

# Accurate Detection and Recognition of Dirty Vehicle Plate Numbers for High-Speed Applications

Rahim Panahi, *Member, IEEE*, and Iman Gholampour, *Member, IEEE*

**Abstract**—This paper presents an online highly accurate system for automatic number plate recognition (ANPR) that can be used as a basis for many real-world ITS applications. The system is designed to deal with unclear vehicle plates, variations in weather and lighting conditions, different traffic situations, and high-speed vehicles. This paper addresses various issues by presenting proper hardware platforms along with real-time, robust, and innovative algorithms. We have collected huge and highly inclusive data sets of Persian license plates for evaluations, comparisons, and improvement of various involved algorithms. The data sets include images that were captured from cross-roads, streets, and highways, in day and night, various weather conditions, and different plate clarities. Over these data sets, our system achieves 98.7%, 99.2%, and 97.6% accuracies for plate detection, character segmentation, and plate recognition, respectively. The false alarm rate in plate detection is less than 0.5%. The overall accuracy on the dirty plates portion of our data sets is 91.4%. Our ANPR system has been installed in several locations and has been tested extensively for more than a year. The proposed algorithms for each part of the system are highly robust to lighting changes, size variations, plate clarity, and plate skewness. The system is also independent of the number of plates in captured images. This system has been also tested on three other Iranian data sets and has achieved 100% accuracy in both detection and recognition parts. To show that our ANPR is not language dependent, we have tested our system on available English plates data set and achieved 97% overall accuracy.

**Index Terms**—Digital signal processors, plate detection, automatic number plate recognition, RANSAC, support vector machine (SVM).

## I. INTRODUCTION

VEHICLE plate detection and recognition appear in vast variety of applications, including travel time estimation, car counting on highways, traffic violations detection, and surveillance applications [1]–[3]. Traffic monitoring cameras are mounted four to seven meters above the street level. Plate recognition range, where the cameras are able to capture the vehicles plates with sufficient resolution, starts from 20 to more than 50 meters away from the camera location. This range depends on the camera resolution and the lens mounted on the camera. At these heights and distances, vehicles plates are not as clearly visible as in other applications such as toll and parking fee payment systems. High camera installation point causes

Manuscript received October 28, 2015; revised March 18, 2016 and June 16, 2016; accepted June 23, 2016. The Associate Editor for this paper was L. M. Bergasa. (*Corresponding author: Iman Gholampour*)

The authors are with the Electronic Research Institute, Sharif University of Technology, Tehran 11365-11155, Iran (e-mail: panahi@mehr.sharif.edu; Imangh@sharif.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TITS.2016.2586520

some difficulties against the correct detection of vehicles plates. Vehicles with dirty plates make the situation even more complicated. On the other hand, number plate is the only trustworthy identity of a vehicle in Intelligent Transportation Systems (ITS) and correct vehicle identification depends highly on the accuracy of automatic number plate recognition (ANPR) systems.

This paper focuses on design, deployment and evaluation of a system for two industrial applications: detecting traffic violations at urban intersections and vehicle counting on highways. These two applications are among the most important ones in the ITS industry.

An ANPR system consists of three different modules: a) Monochrome/Color cameras, b) IR projector, and c) the processing board. In addition to compatibility of interfaces, each section must be chosen properly for a specific application. In this paper, a detailed exploration on the important parameters of an ANPR module has been done. Most of the key parameters are discussed and a comparison between final cost and the desired accuracy is shown.

A monochrome camera is usually used with an IR projector for plate detection and recognition. Monochrome camera sensors are capable of providing higher details and sensitivity compared to color camera sensors. An IR projector increases the contrast between plates characters and plates backgrounds. The IR projector is mostly useful at night to brighten the plates. We also need to synchronize the camera exposure time with the IR projector pulses in order to capture images with clearer plates. IR projector which operates invisibly, is an important alternative to the flash lights.

In many countries, a color image of the violation scene must be stored as an evidence. For such purposes, a color camera is located beside the monochrome camera. Color cameras can be deployed alone in places where controlled lighting conditions exist, e.g., in tunnels. Since a single camera is sufficient in these cases the final cost of the system is reduced. Such systems can also be employed in cases where violations are considered only in the daylight. For example, congestion charging systems need to detect the vehicles that enter a specific zone during specific hours.

Basically, the License Plate Recognition (LPR) process is divided into three main parts: Plate Detection, Character Segmentation, and Character Recognition. Each of these parts plays an important role in the final accuracy. Many problems such as size variations, viewing angle, low contrast plates, vehicles high speed and time consuming algorithms have prevented researchers from introducing a single class of algorithms to solve the problem. There have been, however, many algorithms proposed for each part.



Fig. 1. Results of applying the Sobel edge detector on (a) a clean plate and (b) a dirty plate.



Fig. 2. Results of applying the Hough transform on clean and dirty plates. (a) Successful detection (white rectangle). (b) Failed detection.

For plate detection, several algorithms have been proposed. Some of these algorithms are based on finding image edges, such as horizontal and vertical edges [4], [5]. For example, in [6], plates are localized using the Canny edge detector. Sobel operator is used by some other methods that work based on detecting image edges. These methods have two main advantages: smoothing the image noise because of the included averaging, and generating thick and bright edges because of the involved differentiation on two rows and columns [7]. An example of Sobel edge detection is shown in Fig. 1. The advantage of edge detector methods is their low computational complexity and memory requirements. In some other algorithms, plate detection is performed by finding the borders of a plate using the Hough transform [8], [9], which is a memory and time consuming process [10]. Such method fails in detecting plates without clear borders. Fig. 2 compares the results of applying the Hough transform on both clean and dirty plates. Wavelet analysis has also been utilized for detecting plates [11], [12]. In Wavelet based methods, high-frequency coefficients are used to detect plate candidates. Since these coefficients correspond to the edges, these algorithms suffer from the same disadvantages of edge detection algorithms. In [13], and [14], color is incorporated as an important feature in detecting plates. These algorithms fail on gray-scale images or images with low color disparity. Some detection algorithms are based on a combination of Mathematical Morphology and Connected Component analysis [15], [16]. There are algorithms which first enhance the plate contrast and then apply the detection algorithm [4], [17]. Most of these algorithms are successful in identifying clean plates, but fail when it comes to detecting dirty and low contrast plates. This is due to the fact that these algorithms need a medium to high contrast images for plate detection. Moreover, for dirty plates color is not a reliable feature for plate detection.

For character segmentation, there are many algorithms based on morphological operations [18], and connected component analysis (CCA) [19], [20]. In such methods, it is necessary to apply a proper thresholding method to obtain a binary image of the plate before any further processing. For example, CCA, which has been used in many research works, depends highly on the previously applied thresholding method. Thresholding methods like Niblack [21], SAUVOLA [22], Wolf and Jolion [23] and OTSU [24] are good candidates for plate binarization. Applying such algorithms on plate candidates relies on appropriate setting of the involved parameters. Unfortunately, at the detection step there is no information about the input plate qual-

ity and the parameters cannot be tuned appropriately. Therefore, the recognition accuracy of such algorithms decreases when different plate qualities are involved.

For character recognition, many different classification tools and techniques have been utilized so far, such as Artificial Neural Networks (ANN), Support Vector Machines (SVM), Bayes classifier, K-nearest neighbor and so on. Classifiers are applied to the features extracted from the image segmentation. For feature extraction, many methods including character skeleton [25], active areas [26], HOG [27] and horizontal and vertical projection [28], and multiclass AdaBoost approach [29] have been proposed. In some methods, the character recognition system is based on keypoints localization like SIFT [30] and SURF [31]. In [32], SIFT is utilized in the plate recognition process. In [19], direction contributive density is used for character recognition. In [33], a comprehensive study of character recognition features is presented. We have proposed a new feature set for the character recognition part of our system. SVM classifier, which has been utilized in various character recognition and classification methods [34]–[36], are employed in our system as well.

Our final system is capable of detecting and recognizing multiple license plates in a single frame in high speed applications, such as highway vehicle detection. Based on the resolution of our cameras and the geometrical setup of our system the maximum number of plates in each captured frame is at most seven. There are several challenges in design and deployment of robust highly accurate ANPR systems. These challenges arise in handling high vehicles speeds, different weather conditions (such as rainy, snowy and dusty), different lighting conditions (such as sunrise and sunset effects on license plates), and camera vibrations. Such phenomenons and problems make many ANPR systems to present poor recognition results. In addition to that, and for the sake of computational complexity, we have tried many different algorithms for different parts of our ANPR system. For example, algorithms with high computational needs were rejected in spite of their high accuracy, such as the algorithms in [37]–[39].

To sum up, we must say that many ITS applications depend on real-world Automatic Number Plate Recognition (ANPR) systems. The performance of ANPR systems are degraded by low clarity of the vehicles plates, variations in weather and lighting conditions, and high vehicle speeds. For instance, most of well-known plate detection and recognition algorithms fail on dirty plates. This paper presents a system that overcomes such issues by presenting proper hardware platforms along with real-time, robust, and innovative algorithms. The system can deal with, various weather and environmental conditions, different plate formats and languages, low plates clarities, high vehicle speeds and various traffic situations in online real-world applications.

