ISSN 2229-5518

A Localization System for Car License Plate

Nada Najeel Kamal, Loay E. George,

Abstract— A License Plate (LP) is a rectangular metal plate contains numbers and characters or words, fixed on to the car body and it is used to identify the vehicles. In recent years, the importance of finding the accurate location of the license plate in an automatic manner has largely increased. This work aims to design a new robust method to find the accurate location of the LP in the digital image based on a set of morphological and statistical features extracted from the LP segment, then the necessary adjustments on the extracted LP are made using rotation and affine transform. The developed method is tested using with 393 images of Iraqi LPs with white and black background color; the images are taken from different distances, different view angels, and under different illumination conditions. The overall performance of the system reached to 82.44%.

Index Terms— License Plate, Rectangle, LP, Localization, Corners, Digital image, Rotation, Affine Transform, Seed filling, Linear Regression

---

1 INTRODUCTION

In the recent years, the number of vehicles has largely increased because of the growth of the urban population and the power of purchasing becomes stronger. In fact, the manual methods for dealing with a large number of vehicles are so difficult. So this leads to difficulties in the management of vehicles and also leads to many shortcomings such as the poor working environment, intensive labor, tedious work, and low efficiency [1], [2], [3].

Due to the above mentioned problems and to improve the efficiency of vehicle management and traffic control there is a demand for unmanned license plate recognition system for the automatic identification of vehicles. License plate recognition is an image processing technology used to identify vehicles by their license plates only. LP recognition is one of the most important topics of using computer vision and pattern recognition [4], [5], [6].

Many research efforts have been spent to develop the license plate localization methodology, such as: an algorithm based edge detection, image subtraction, and mathematical morphology [7], morphological operations and image projection technology [8], LP extraction based on color and geometrical features of LP [6]. For LP adjustment, line slope is obtained through least square fitting with perpendicular offsets which is used to rotate the LP in horizontal direction, the inverse affine transform used for tilt correction in vertical direction [9], the Matchstick model had been used as a horizontal tilt correction of vehicle LP [10].

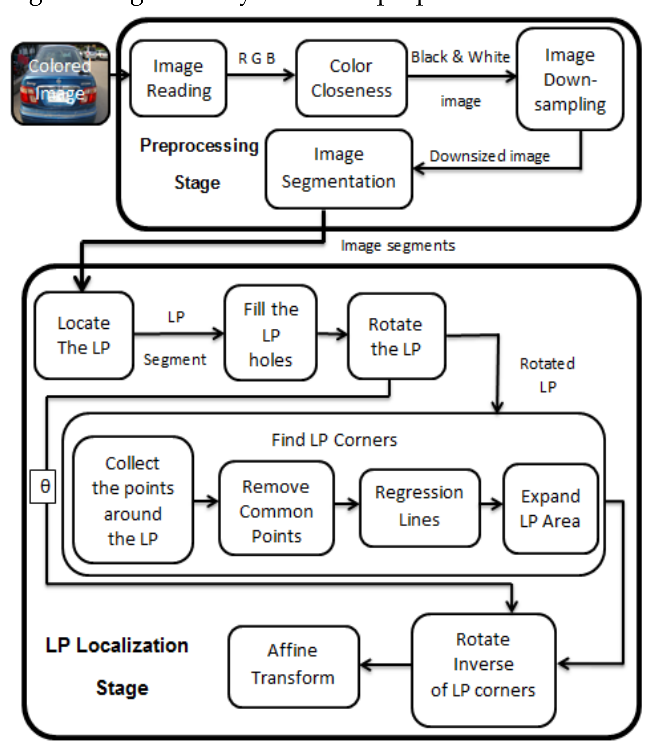
This work aims to introduce a license plate localization method; which consist of the following stages: (1) preprocessing which includes image loading, image binarization through a proposed color closeness criteria, image down sampling followed by image segmentation using seed filling algorithm, (2) A new method to find the location of the LP in the input image, based on the morphological and the statistical features extracted from the image segment. After that the holes in the LP will be filled, and its tilt is corrected.

Then, the four sides of the LP rectangle are fitted to a straight line using linear regression analysis because their intersection points, which represent the corners of the LP rectangle, is used as control point to apply the affine transform as a final adjustment on the LP segment.

