping, "A hybrid license plate extraction method based on edge statistics and morphology," IEEE International Conference on Pattern Recognition, vol.2, pp.831-834, 2004.

[4] N. Ballas, S.M. Chai, M. Dwyer, and D. Linzmeier, "FPGA implementation of a license plate recognition SoC using automatically generated streaming accelerators," IEEE International Conference on Parallel and Distributed Processing Symposium, 2006.

[5] H.P. Wu, H.H. Chen, and R.J. Wu, "License plate extraction in low resolution video," IEEE International Conference on Pattern Recognition, vol.1, pp.824-827, 2006.

[6] Y.S. Soh, B.T. Chun, and H.S. Yoon, "Design of real time vehicle identification system," IEEE International Conference on Humans, Information and Technology, vol.3, pp.2147-2152, 1994.

[7] V.S.L. Nathan, J. Ramkumar, and S.K. Priya, "New approaches for license plate recognition system," IEEE International Conference on Intelligent Sensing and Information Processing, pp.149-152, 2004.

[8] V. Seetharaman, A. Sathyakhala, and N.L.S. Vidhya, "License plate recognition system using hybrid neural networks," IEEE International Conference on Fuzzy Information, vol.1, pp.363-366, 2004.

[9] R. Bermananth, A. Chitra, and V. Seetharaman, "A robust video based license plate recognition system," IEEE International Conference on Intelligent Sensing and Information Processing, pp.175-180, 2005.

[10] C.T. Hsieh, Y.S. Juan, and K.M. Hung, "Multiple license plate detection for complex background," IEEE International Conferenceon Advanced Information Networking and Applications, vol.2, pp.389-392, 2005.

[11] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide-baseline stereo from maximally stable extremal regions," Image and Vision Computing, vol.22, no.10, pp.761-767, 2004.

[12] G.S. Hsu, J.C. Chen, and Y.Z. Chung, "Application-Oriented license plate recognition," IEEE Transactions on Vehicular Technology, vol.62, no.2, pp.552-561, 2013.

[13] M. Rasooli, S. Ghofrani, and E. Fatemizadeh, "Farsi license plate detection based on element analysis and characters recognition," International Journal of Signal Processing, Image Processing and Pattern Recognition, vol.4, no.1, 2011.

Authors' Profiles

Mohammad Mehdi Zeinali received the B.S degree in telecommunications engineering from Islamic Azad University, Shahre Rey branch, Tehran, Iran, in 2009 and the M.S degree in computer engineering from Islamic Azad University, Tehran-South branch, Tehran, Iran in 2014. His research interest is image processing and especially license plate recognition applications.

Sedigheh Ghofrani received her Ph.D. from Iran University of Science and Technology in 2004. She is associate professor since 2012. Her area of research includes image processing and signal processing. In 2003, she spent eight months at the School of Electronic and Electrical Engineering, the University of Leeds, UK, supported by British Council foundation. In 2012, she spent eight months at the Center for Advanced Communications (CAC) at Villanova University, PA, USA, as visiting research professor.

Abdulkadir Sengur obtained his Ph.D. degree from the Department of Electrical and Electronics Engineering at Firat University, Turkey, in 2006. He is an associated professor in the Technology Faculty, Department of Electrical and Electronics Engineering at Firat University. His research interests include pattern recognition, machine learning and image processing.