This paper is organized as follows: in Section II, the proposed license plate detection algorithm for both day and night cases is elaborated. We have proposed and utilized a revised version of RANSAC in this section. The original RANSAC is introduced in [40]. The revised RANSAC not only identifies the vehicles plates with a high detection rate but also drastically reduces the processing time in comparison with the well-known methods.



Fig. 3. Comparing monochrome camera outputs for (a) variable exposure time (high-quality images in any weather conditions) and (b) fixed exposure time (poor quality in bad environmental conditions).

In Section III, the proposed character segmentation algorithm is discussed. The character recognition part and the proposed features are described in Section IV. Section V is devoted to evaluations, the comparison results, implementation and the practical considerations. Finally, Section VI concludes the paper.

## II. LICENSE PLATE DETECTION

The goal of this section is to elaborate on the methods of finding the vehicles plates location in captured images. Generally, a monochrome camera with a synchronous IR projector and a color camera are employed in a multi-purpose industrial ANPR system. The monochrome camera with IR projector is responsible for plate detection during the night or other low illumination conditions. It is worthwhile to note that for the IR projector to be effective the vehicles plates should have been coated with IR reflective materials. The role of IR projectors is also important in detecting dirty plates even in daylight by taking care of the camera exposure time. IR projector power has a close relation with the camera exposure time and the exposure time plays an important role in the final clarity of the vehicles plates. Since vehicles move swiftly, high values of exposure time lead to blurred images while low exposure time values produce dark images. Therefore, it is important to tune the output power of IR projector with respect to the exposure time of the monochrome camera. It is also necessary to have an adaptive procedure to fine-tune the exposure time based on the lighting conditions. Modifying the exposure time is performed in an adaptive procedure that gets its feedback from the thickness of plate characters. Having thin characters is a sign of high ambient light. In this case, we must decrease the exposure time. On the other hand, achieving thick characters shows that the environmental light is low and we must increase the exposure time. The modification steps are dependent on the setup and application and must be found experimentally. For example, at sunrise, sunlight reflects from vehicles that move from east to west. In such cases, exposure time should be lowered down to a value that eliminates the reflections. In Fig. 3, a comparison between fixed and variable exposure time algorithms is demonstrated. Color cameras are needed to provide visual evidences for the violation scenes in order to support the corresponding traffic tickets.

As discussed in the introduction section, there are many algorithms to detect the exact location of plates in an image. We have tried most of the algorithms proposed so far. All of these algorithms fail on dirty plates and the plates with low contrast between plate characters and the background. Figs. 1 and 2 show the results of applying vertical Sobel edge operator and Hough transform on dirty and clean plates.

### A. Finding License Plate Regions

The block diagram of our proposed ANPR system is shown in Fig. 4. This system has been tested extensively over one year of operation period in different streets and highways of Tehran, the capital city of Iran. Our efforts and eagerness to have the data sets of other researchers, mentioned in our tests and references, have been unsuccessful, as of now. Thus, to perform valid and fair comparisons, we have collected huge and highly inclusive data sets ourselves. As a matter of fact, we can claim that our test-bed (based on our data sets) is the most difficult and comprehensive one presented so far. In this paper, two different data sets are prepared and used for training, tests and evaluation purposes. They are named “Crossroad Data set” and “Highway Data set.” The Crossroad Data set was collected from several crossroads to identify the vehicles that pass over the pedestrian line. It consists of more than 10 000 images captured by traffic cameras. For 8000 images of this data set, one to seven vehicles are present in each image. In 2000 images, no vehicle is present, that is, the scene shows an empty street. These 2000 images were used to measure the false alarm rate for the plate detection process. The Highway Data set was collected from highways and streets for vehicle counting applications. Plate based car counting systems can capture valuable information about traffic congestions, type of vehicles in different lanes and so on. The Highway Data set consists of more than five million frames which include more than two hundred thousand different vehicles. The images of Highway Data set have been captured at different hours of the day and different days of the year, so that the changes in shadows, illuminations, reflections, and weather conditions affect the images considerably. It is worth mentioning that three different background colors exist in Iranian plates: red, yellow and white. The characters color, like those of many other countries, is either black or white. In most industrial projects, due to the resolution of the traffic cameras, vehicles are not visible in regions where the vehicles distance to camera is substantial. Also, there are regions, such as pavement or sidewalk, where no vehicles exist. Therefore, an ROI (Region of Interest) is chosen to exclude the irrelevant regions. An ROI in our system can be a polygon, in its most general form. Polygon ROI selection is very useful for industrial ANPR systems since it helps to eliminate the regions where no plate exists and hence reduces the overall processing time. From now on, for evaluating the proposed algorithms, only the ROI is shown. Both Crossroad and Highway data sets are composed of clean, medium, and dirty plates. For quantitative evaluations and comparisons, we have come up with a clear definition of these plate conditions: In clean plates all the characters are legible, medium quality plates are those which include one or two illegible characters, and dirty plates are those with less than four legible characters. The proportion of clean, medium and dirty types in our data sets are 70%, 25%, and 5%, respectively. These proportions reflect the approximate occurrence rate of these plate types in real world, as per our experience. Fig. 5 illustrates some samples of each plate type.

The detection process is initiated by a gray scale image, as depicted in Fig. 4. The main reason is that color cannot be used as a discriminative feature in detecting dirty plates. Moreover, processing gray scale images not only helps to reduce the

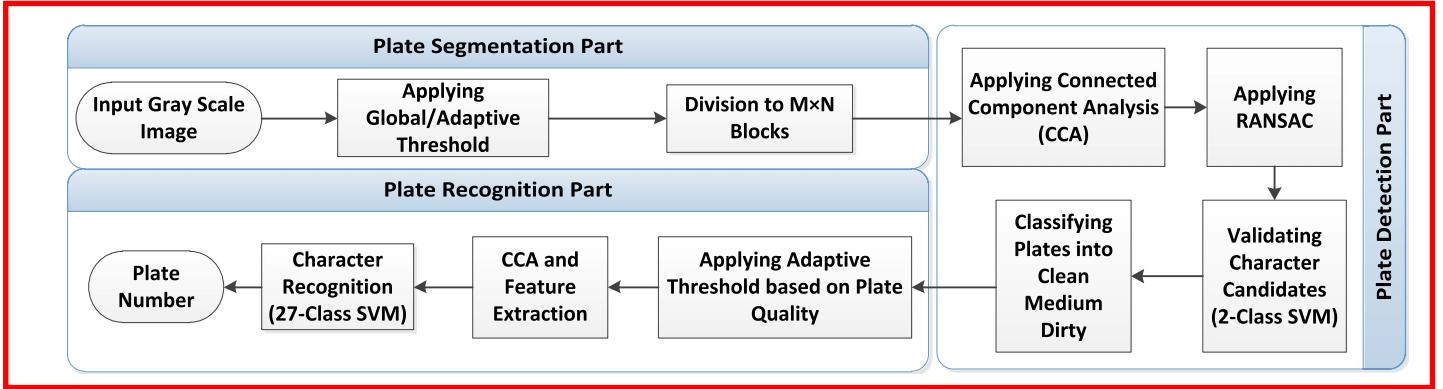


Fig. 4. Block Diagram of our ANPR system.

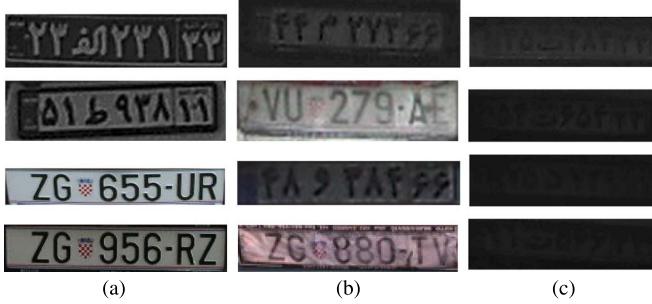


Fig. 5. Samples of plates in our data sets. (a) Clean plates. (b) Medium quality plates. (c) Dirty plates.

processing time, but also makes the algorithm more robust to color changes caused by different lighting condition throughout the day. Hence, Our detection algorithm is directly applicable to both color and monochrome cameras. In the next step, an algorithm to detect moving objects in two consecutive video frames is applied. There are a variety of methods to calculate the dynamic parts of images based on the comparison with previous video frames. Algorithms based on optical flow [41]–[44] and background subtraction [45]–[47] are just to name a few reported detection methods.