2 SYSTEM LAYOUT

The general layout for the system is presented in figure (1).

Fig. 1. The general layout for the propose LP localization



system and LP adjustment

- *Nada Najeel Kamal is currently MSc student in the Department of Computer Science, Baghdad University, work under supervision of Dr. Loay E. George. E-mail: nadanajeelkamal@yahoo.com*
- *Loay E. George is currently an assistant professor in the Department of Computer Science at Baghdad University, Iraq. E-mail:loayedwar57@yahoo.com.*

The details of the proposed LP localization and adjustment are presented as the following:

2.1 Preprocessing

The preprocessing stage is necessary to prepare the digital image for further processing; this stage includes the following steps:

2.1.1 Image Reading

The input to the proposed LP localization system is a colored image of LP, fixed onto a car body. This image is a high resolution image, with BMP format. Figure (2) shows an example of this input image.

The RGB bands are extracted from the colored image and put into three different 2D arrays, one array for each band.

Fig. 2. An example of input image to the proposed LP localization system

2.1.2 Color Closeness Operation

The digital image is needed to be converted into a binary form. In fact, instead of traditional binarization process, which costs memory and time, color closeness criteria is used to determine candidate regions of LP.

The proposed color closeness works as in the following:

$$Bin\,(x,y) = \begin{cases} 0 & if\ (\Delta r > T) or\ (\Delta g > T) or (\Delta b > T) or \\ & (\Delta r + \Delta g + \Delta b) > Ta \\ 1 & otherwise \end{cases} \quad (1)$$

Where:

$$\Delta r = |Red(x,y) - TargetRed| \quad (2)$$

$$\Delta g = |Grn(x,y) - TargetGrn| \quad (3)$$

$$\Delta b = |Blu(x,y) - TargetBlu| \quad (4)$$

Bin (x,y) is the obtained binary array from the color closeness test operation. It holds 1 for pixel classified as being part of the candidate pixel(s) of LP otherwise it will hold 0 value. This image has same size of the original image. The TargetRed, TargetGrn and TargetBlu are parameters whose values are predefined to have specific values, these values depend on the dominant background color of LP being extracted. Δr, Δg, and Δb register the deflection of (RGB) color component of the tested pixel from the TargetRed, TargetGrn and TargetBlu variables respectively. T is the allowed permissible deflection of any color component value from its corresponding target color value. Ta is the allowed permissible overall deflection of the (R G B) components.

Figure (3) presents the resulted binary image Bin() from color closeness test operation.

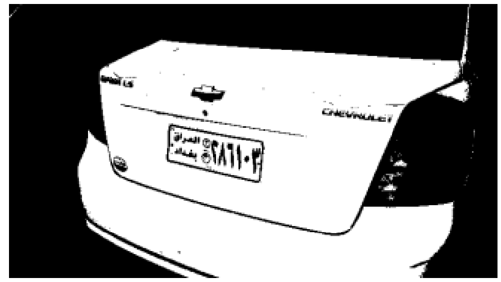


Fig. 3. LP candidate regions resulted from the proposed color closenss test operation

2.1.3 Image Down-Sampling

The R, G, B and the binary arrays have been down sampled to the ninth of its original size, to save time and memory.

For down-sampling the averaging method is used, where every nine adjucent pixels in the original image are mapped into one pixel in the downsized image (i.e. by summing the values of each 3×3 pixels and dividing them by 9). Thus the height and width of every new array will be equal to the third of its original height and width in the original image.

2.1.4 Image Segmentation

What is needed now is partitioning the image into disjoint parts or objects to decide which of these objects represent the LP, this was done by image segmentation process.

In the proposed LP localization system the seed filling algorithm is applied. According to this algorithm: every pixel in the binary image is checked to test whether it satisfies the condition of being a white pixel. Therefore, if the condition is met then this pixel is considered as a seed of a segment, then it is added to a certain buffer, and its 4-connected neighbors are checked if any one of them satisfies the condition it is added to the buffer. The values of all the registered pixels in the binary array are assigned zero value. All these steps are repeated for all the pixels added to the buffer. The repetition is stopped when no more white pixel could be added to the segment. The meaning of “adding a pixel to a certain buffer” is adding its location (i.e., X and Y coordinates) to an appropriate data structure.