Background subtraction methods need a background model to detect the foreground objects. These methods are computationally expensive and memory consuming for images of large sizes. Besides, they do not provide any sense about the speed of the moving objects and suffer from the noise components. On the other hand, optical flow based algorithms such as variational methods [44] are great for calculating speed vectors and removing the noise components. In this paper, the Lucas and Kanade method [48] is utilized for extracting speed vectors. First, key-points are extracted using Shi-Tomasi [49] method and then these key-points are tracked in each frame to calculate the speed vectors. These vectors are connected to each other using simple morphological operations such as dilation and closing. This step helps to find the regions that contain the vehicles and reduces the total computational requirements.

In the next step, an adaptive thresholding is applied to the gray scale image. The adaptive threshold is calculated based on the local mean of pixels intensity in windows of  $m \times n$  pixels:

$$O(x, y) = \begin{cases} 255 & I(x, y) < \frac{\sum_{i=x-m/2}^{x+m/2} \sum_{j=y-n/2}^{y+n/2} I(i, j)}{m \times n} \\ 0 & I(x, y) \geq \frac{\sum_{i=x-m/2}^{x+m/2} \sum_{j=y-n/2}^{y+n/2} I(i, j)}{m \times n} \end{cases} \quad (1)$$

where  $I$  and  $O$  are the input and output images respectively. The window size parameters,  $m$  and  $n$ , are chosen based on the characters size in the region. As we found empirically, the thresholding window based on the local mean outperforms other methods like local Gaussian and local median windows. After thresholding, the intersection of the two images from the two last steps (morphology and thresholding) is determined. The intersection helps to eliminate irrelevant regions from further processing. Sample output images at all of the steps are shown in Fig. 6(e).

In Fig. 6(d), all bright regions are most likely to contain plate candidates. On the contrary, it is less probable to find a dark region which includes any plate candidates. Since the vehicles that are located far from the cameras have smaller plates compared to the closer vehicles, a linearly increasing function is utilized to set the local window sizes [see Fig. 6(a)]. Using (2) and (3) the size issue is compensated through:

$$H_w = C_h \times \left[ \left\lfloor \frac{R_p \times (H_{bc} - H_{sc})}{H} \right\rfloor + H_{sc} \right] \quad (2)$$

$$W_w = C_w \times \left[ \left\lfloor \frac{R_p \times (W_{bc} - W_{sc})}{H} \right\rfloor + W_{sc} \right] \quad (3)$$

where  $R_p$  is the current pixel row,  $H$  is the height of the image;  $H_{bc}$  and  $H_{sc}$  are maximum and minimum characters height in the image, respectively. Similarly,  $W_{bc}$  and  $W_{sc}$  are maximum and minimum characters width in the image.  $C_h$  and  $C_w$  are two constants that their values depend on the plate format. For example, for vehicles plates that consists of a single row of characters, the best range for  $C_h$  is found to be between one and two.  $H_w$  and  $W_w$  are the height and the width of the local window.

In our system the character size parameters must be set manually based on the setup and the application. Such possibility can be seen as a common feature in many industrial ANPR systems. As a matter of fact, similar to other industrial ANPR systems, the character size parameters for all relevant plate formats must be specified to our system. Any plates with unspecified formats (out of the pre-known set of formats) cannot be recognized. Fig. 6(e) is the intersection of Fig. 6(d) and (a). The bright components on the intersection are candidates for plate localization.

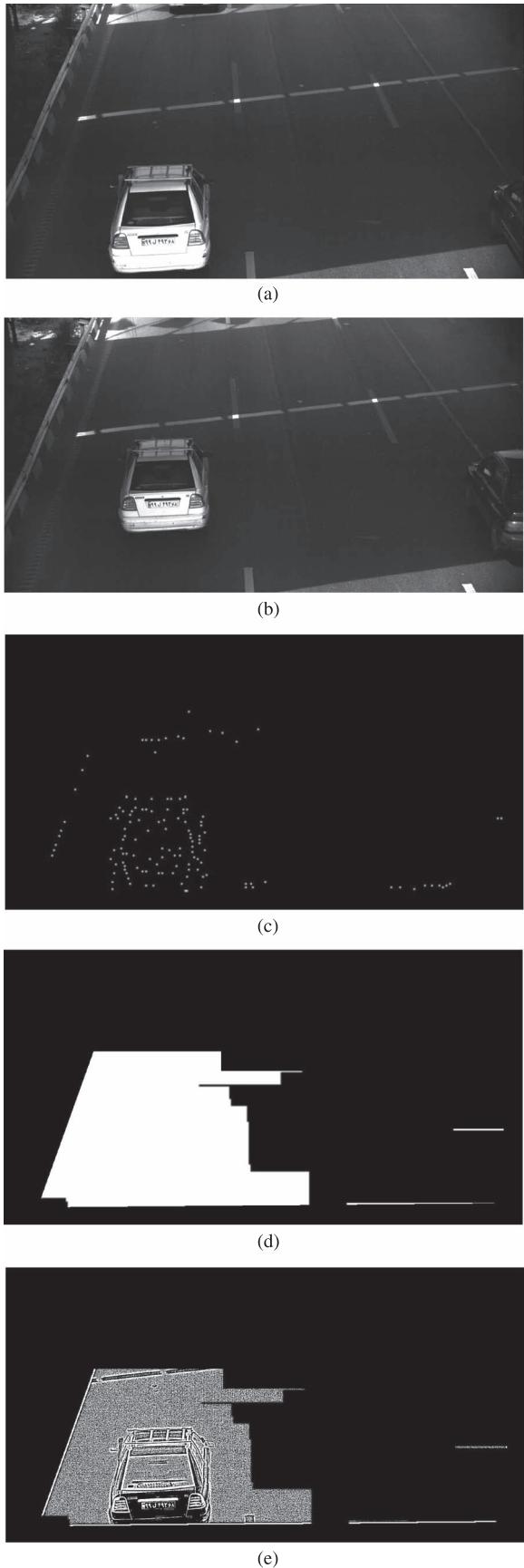


Fig. 6. Procedure for finding most likely plate regions. (a) and (b) Two consecutive input frames. (c) Shi-Thomasi feature points. (d) Morphological operation output. (e) Output of adaptive local mean thresholding.

### B. License Plate Localization in Daytime

After extracting the regions that are most likely to include plates, as explained in previous section, a more accurate vehicles plates localization is performed based on the concept of Random Sampling Consensus (RANSAC). RANSAC is an iterative algorithm used to fit a robust mathematical model to a set of observed data [40]. This method ignores the outliers and finds the best model to the rest of the given data. The main application of RANSAC in machine vision field is in stereo vision, and specifically in finding the Fundamental Matrix [50].

In this paper, a revised version of RANSAC algorithm is designed and exploited in license plate localization process. Our procedure to locate the plate candidates is performed in four steps:

- Step 1: A sliding window is moved all over the binarized image. Fig. 6(e) is a sample input image to this step. The size of the sliding window in any region is chosen to be two times larger than the vehicle plate size in that region. We use (2) and (3) to calculate the window size at this step.
- Step 2: Connected Component Algorithm is applied to the window of step 1.
- Step 3: The components with heights and widths of a plate character are chosen and the rest of them are discarded.
- Step 4: RANSAC is utilized to determine the best line that represents the given data. This line should have two main properties: first, it should cross more than  $k$  character candidates with specific space between them, and next, it should have a reasonable slope. The limiting slope should be set based on the camera view. In this paper, the limiting numbers for crossing characters in Iranian plates and slope are found experimentally as 5 components and 20 degrees respectively.

Fig. 7 illustrates the effect of applying RANSAC on windows with and without plates. The character candidates are shown in gray. Note that for any window, the exact number of inliers and outliers are unknown. Therefore, all of the components should be tried in order to find the best candidate model. If the numbers of components are equal to  $n$ , the number of operations on  $n$  components to get the best model would be:  $N = n(n - 1)/2$ . Where  $N$  is the total number of operations for applying RANSAC to the whole labeled components.  $N$  or the computational complexity increases as the number of components increase. In this study, we have modified this process to reduce the computational complexity of this algorithm to  $(n - 3)$ . The modified process includes the following steps:

- Step 1: The candidate components positions are determined based on their horizontal and vertical distance to the origin of the sliding window. Then, they are sorted based on their distance to the vertical axis origin in the sliding window.
- Step 2: The  $n_i$ th and the  $n_{i+3}$ th candidates are chosen and connected by a line.

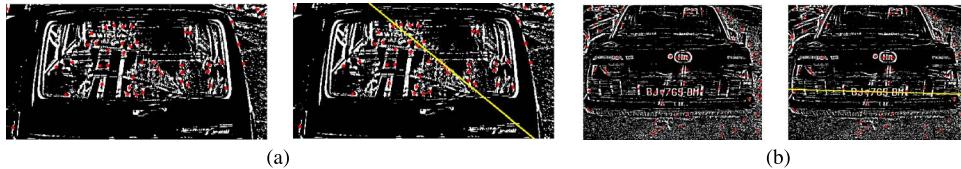


Fig. 7. Results of applying our revised RANSAC algorithm on (a) nonplate candidates and (b) plate candidates.

Step 3: The total distance of other points to this line is calculated based on:

$$D = \sum_{i=0}^n \left| \left[ \left( P_{1x} - \frac{P_{2y} - P_{1y}}{m} \right) - P_{ix} \right] \times \sin \theta \right| \quad (4)$$

$D$  is the total distance of other points to the line determined at step 2.

Step 4: Step 2 and 3 are repeated for the rest of the candidates.

Step 5: The line with lower total distance ( $D$ ) is chosen as the representative model of the RANSAC Algorithm.

In (4),  $P_1$  is one of the two points which represents the model line.  $P_i$  is the  $i$ th point in observed data and  $m$  is the slope of the line between the points  $P_1$  and  $P_2$ . Thus:

$$m = \frac{P_{2y} - P_{1y}}{P_{2x} - P_{1x}} \quad (5)$$

and  $\theta$  is the angle of the connecting line between the  $n_i$ th and the  $n_{i+1}$ th candidates with respect to horizontal axis:

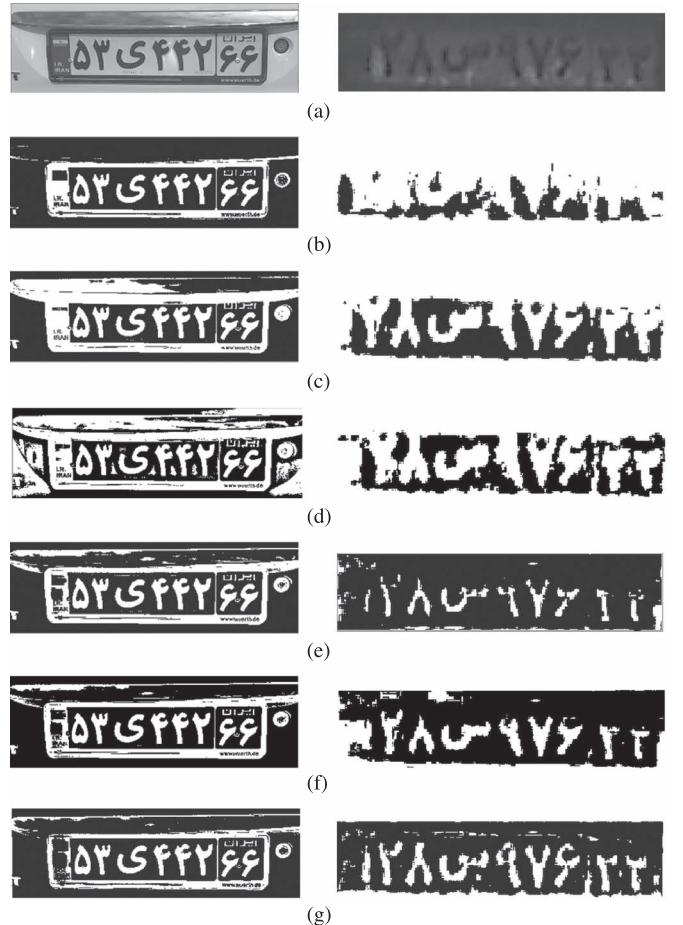
$$\theta = \arctan(m). \quad (6)$$

We have dropped  $n_{i+1}$ th and  $n_{i+2}$ th points to improve the processing time and remove the effect of noisy points. Our experiments showed a small improvement in detection accuracy after dropping intermediate points. In calculating the total distance, a Gaussian weighting function is used to assign higher weights to the points closer to the line and lower weights to the points farther from the line. We have found empirically that Gaussian weighting helps to reduce the false positive (FP) probability in detecting plate candidates considerably.

At this stage, by finding the best line through each plate, the skewness and the rotation of the plates can be determined and compensated. The model line slope,  $m$ , is a good measure of how severely the plate is rotated. By calculating a rotation matrix based on the value of  $m$ , the plate rotation can be reversed to improve the next steps.

### C. License Plate Localization in Nighttime

At night and other low light conditions, vehicles plates are not visible by traffic cameras. In countries like Iran where the vehicles plates are coated by infrared sensitive materials, IR projectors are utilized to make the plates visible without distracting the drivers. The plates, when exposed to IR rays, look shiny, clear and illuminated in images captured by monochrome IR cameras. Plate detection is generally much easier and faster using IR cameras. In an IR image, plates are the only high contrast regions compared to the rest of the image. In low light conditions, a large proportion of the image area is covered with low gray level regions. As such, the number of false positive errors in plate detection is reduced considerably. After detecting

Fig. 8. Applying different thresholding methods on clean and dirty plates. (a) Input plates. (b) Global thresholding  $G=60$ . (c) OTSU method. (d) Niblack method. (e) Sauvola method. (f) Wolf and Jolion method. (g) Our method.

the high contrast areas, the plate detection process is continued by the RANSAC algorithm. The remaining procedures are the same as the counterparts in the daytime version of the algorithm, which are explained in the previous subsection. To sum up, we can say that the main difference between nighttime and daytime algorithms is in the plate detection part. The next steps are completely the same.

### III. CHARACTER SEGMENTATION

Connected Component Analysis, CCA, is one of the most widely selected algorithm for the initial step of character recognition in various segmentation methods [1], [19], [20]. After detecting the exact location of the plates in a captured image, a binarization process is performed on the detected plates. Several thresholding methods have been proposed for the binarization process at this step. In Fig. 8, the results of applying some state-of-the art thresholding algorithms on

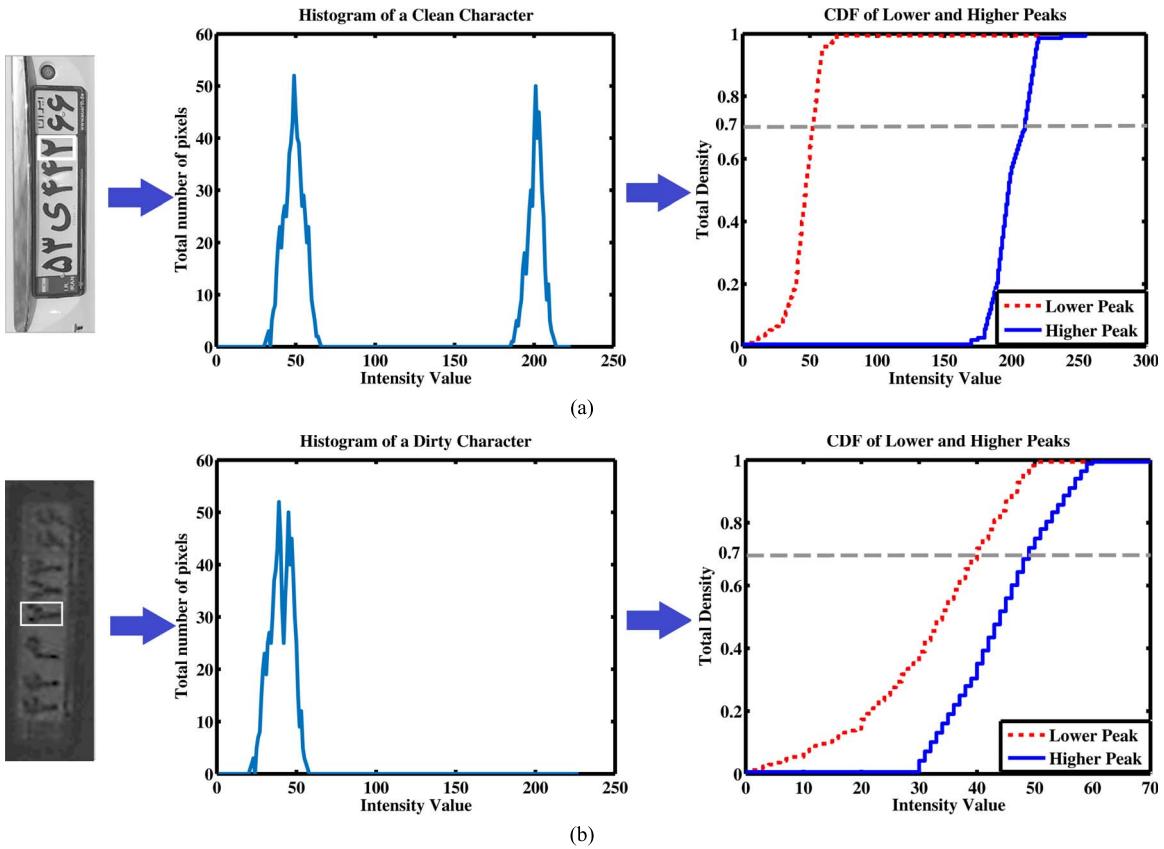


Fig. 9. Histogram and CDF of (a) a clean character and (b) a dirty character.

clean and dirty plates are shown. As can be seen in Fig. 8, most of the well-known thresholding methods produce desired segmentation results, but they all fail on dirty plates.