Thus, the result of applying seed filling algorithm on the binary image shown in figure (4). The extracted segments from that binary will be fed to be used in the next stages of the proposed LP localization system.

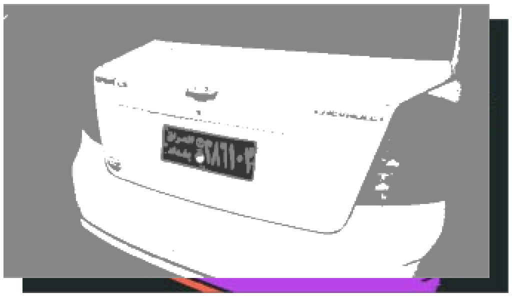


Fig. 4. Image segments after applying seed filling algorithm on the binary image

2.2 LP Localization and Adjustment Stage

In this stage, all the necessary steps to find the accurate location of the LP in the digital image and all the necessary adjustments on that LP are made. All that is done throughout the following steps:

2.2.1 Locate the LP

The goal of this step is to determine which one of segments extracted from the image segmentation process represents the LP region in the digital image. Since the LP has a rectangular shape, then, an algorithm has been proposed and applied to specify whether the tested segment has a semi like rectangular shape similar to that of LP depending on certain features computed for that segment.

The first feature that used to classify segments is the segment size feature. That means, if a segment is too small (i.e., consist of a very few number of pixels) it will be ignored; otherwise this segment will be considered as LP candidate and other features will be extracted from it in the next stages.

When the segment has an acceptable size, then the following will be applied on it:

A. The whole segment will be scanned to find four parameters values: (1) the minimum x value (Xmin), (2) the maximum x value (Xmax), (3) the minimum y value (Ymin) and (4) the maximum y value (Ymax).

B. For every column starting from Xmin to Xmax, all the lowest Y values and all the highest Y values will be stored in two different arrays; both are one dimensional (1D) arrays. The difference between the corresponding elements of these two arrays will be stored in one dimensional array with an index starting from Xmin and ending with Xmax. For more explanation to this step see figure (5).

C. Then, the values of Y differences will be sorted in an ascending order. Then the first %2 and the last %2 of the sorted values are omitted because they may hold abnormal values of the Y differences.

D. The *mean* and the *standard deviation* are calculated for the remaining %96 of the Y differences. If the tested segment has a rectangular shape, then the value of the determined standard deviation should be small. Also, its determined average object height (i.e. mean of the differences in Y) should not be small.

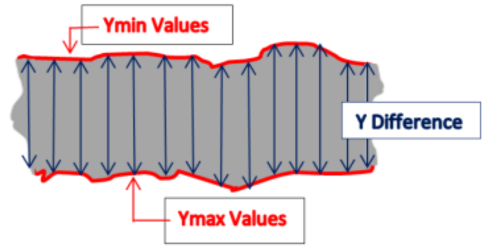


Fig. 5. The height variability of a tested segment

E. In the same way, the above mentioned (B, C, and D) steps will be applied on the X differences. For every row lay within the range [Ymin to Ymax] in the candidate segment the left most X values and the right most X values will be allocated and stored in two different 1D arrays, then the difference between the corresponding elements of these two arrays will be stored in another 1D array. The elements of this array will be sorted and only the mid %96 of the sorted values will be used to calculate the mean and the standard deviation for differences in X. If the tested segment is the LP then the average width of it should not be small.

F. Another feature that will be extracted from the candidate segment is the aspect ratio:

$$Aspect\ Ratio = Mean_{Dif\ X} / Mean_{Dif\ Y} \quad (5)$$

Where $Mean_{Dif\ X}$ and $Mean_{Dif\ Y}$ represent the mean values for the differences in X and Y respectively. If the tested segment has an aspect ratio comes within certain range of values, which depends on the style and the shape of the LP, then the tested segment is considered as good LP candidate.

G. Then the candidate segment is tested to see whether it intersects the boundaries of the image (i.e. with the first or the last row of the image; the first or last column of the image). If the segment intersects with the image boundary then it will not be considered as LP, otherwise the candidate segment (from step F) is considered as a LP segment. In general, this rule classifies the segments according to their position in the image because the LP is expected to be in the middle of the image. Figure (6) shows the candidate LP segment.