Most of the well-known thresholding methods mentioned in the introduction section have been evaluated on our data sets. Some of these algorithms, such as Niblack and OTSU fill the area around characters and make it impossible to recognize the characters correctly, e.g., see Fig. 8(c) and (d). On the contrary, some other algorithms, like Wolf and Jolion methods, produce thin characters which results in losing valuable details, e.g., see Fig. 8(f). Processing thin and reduced-size characters lead to character misclassification in the next parts. To address such problems and other thresholding-stage issues, some researchers propose to apply an equilibrium algorithm to the image in order to enhance its contrast [4], [17]. Such contrast enhancing methods rely on the parameters tuning. The values of the parameters are obtained empirically through a large number of experiments. Unfortunately, the equilibrium algorithms also fail in correct segmentation of the characters in our data sets. The reason is that there are no general optimum values for parameters such as window size, standard deviation, etc., that work equally well for all plate types. In other words, the optimum parameters values for dirty and clean plates are not the same.

In this paper, a new thresholding method based on the plate quality is proposed. Compared to the well-known algorithms discussed so far, the proposed algorithm is not only more accurate but also faster. Moreover, it does not rely on the enhancement of input images.

Recall that at the end of the previous part, we have the locations of plate candidates. If a location is marked to be a plate candidate, the characters on the RANSAC output line must be fed into character recognition part. The details of character recognition process are discussed in Section IV. It is worth mentioning that a two-class SVM is trained at this step to determine whether a component is a character or not. If the answer is positive, the cumulative density functions (CDFs) of inner and outer areas are calculated for each character independently. Then, using CDF we can find a value that more than 70% of the pixels have intensity of less than that value. We mark this point on each CDF and refer to it as a representative point. Since there are two CDFs, the standard deviation of the two representative points can be calculated. Fig. 9 shows the process for finding the two representative points. Based on this standard deviation, an adaptive thresholding algorithm is applied on the plate, according to (7).

$$O(x, y) = \begin{cases} 255 & I(x, y) < \alpha + \beta \\ 0 & I(x, y) \geq \alpha + \beta \end{cases} \quad (7)$$

where  $\alpha$  and  $\beta$  are calculated as follows:

$$\alpha = \frac{\sum_{i=x-m/2}^{x+m/2} \sum_{j=y-n/2}^{y+n/2} k(i, j) \times I(i, j)}{m \times n} \quad (8)$$

$$\beta = \gamma \times \alpha. \quad (9)$$

$\gamma$  value is calculated by:

$$\gamma = \begin{cases} \lceil \sqrt{\frac{s}{2}} \rceil & s < 10 \\ \lceil \frac{s}{2} \rceil & s < 20 \\ 0.7 & \text{otherwise.} \end{cases} \quad (10)$$

In (8),  $k$  is a kernel function. Well-known kernel functions such as Uniform, Gaussian, Epanechnikov, Cosine and etc. have been tested to find out which one provides the best results. Among these kernels, Uniform and Gaussian kernels perform better compared to other kernels. In (10),  $s$  is the standard deviation of the two representative points. The relation between  $\gamma$  and  $s$  is found through a huge set of experiments. By conducting additional experiments on dirty plates, and calculating the standard deviations, it is found that the standard deviation values are always less than ten. Therefore, a small deviation from  $\alpha$  value leads to desired results. According to Fig. 8, for clean plates all the thresholding methods achieve the same character segmentation results, however, for the dirty plates, the segmentation process using the proposed method outperforms the conventional algorithms. The proposed thresholding Algorithm helps to increase the detection accuracy for dirty plates. The advantage of the proposed thresholding method lies in its adaptive nature for setting different parameters of clean and dirty plates.

#### IV. LICENSE PLATE CHARACTER RECOGNITION

In this section, we elaborate on the character recognition process. The features exploited in this process are extracted from the characters in their original size. In other words resizing is avoided, since it results in losing information. In Fig. 11, the complete set of characters that appear on Iranian plates is demonstrated. For each character candidate, four scale invariant features are calculated. The details of each feature can be described as follows:

- a) Distance to Walls (DTW): Calculation of this feature is based on Active Pixels. Active pixels are pixels with the highest possible intensity values. The highest intensity value depends on the image format. As an example, for 8-bit gray scale format the maximum level is 255. To calculate DTW, the character area is divided into 12 regions ( $4 \times 3$  rows  $\times$  columns) and in each region the vertical and horizontal distance of active pixels are calculated with respect to the borders of the character area. For each region the shortest and the longest distances are stored. Fig. 10(a) shows the calculation scheme for DTW feature. DTW is a 48-dimensional vector.
- b) Cross-Time Feature (CTF): This feature includes the six widest white to black and black to white transitions in traversing a character horizontally and vertically. For each row and column of a character candidate, the six widest transitions are calculated. These crossings are sorted and the first three rows and columns with highest and lowest transitions are stored, along with their locations. Storing the locations improves the classification results for random transitions in non-character candidates. In Fig. 10(b), the details of calculating this feature is depicted. The gray lines show the black to white crossing widths and white

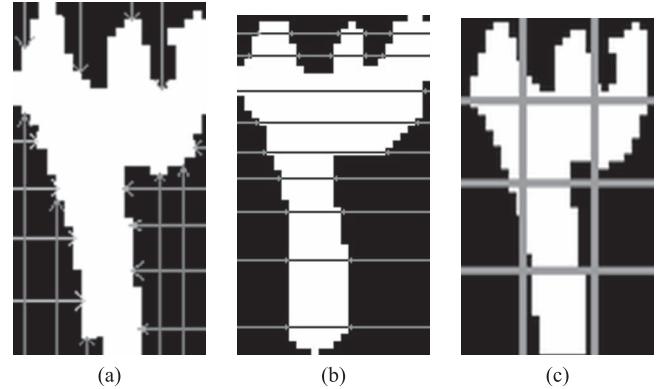


Fig. 10. Features used for character recognition. (a) DTW. (b) CTF. (c) ARR.

TYPE 1	۶۵۴۳۲۱۰۹۸۷۶۵۴۳۲۱۰
TYPE 2	۰۹۸۷۶۵۴۳۲۱۰۹۸۷۶۵۴۳۲۱۰
TYPE 3	۰۹۸۷۶۵۴۳۲۱۰۹۸۷۶۵۴۳۲۱۰

Fig. 11. Different types of characters and their members in Persian plates.

lines depict the white to black crossing widths. For each component, CTF is a 24-dimensional vector.

- c) Active Region Ratio (ARR): For calculating this feature the candidate rows and columns are divided into four and three parts respectively. The number of active pixels in each block divided by the total number of active pixels is chosen as a feature for that block. Fig. 10(c) shows an example of calculating this feature.
- d) Height to Width Ratio (HWR): Alphanumeric characters in any language can be classified under three types, when it comes to height to width ratio: Characters with greater heights, equal heights and widths, and greater widths. A list of these three types for Iranian plate characters set is shown in Fig. 11. HWR, which is a one dimensional scalar, discriminates between these three types.

These four features are able to recognize vehicles plates with character size of at least  $7 \times 3$  pixels. This is the basic requirement for detecting the smallest character in Persian which is the character for number one. Due to the large size of the feature vector, a feature reduction process is performed on the concatenated feature vector. F-Score [51] is chosen among various feature reduction methods like Recursive Feature Elimination (RFE) [52], Principal Component Analysis (PCA) [53], random projection [54] because of its lower processing time and satisfactory results. F-Score assigns a value to each feature which determines the discriminative capability of the feature in the classification process:

$$F(i) = \frac{\left( \bar{x}_i^{(p)} - \bar{x}_i \right)^2 + \left( \bar{x}_i^{(n)} - \bar{x}_i \right)^2}{\frac{1}{N_p-1} \sum_{k=1}^{N_p} \left( x_{k,i}^{(p)} - \bar{x}_i^{(p)} \right)^2 + \frac{1}{N_n-1} \sum_{k=1}^{N_n} \left( x_{k,i}^{(n)} - \bar{x}_i^{(n)} \right)^2} \quad (11)$$

where  $\bar{x}_i$ ,  $\bar{x}_i^{(p)}$ ,  $\bar{x}_i^{(n)}$  are the average value of the  $i$ th feature over the whole, positive decision, and negative decision parts of the data sets respectively. In our case, positive set includes the

characters class and negative set consists of the non-characters class.  $x_{k,i}^{(p)}$  is the  $i$ th feature of the  $k$ th positive instance and  $x_{k,i}^{(n)}$  is the  $i$ th feature of the  $k$ th negative sample in the data set. The numerator shows the discrimination between the positive and negative sets. The denominator shows the discrimination within each of the two sets. In this paper, features were sorted based on their F-score value. As per our experiments, the first 90% of the strongest features presented the best results in the classification process. DTW and CTF were the most important features in the F-Score output for character recognition part. Final features were employed for both Crossroad and Highway Data sets.