2.2.2 Fill the Holes of the LP

The goal of this step is to fill the holes founded in the LP segment. This stage is necessary to get a solid segment. To do "holes filling" task two temporary arrays are needed; their dimension would be the same as the dimension of the downsized image.

ISSN 2229-5518

Fig. 6. The dark gray segment is the one classified as LP rectangle while the white segments classified as not being the LP according to the proposed algorithm for allocating LPs.

TempA and TempB are the names of the two arrays, respectively.

First, the elements of both arrays will be initialized with zero, Then, every point in the license plate segment would be set equal to 1 in both of the temporary arrays, as in figure (7). After that the seed filling algorithm is applied on TempA array to fill the black holes of the image. Then, the size of the filled buffer is tested, if it is less than a predefined value, such as the segments appear in figure (8), then every point registered in the buffer would be set to 1 in TempB array, as shown in figures (9).

Finally, to recover the segment of the filled LP a seed filling algorithm is applied on the white pixels of TempB array. Note figure (10).

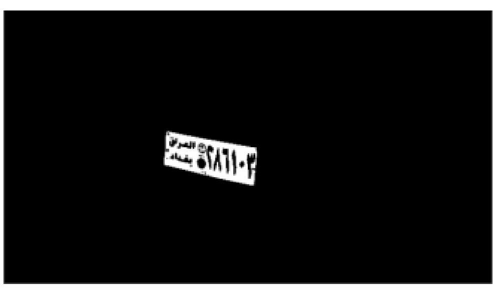


Fig. 7. Every point in the LP segment would be set to 1 in both TempA and TempB arrays

Fig. 8. The segments of the black pixels of TempA array with size less than a specific value

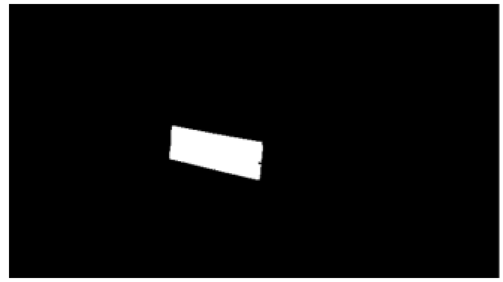


Fig. 9. TempB array contains a filled LP

Fig. 10. The solid segment of LP which is obtained after the seed filling algorithm is applied on the white contents of TempB array

2.2.3 Rotate the Plate

The goal of this stage is to adjust the angle of the LP segment by rotating it about its center.

The first step in this stage is calculating the center point of the plate because it will be considered as the pivot point of the taken rotation process. The coordinates of the center point are determined using the following equations:

$$x_c = \frac{1}{N}\sum_{i=1}^{N} \mathrm{Buf}(i).x \tag{6}$$

$$y_c = \frac{1}{N}\sum_{i=1}^{N} \mathrm{Buf}(i).y \tag{7}$$

Where x_c and y_c are the X and the Y coordinate of the center point, Buf(i) the LP collected segment points.

The two equations to rotate the LP segment are:

$$x` = (x - x_c)\cos(\theta) + (y - y_c)\sin(\theta) + x_c \tag{8}$$

$$y` = -(x - x_c)\sin(\theta) + (y - y_c)\cos(\theta) + x_c \tag{9}$$

The assessment of required rotation angle (θ) is made through the following equation:

$$\theta = \frac{1}{2}\tan^{-1}\left(\frac{2\sum_{x,y}(x - x_c)(y - y_c)}{\sum_{x,y}\{(x - x_c)^2 + (y - y_c)^2\}}\right) \tag{10}$$

Note figure (11) clarifies the LP tilt correction process.

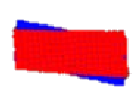


Fig. 11. The blue color represents the old alignment of the plate before tilt correction is made while the red color represents the new alignment of the rotated plate

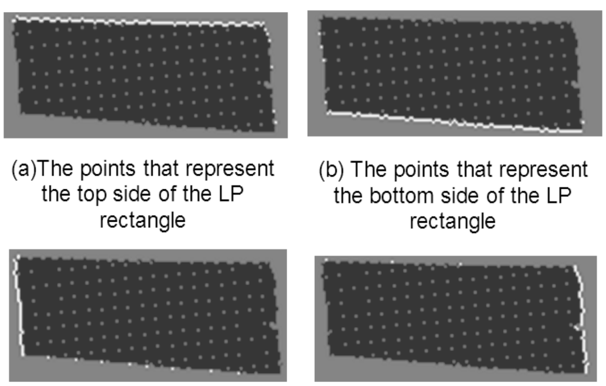
2.2.4 Allocation of LP Corners

The goal of this step is to find a set of control points which will be used to apply additional adjustment on the LP segment using affine transform. Since the LP has a rectangular shape, so its corners will be very useful as control points for making the LP figure as a perfect rectangle as possible.

The corners of a rectangle represent the intersection points of the edge lines which represent the four sides of that rectangle (i.e., top, bottom, left and right sides).

So, the four lines representing the sides of the LP segment are needed to be extracted first. To do this the following steps are made:

1. First, the whole segment will be scanned to find the four sets of LP edge coordinates; these coordinates will be stored in four different buffers (i.e. a buffer for each side: the "top", the "bottom", the "left" and the "right" buffer). The top buffer contains the minimum y values for each scanned columns of the LP segment, the bottom buffer contains the maximum y values for each scanned columns of the LP segment, the left buffer contains the minimum x values for each scanned rows of the LP segment, while the right buffer contains the maximum x values for each scanned rows of the LP segment. A focus is made on the LP segment and a closer view is used in the following illustrations and figures to make the explanation of the procedures more legible, as shown in figure (12).



(a)The points that represent the top side of the LP rectangle

(b) The points that represent the bottom side of the LP rectangle

(c) The points that represent the left side of the LP rectangle

(d) The points that represent the right side of the LP rectangle

Fig. 12. The points that represent each side of the rectangle

2. Secondly, after collecting the points of each side of the LP segment then if any one of them is found to be a common point between any two neighboring sides of the rectangle then this point should be removed from the buffers of the two sides.

   In some samples not all common points are removed because of their position in the segment. The process of discarding the highly deflected points is based on the fact that the mean and the standard deviation are calculated for the points of each side separately. Then each point belong to a certain side will be registered into a new buffer unless that point has a deflection from the mean by $\alpha * \text{Standard deviation}$ (STD). This is an iterative operation and it won't stop unless the new value of STD becomes less than certain predefined value.
3. Thirdly, since each set of points represent certain side of the rectangle; they will appear scattered around a line, so they will be used to find the best closest line pass through them using linear regression analysis method. See figure (13).

So, to find the straight line equation for both of the top side and the bottom side, the following equation is used:

$$\mathrm{y} = \mathrm{ax} + \mathrm{b} \tag{11}$$

where the (x, y) are the coordinates of the top buffer or bottom buffer points. The gradient of the line a, and the Y-axis interception value b are found using the following equations:

$$a = \frac{\mathrm{N} \sum \mathrm{x_i}\, \mathrm{y_i} - (\sum \mathrm{x_i})(\sum \mathrm{y_i})}{\mathrm{N} \sum \mathrm{x_i}^2 - (\sum \mathrm{x_i})^2} \tag{12}$$

$$b = \frac{(\sum x_i{}^2)(\sum y_i) - (\sum x_i)(\sum x_i\, y_i)}{N \sum x_i{}^2 - (\sum x_i)^2} \tag{13}$$

In same way, to find the straight line equation for both of the left and the right side, the following equation is used:

$$x = cy + d \tag{14}$$

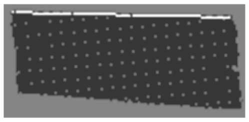
where (x, y) represent the coordinates of the left buffer and the right buffer points. The gradient of the line c, and the X-axis interception value b are found according to the following equations:

$$c = \frac{\mathrm{N} \sum \mathrm{y_i}\, \mathrm{x_i} - (\sum \mathrm{y_i})(\sum \mathrm{x_i})}{\mathrm{N} \sum \mathrm{y_i}^2 - (\sum \mathrm{y_i})^2} \tag{15}$$

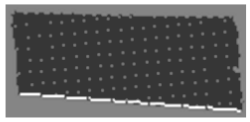
$$d = \frac{(\sum y_i{}^2)(\sum x_i) - (\sum y_i)(\sum y_i\, x_i)}{N \sum y_i{}^2 - (\sum y_i)^2} \tag{16}$$

The results of individual application of the linear regression on the collected points of the four sides of the rectangle are presented in figure (13). Note that if the common points are not removed from the rectangle sides then the lines of the linear regression will be affected.