In our system, two SVM classifiers are used, one for character detection (in plate detection part) and the other one for character recognition (in the character recognition part). As stated in Section II, for better segmentation results, a character detection process is applied to each character candidate. In other words, our character detector checks whether a candidate can be regarded as a character or not. Character detection is performed by an SVM, which classifies the candidates into two classes; class of characters and class of non-characters. Obviously, the complexity of this classifier is much lower than the recognition part classifier, in which 27 classes (one for each character) exist. It is worth mentioning that for the two-class SVM in character detection part, there is no need to extract all the features. Our experiments showed that CTF and HWR features are sufficient to decide between characters and non-characters. The size of training data for characters class was ten thousand and for non-characters was more than forty thousand samples, in our experiments. A linear kernel is used for character detection SVM. Linear kernels are widely used in two-class (binary) problems and lead to less computational time in both training and test phases.

The other SVM classifier is trained for the character recognition part. Using the set of four features, this SVM is trained to distinguish between 27 classes. The number of samples for each class was at least five thousand characters. The total size of the training set was more than two hundred thousand samples. The training data were cross validated to find the best SVM kernel parameters, using  $k$ -fold cross validation procedure. As a common choice, Radial Basis Function (RBF) kernel is employed as SVM kernel, in character recognition part. The kernel is:

$$K(x, x') = e^{-\gamma \|x-x'\|} \quad (12)$$

where  $\gamma = 1/(2\sigma^2)$ , and  $\sigma$  is a parameter chosen based on  $k$  fold cross validation. To each candidate, a vector of probabilities is assigned by the classifier. The size of this vector is equal to the number of classes, namely 27. Elements of this vector shows the probability of being a member of each class.

It is worth mentioning that in every language there are confusing character pairs that make the character recognition task more challenging. As an example in English, the letter "I" and digit "1" are very close to each other. Letter "O" and digit "0" is another example. For better recognition results, it is a common practice to put each of such pairs in a single group at the beginning and then try to discriminate between them, based on their differences. We have used such methods for some letters and numbers in Persian characters.

## V. EXPERIMENTAL RESULTS

The whole system is implemented on two hardware platforms: a) Industrial biscuit PC with Intel core i5 2.2 GHz CPU, and 4 Gbytes DDR3 random access memory, b) TI DSP board with OMAP3530 dual core (C64x+ DSP core and Cortex A8 ARM core), 256 MBytes DDR2 RAM. From software point of view, on both hardware platforms, the algorithms were developed in C++, under Embedded Linux operating system empowered with OPENCV 2.4.3 library.

The Crossroad and the Highway data sets of this paper consist of only Iranian plates. Each plate includes seven digits and one Persian letter. Total character set includes 27 elements, 9 digits and 18 alphabetical letters. The plate sizes in the data sets vary from  $16 \times 80$  to  $25 \times 150$ . The proposed system was compared with three previous methods [26], [29], [56] on their own data sets. The results for plate detection and recognition are included in Table II. It is worth mentioning that in these references, the data sets are not as challenging as the data sets we used in this paper. For instance, in none of their images more than five vehicles exist in the scene. Therefore, no comparison on computational complexity and processing time could be performed. Most of the data sets in the ANPR literature consist of images captured by handheld cameras and the distance to vehicles is less than three meters.

An online, publicly available English vehicle plate data set is also utilized in our evaluations [55]. This data set and our evaluations are used to show that our system is not dependent on Persian plates and can also detect and recognize English plates very accurately. Fig. 12 shows the results of applying our ANPR system on different data sets. The white rectangles on gray scale images and yellow rectangles in color images are the outputs of the plate detection part. As shown in Fig. 12, the images have captured with different backgrounds, illuminations, and camera viewing angles. In Table I, the processing time of our algorithms are summarized. The results are based on how many plates exist in an image. The first row in Table I shows that our system is capable of processing 50 frames per second when no vehicle plates exist in the image. As mentioned in Section II, 2000 images in Crossroad Data set include no vehicle plates. Less than 10 false alarms were detected on these images. Thus, the false alarm rate of our system is less than 0.5%. This is equivalent to true rejection rate of more than 99.5%.

Table II reports the evaluation of the proposed system over various data sets, including our data sets and the ones used in other references. According to this table, in all cases our system outperforms other competitors on their own data sets. In our evaluations, the overall accuracy is calculated based on:

$$\text{Overall Accuracy} = (D \times S \times R)\% \quad (13)$$

where  $D$  is the plate detection rate,  $S$  is the character segmentation rate and  $R$  is the plate recognition rate. The plate detection rate,  $D$ , is defined as the number of detected plates divided by total present plates in a data set. The character segmentation rate,  $S$ , equals to the ratio of number of correctly segmented characters to total number of characters in a data set. The character recognition rate,  $R$ , is the proportion of correctly



Fig. 12. (a) Our Crossroad data set samples. (b) Data set of [26] and [29] samples. (c) Data set of [55] samples. (d) ANPR results in snowy weather (from our Highway data set). (e) ANPR results in congestion time (from our Highway data set). (f) Our ANPR system web interface.

TABLE I  
AVERAGE PROCESSING TIME (IN MILISECONDS) FOR THE PROPOSED ANPR SYSTEM

No. of plates	Platform	Plate Detection		Plate Recognition		Total Processing Time	
		PC	OMAP 3530	PC	OMAP 3530	PC	OMAP 3530
Zero		20	65	0	0	20	65
One		36	90	10	27	46	117
Two		40	105	20	55	60	160
Three		44	118	30	80	74	198
Four		47	131	40	110	87	241
Five		52	145	50	140	102	285
Six		55	170	60	185	110	355
Seven		60	195	70	210	130	405

TABLE II  
EVALUATION OF THE PROPOSED ANPR SYSTEM AND COMPARISON WITH OTHER REPORTED SYSTEMS OVER VARIOUS DATA SETS

Dataset		Plate Detection Accuracy		Plate Recognition Accuracy		Overall Accuracy	
		Reported Accuracy	Our Method Accuracy	Reported Accuracy	Our Method Accuracy	Reported Accuracy	Our Method Accuracy
Crossroad Dataset			98.7 %		97.6%		96.3%
Dirty Plates of Crossroad Dataset			97.9 %		93.4%		91.4%
Ref. [55]	English Plate Dataset		100 %		97%		97%
Ref. [26]	Far distance set	98%	100%	98%	100%	96%	100%
	Close angle set	100%	100%	95%	100%	95%	100%
Ref. [29]		96%	100%	94%	100%	90%	100%
Ref. [56]		99%	100%	97%	100%	96%	100%

classified characters over a data set. It is worth noting that some references did not report their segmentation accuracy. Therefore, for the sake of fair comparisons, the overall accuracy is calculated without considering the segmentation accuracy.

For example in Table II, the segmentation accuracies are not included. Ref. [26] is one of a few literatures that tested their own proposed algorithm on images containing two vehicles plates. The result for each set of this reference is depicted.

TABLE III  
STATE-OF-THE-ART ANPR SYSTEMS COMPARED WITH THE PROPOSED ANPR SYSTEM

System	Reported Accuracy				Plate Characters	Number of plates	Image Size	Platform	Processing Time(ms)	Methodology
	Plate Detection	Character Segmentation	Plate Recognition	Overall Accuracy						
Ref. [13]	NR <sup>1</sup>	99%	NR	NR	Korean	120	640 × 480	CPU P IV 2.4 GHz RAM 1 Gbyte	NR	D: sliding concentric windows R: ANN
Ref. [19]	97.16%	98.34%	97.88%	93.54%	English Japanese	9026	NR	Intel Core 1.8 GHz RAM 1.5 Gbyte	288	D: improved Bernsen algorithm R: SVM
Ref. [20]	96.5%	NR	89.1%	86%	English	1334	NR	CPU P IV 3 GHz RAM 500 MByte	276	D: Sliding Concentric Windows and CCA R: Probabilistic NN
Ref. [26]	97.3%	NR	94.5%	91.94 %	Persian	320	640 × 480	NR	NR	D: Edge Features(Sobel) R: Multi-layer perceptron ANN
Ref. [29]	96.93%	98.75%	94.5%	90.45%	Persian	1185	1024 × 768	NR	NR	D: Morphological operations and AdaBoost R: SAMME
Ref. [57]	95.9%	NR	92.3%	90%	Chinese English	5026	720 × 576	CPU 3 GHz	125	D: edge features(Sobel) R: feed forward NN
Ref. [56]	99.33%	NR	96.6%	96%	Persian	150	640 × 480	Core i3 2330M RAM 2Gbyte	NR	D: Color Features R: ANN
Ref. [58]	97.3%	NR	95.7%	93.1%	English	1176	640 × 480	CPU PIV 2.26 GHz	223	D: salient features R: Self-Defined classifier
Ref. [59]	97.1%	NR	96.4%	93.6%	English	332	867 × 623	CPU P IV 3.2 GHz RAM 512 MByte	594	D: Color feature and hough transform R: Feedback self-learning
Our System	98.7%	100%	97.6%	96.33%	Persian	10000	variable	CPU Corei5 2.2 RAM 4Gbyte	180	D: CCA and RANSAC R: Probabilistic SVM

NR: Not Reported

“Dirty Plates of Crossroad Data set,” the second row in Table II, reports the ANPR tests on dirty plates portion of our Crossroad Data set. As mentioned in Section II, less than 5% of the passing vehicles have dirty plates. In Crossroad Data set around 400 images contain dirty vehicle plates. On these dirty plate portion, our system achieved detection accuracy of 97.9% and recognition accuracy of 94.1%. Note that, the accuracy of segmentation and recognition parts are calculated based on the legible characters and not all of the characters. It is worth mentioning that, the proposed thresholding algorithm plays a crucial role in achieving the reported accuracies. Note that, for dirty plates, the detection rate is the most important part. In Vehicle counting projects, for example on entry/exit routes of a city, it helps to collect more accurate statistical information about the traffic density.

It is worth noting that the only language dependent part of our system is in character recognition section. To evaluate our character recognition algorithm beside the previous English data set, we have conducted some experiments on a well-known handwritten English numbers data set, called mnist. Our proposed features have been tested on mnist data set. This data set is the most famous data set for handwritten English numbers and includes 60 000 training samples and 10 000 test samples. The accuracy of our algorithm on this data set is 98.5%. The best reported accuracy is 99.77% achieved by the convolutional neural networks [60]. The difference (about 1%) is the price we have paid for lower computational requirements.

Data set of Ref. [55] row in Table II presents the accuracy of our ANPR system on English vehicles plates. The data set used in [55] is publicly available over the Internet. Similar to Iranian data sets in [26], [29], and [56], this data set was captured using handheld cameras and contains about 500 images. We have

used 400 of these images for training and 100 images for testing our system. Similar methods, procedures and features to train and test Persian characters were repeated for English characters as well. To recall briefly, we have employed two SVMs, one for detecting characters (deciding between character and no-character candidates, by a 2-SVM) and the other one for recognizing the 36 characters used in English plates (by a 36-SVM). Similar to Persian characters case, the utilized features were CTF+HWR for detection and DTW+ARR+CTF+HWR for recognition. The achieved accuracies of 100% in plate detection and 97% in plate recognition parts show the flexibility of our system in handling vehicles plates of different formats and languages.

In Table III, various evaluations of different ANPR systems are summarized. The table presents the processing time, captured image size, platform and methodology specifications along with various evaluations of each system to provide thorough and detailed comparisons. Accuracies, in this table, are evaluated at three parts: Detection, Segmentation, and Recognition. According to the evaluation results in Tables I and III, our system achieves the least processing time for largest image sizes, even when multiple plates exist in the view.

Our Highway Data set includes more than  $5 \times 10^6$  video frames in which more than  $2 \times 10^5$  vehicles have passed. This data set was collected under different traffic levels (Light/Medium/Heavy) and lighting conditions. At nighttime, an IR projector was employed to improve the plate detection and recognition. It is worth mentioning that this data set was utilized just for testing our system in highway vehicle counting applications. All trainings related to detection and recognition parts have been done on the Crossroad Data set. Unlike the Crossroad Data set, all the images in Highway Data set were

captured by monochrome cameras. We designed and collected the Highway Data set to include all plate types and environmental, weather, and lighting conditions for the most exhaustive tests. On this data set, our system achieved the accuracy of 99.1% in vehicle counting. This accuracy is based on vehicle plates with minimum of 3 legible characters. This result shows the reliability of our system in such applications.

We have worked with many different industrial cameras (both color and monochrome, e.g., Basler, Brickcom, Hikvision,...) and IR projectors (e.g., Komoto, RayTec,...). The synchronization method depends on the camera model and the IR projector specifications. Each camera manufacturer has its own way to generate a synchronization pulse in order to connect to IR projectors. Basler camera was used in our final system along with a Komoto IR projector. We had to design and make a signal conditioning circuit board using an Opto-coupler and some transistors to generate a suitable pulse for the IR Projector. This glue circuit was needed to convert the camera pulse to the format needed by the IR projector.

## VI. CONCLUSION

In this paper, an industrial, robust and reliable ANPR system for high speed applications is proposed. The main advantage of our system is its high detection and recognition accuracies on dirty plates. To achieve reliable evaluations, two new data sets were created and used in this paper: one for violation detection called “Crossroad Data set” and the other for vehicle counting in highways called “Highway Data set.” The accuracies of our system on the Crossroad Data set are 98.7%, 99.2%, and 97.6% for plate detection, character segmentation, and plate recognition parts, respectively. In vehicle counting application, the detection rate and false alarm rate over the Highway data set are 99.1% and 0.5%, respectively. We have tested this system on a publicly available English plate data set as well and achieved an overall accuracy of 97%. The proposed system is compared to many reported ANPR systems from different point of views. By considering the practical aspects, several copies of our ANPR system have been installed in different intersections and highways of Tehran, capital city of Iran. These systems have been tested day and night over a year and presented robust and reliable performances, in different weather conditions, such as rainy, snowy, and dusty. The character recognition part of our system has been tested separately over the mnist data set and achieved 98.5% accuracy, with comparably low computational requirement. The presented techniques, algorithms and parameter setting procedures, along with our data sets and related evaluations, provide a complete set of solutions to issues and challenges involved in incorporating ANPR systems in various ITS applications.

## ACKNOWLEDGMENT

The authors would like to thank P. Tavafi, CEO of Mahya ITS Company, for his great help and support and also the Research Office of Sharif University of Technology for partially supporting the original research work.

## REFERENCES

- [1] H. Caner, H. S. Gecim, and A. Z. Alkar, “Efficient embedded neural-network-based license plate recognition system,” *IEEE Trans. Veh. Technol.*, vol. 57, no. 5, pp. 2675–2683, Sep. 2008.
- [2] S. D. Palmer and O. N. Aharoni, “System for collision prediction and traffic violation detection,” U.S. Patent 20 130 093 895, Apr. 18, 2013.
- [3] G. Liu, Z. Ma, Z. Du, and C. Wen, “The calculation method of road travel time based on license plate recognition technology,” in *Advances in Information Technology and Education*. New York, NY, USA: Springer, 2011, pp. 385–389.
- [4] V. Abolghasemi and A. Ahmadyfard, “An edge-based color-aided method for license plate detection,” *Image Vis. Comput.*, vol. 27, no. 8, pp. 1134–1142, Jul. 2009.
- [5] B. Hongliang and L. Changping, “A hybrid license plate extraction method based on edge statistics and morphology,” in *Proc. IEEE 17th ICPR*, 2004, vol. 2, pp. 831–834.
- [6] A. Mousa, “Canny edge-detection based vehicle plate recognition,” *Int. J. Signal Process., Image Process. Pattern Recognit.*, vol. 5, no. 3, pp. 1–8, 2012.
- [7] W. Gao, X. Zhang, L. Yang, and H. Liu, “An improved Sobel edge detection,” in *Proc. IEEE 3rd ICCSIT*, 2010, vol. 5, pp. 67–71.
- [8] T. D. Duan, D. A. Duc, and T. L. H. Du, “Combining Hough transform and contour algorithm for detecting vehicles’ license-plates,” in *Proc. IEEE Int. Symp. Intell. Multimedia, Video Speech Process.*, 2004, pp. 747–750.
- [9] K. Deb, A. Avilin, and K.-H. Jo, “An efficient method for correcting vehicle license plate tilt,” in *Proc. IEEE Int. Conf. GrC*, 2010, pp. 127–132.
- [10] S. Du, M. Ibrahim, M. Shehata, and W. Badawy, “Automatic License Plate Recognition (ALPR): A state-of-the-art review,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 23, no. 2, pp. 311–325, Feb. 2013.
- [11] P. Kanani, A. Gupta, D. Yadav, R. Bodade, and R. B. Pachori, “Vehicle license plate localization using wavelets,” in *Proc. IEEE ICT*, 2013, pp. 1160–1164.
- [12] R. T. Lee and K.-C. Hung, “Real-time vehicle license plate recognition based on 1-d discrete periodic wavelet transform,” in *Proc. IEEE IS3C*, 2012, pp. 914–917.
- [13] K. Deb, V. V. Gubarev, and K.-H. Jo, “Vehicle license plate detection algorithm based on color space and geometrical properties,” in *Emerging Intelligent Computing Technology and Applications*. Berlin, Germany: Springer, 2009, pp. 555–564.
- [14] K. Deb, H. Lim, S.-J. Kang, and K.-H. Jo, “An efficient method of vehicle license plate detection based on HSI color model and histogram,” in *Next-Generation Applied Intelligence*. Berlin, Germany: Springer, 2009, pp. 66–75.
- [15] J.-W. Hsieh, S.-H. Yu, and Y.-S. Chen, “Morphology-based license plate detection from complex scenes,” in *Proc. IEEE 16th Int. Conf. Pattern Recognit.*, 2002, vol. 3, pp. 176–179.
- [16] Llorens, Marzal, Palazon, and Vilar, *Car License Plates Extraction and Recognition Based on Connected Components Analysis and HMM Decoding*, vol. 3522. Berlin, Germany: Springer, 2005, pp. 571–578.
- [17] D. Zheng, Y. Zhao, and J. Wang, “An efficient method of license plate location,” *Pattern Recognit. Lett.*, vol. 26, no. 15, pp. 2431–2438, Nov. 2005.
- [18] J. C. Poon, M. Ghadiali, G. Mao, and L. M. Sheung, “A robust vision system for vehicle license plate recognition using gray-scale morphology,” in *Proc. IEEE ISIE*, 1995, vol. 1, pp. 394–399.
- [19] Y. Wen, Y. Lu, J. Yan, Z. Zhou, K. M. von Deneen, and P. Shi, “An algorithm for license plate recognition applied to intelligent transportation system,” *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 3, pp. 830–845, Sep. 2011.
- [20] C. N. E. Anagnostopoulos, I. E. Anagnostopoulos, V. Loumos, and E. Kayafas, “A license plate-recognition algorithm for intelligent transportation system applications,” *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 3, pp. 377–392, Sep. 2006.
- [21] W. Niblack, *An Introduction to Image Processing*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1986, pp. 115–116.
- [22] J. Sauvola and M. Pietikainen, “Adaptive document image binarization,” *Pattern Recognit.*, vol. 33, no. 2, pp. 225–236, Feb. 2000.
- [23] C. Wolf, J.-M. Jolion, and F. Chassaing, “Text localization, enhancement and binarization in multimedia documents,” in *Proc. IEEE 16th Int. Conf. Pattern Recognit.*, 2002, vol. 2, pp. 1037–1040.
- [24] N. Otsu, “A threshold selection method from gray-level histograms,” *Automatica*, vol. 11, no. 285–296, pp. 23–27, 1975.
- [25] H. Caner, H. S. Gecim, and A. Z. Alkar, “The characteristic feature based on four types of structural information and their effectiveness for character recognition,” *IEICE Trans.*, vol. J68-D, no. 4, pp. 789–796, 1985.
- [26] S. Ghofrani and M. Rasouli, “Farsi license plate detection and recognition based on characters features,” *Majlesi J. Electr. Eng.*, vol. 5, no. 2, pp. 44–51, 2011.

- [27] M. S. Sarfraz, A. Shahzad, M. A. Elahi, M. Fraz, I. Zafar, and E. A. Edirisinghe, "Real-time automatic license plate recognition for CCTV forensic applications," *J. Real-Time Image Process.*, vol. 8, no. 3, pp. 285–295, Sep. 2013.
- [28] M. H. Glauberman, "Character recognition for business machines," *Electronics*, vol. 29, pp. 132–136, 1956.
- [29] M. M. Dehshibi and R. Allahverdi, "Persian vehicle license plate recognition using multiclass AdabBoost," *Int. J. Comput. Electr. Eng.*, vol. 4, no. 2, pp. 355–358, 2012.
- [30] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
- [31] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in *Proc. ECCV*, 2006, pp. 404–417.
- [32] M. Zahedi and S. M. Salehi, "License plate recognition system based on sift features," *Procedia Comput. Sci.*, vol. 3, pp. 998–1002, 2011.
- [33] O. Due Trier, A. K. Jain, and T. Taxt, "Feature extraction methods for character recognition—A survey," *Pattern Recognit.*, vol. 29, no. 4, pp. 641–662, 1996.
- [34] T. K. Bhownik, P. Ghanty, A. Roy, and S. K. Parui, "SVM-based hierarchical architectures for handwritten Bangla character recognition," *Int. J. Document Anal. Recognit.*, vol. 12, no. 2, pp. 97–108, 2009.
- [35] J.-X. Dong, A. Krzyżak, and C. Y. Suen, "An improved handwritten Chinese character recognition system using support vector machine," *Pattern Recognit. Lett.*, vol. 26, no. 12, pp. 1849–1856, Sep. 2005.
- [36] F. Camstra, "A SVM-based cursive character recognizer," *Pattern Recognit.*, vol. 40, no. 12, pp. 3721–3727, Dec. 2007.
- [37] P. Dollár, Z. Tu, P. Perona, and S. Belongie, "Integral channel features," in *Proc. BMVC*, 2009, vol. 2, pp. 1–11.
- [38] B. Yang, J. Yan, Z. Lei, and S. Z. Li, "Aggregate channel features for multi-view face detection," in *Proc. IEEE IJCB*, 2014, pp. 1–8.
- [39] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.
- [40] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, Jun. 1981.
- [41] B. K. Horn and B. G. Schunck, "Determining optical flow," in *Proc. SPIE Tech. Symp. East*, 1981, pp. 319–331.
- [42] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. IJCAI*, 1981, vol. 81, pp. 674–679.
- [43] B. McCane, K. Novins, D. Crannitch, and B. Galvin, "On benchmarking optical flow," *Comput. Vis. Image Understand.*, vol. 84, no. 1, pp. 126–143, Oct. 2001.
- [44] C. Zach, T. Pock, and H. Bischof, "A duality based approach for real-time TV-L<sub>1</sub> optical flow," in *Pattern Recognition*. Berlin, Germany: Springer, 2007, pp. 214–223.
- [45] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 1999, vol. 2, pp. 246–252.
- [46] M. Piccardi, "Background subtraction techniques: A review," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, 2004, vol. 4, pp. 3099–3104.
- [47] A. Sobral, BGS Library, Jul. 2014. [Online]. Available: <http://code.google.com/p/bgslibrary/>
- [48] J.-Y. Bouguet, "Pyramidal implementation of the affine Lucas Kanade feature tracker description of the algorithm," *Intel Corp.*, vol. 5, pp. 1–10, 2001.
- [49] J. Shi and C. Tomasi, "Good features to track," in *Proc. IEEE CVPR*, 1994, pp. 593–600.
- [50] P. H. Torr and D. W. Murray, "The development and comparison of robust methods for estimating the fundamental matrix," *Int. J. Comput. Vis.*, vol. 24, no. 3, pp. 271–300, 1997.
- [51] Y.-W. Chen and C.-J. Lin, "Combining SVMs with various feature selection strategies," in *Feature Extraction*. Berlin, Germany: Springer, 2006, pp. 315–324.
- [52] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Mach. Learn.*, vol. 46, no. 1–3, pp. 389–422, 2002.
- [53] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics Intell. Lab. Syst.*, vol. 2, no. 1, pp. 37–52, 1987.
- [54] E. Bingham and H. Mannila, "Random projection in dimensionality reduction: Applications to image and text data," in *Proc. 7th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2001, pp. 245–250.
- [55] T. H. Slobodan Ribarić and Z. Kalafati, Vehicle Plate Dataset. Nov. 2015. [Online]. Available: <http://www.zemris.fer.hr/projects/LicensePlates/english/>
- [56] R. Azad, F. Davami, and B. Azad, "A novel and robust method for automatic license plate recognition system based on pattern recognition," *Adv. Comput. Sci., Int. J.*, vol. 2, no. 3, pp. 64–70, 2013.
- [57] J. Jiao, Q. Ye, and Q. Huang, "A configurable method for multi-style license plate recognition," *Pattern Recognit.*, vol. 42, no. 3, pp. 358–369, 2009.
- [58] Z.-X. Chen, C.-Y. Liu, F.-L. Chang, and G.-Y. Wang, "Automatic license-plate location and recognition based on feature salience," *IEEE Trans. Veh. Technol.*, vol. 58, no. 7, pp. 3781–3785, Sep. 2009.
- [59] J.-M. Guo and Y.-F. Liu, "License plate localization and character segmentation with feedback self-learning and hybrid binarization techniques," *IEEE Trans. Veh. Technol.*, vol. 57, no. 3, pp. 1417–1424, May 2008.
- [60] Y. LeCun, MNIST Dataset. Dec. 2013. [Online]. Available: <http://yann.lecun.com/exdb/mnist/>



**Rahim Panahi** (M'14) received the M.Sc. degree from Sharif University of Technology, Tehran, Iran, in 2013.

Since 2011 he has been a researcher and a team leader with the Image Processing and Machine Vision Laboratory, Sharif University of Technology. His research interests include machine vision, image processing, and embedded systems.

Mr. Panahi has participated in several vision competition leagues and won several awards, including Sharif Open Robotic Cup in Machine Vision League in 2013 and 2015 and Amirkabir University of Technology Robotics Cup in 2013.



**Iman Gholampour** (M'14) received the B.S., M.S., and Ph.D. degrees in electrical engineering from Sharif University of Technology, Tehran, Iran.

He is currently an assistant professor with the Electronics Research Institute, Sharif University of Technology. He has worked with several companies and institutes on many digital signal processing and communication projects. His research interests include signal processing systems design and implementation, machine vision, image and video analysis for intelligent traffic systems, and information hiding.