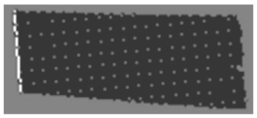
ISSN 2229-5518



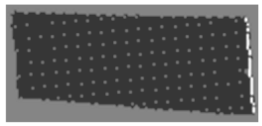
(a) The line of the top side which is gotten throught the equation $y = a_{top}x + b_{top}$



(b) The line of the bottom side which is gotten throught the equation $y = a_{bottom}x + b_{bottom}$



(c) The line of the left side which is gotten by $x = c_{left}y + d_{left}$



(d) The line of the right side which is gotten through the equation $x = c_{right}y + d_{right}$

Fig. 13. The four lines of the LP sides

4. Fourthly, LP area extension is done. Because the shape of some extracted LP segments is not a perfect rectangle, this will affect the determination of edge line positions. Many of the actually collected points will be at the outer side of its corresponding determined side line, and in such case the side lines should be adjusted to ensure high implication of the LP points inside the bounded area. That's why the line of the top side is needed to be lifted up, the line of the bottom side is needed to be moved downwards, the line of the left side is needed to be moved more to the left, and the line the right side is needed to be moved more to the right.

   In order to lift the top line, the points belonging to the top side buffer will be compared with the top line. Since the top side points should be at the top edge or below it, then the number of the top-buffer points which have y-values above the top edge line is counted, and in case this number is found large (relative to the total number of points' buffer, then the interception value b is shifted up (i.e. decreasing the value of b by 1)). The process of counting the number of top edge points above the top edge line and making shift-up for the intercept value will be repeated till reaching the status "the number of points above the top edge line become insignificant" (i.e., its ratio relative to the total number of top buffer is too small).

   Same procedure is followed to adjust the position of the bottom, left, and right edge lines but with the following differences:

   a) For the bottom edge line: the number of the points in the bottom buffer that are below the bottom edge line is counted, and the line position adjustment is done by shifting the line to the bottom side (i.e. increasing the value of b by 1) at each iteration cycle step.
   b) For the left edge line: the number of the left buffer point that are at the left side of the left edge line is counted, and the line position is adjusted by shifting the line to left side (i.e. decreasing the value of d by 1)) at each iteration cycle step.
   c) For the right edge line: the number of the right buffer point that are at the right side of the right edge line is counted, and the right line position is adjusted by shifting that line to right side (i.e. increasing the value of d by 1) at each iteration cycle step.

   The result of applying LP area expansion is shown in figure (14).

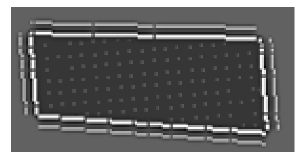


Fig.14. Moving LP lines which lead to LP area expansion

5. The next step is the determination of corner points. The position of the top-left corner point (X_{TL}, Y_{TL}) represents the intersection point between the two neighboring sides (i.e. left & top edges) of a rectangle. The top edge line has the equation $Y = a_T X + b_T$ and the left edge line has the equation $X = c_L Y + d_L$. So, to find the coordinates value of the top-left point X_{TL}, X_{TL} , the fact that this point should satisfy both edges equations is utilized, such that:

$$X_{TL} = \frac{d + cb}{1 - ac} \tag{17}$$

$$y_{TL} = \frac{ad + b}{1 - ac} \tag{18}$$

By following similar steps the coordinate of the top-right corner (X_{TR}, Y_{TR}), the bottom left corner (X_{BL}, Y_{BL}), and the bottom right corner (X_{BR}, Y_{BR}) could be determined using the following three sets of equations:

a. The equations for the top right corner :

$$X_{TR} = \frac{b_T c_{R+} d_R}{1 - a_T c_R} \tag{19}$$

$$y_{TR} = \frac{a_T d_R + b_T}{1 - a_T c_R} \tag{20}$$

b. The equations for the bottom right corner :

$$X_{BR} = \frac{b_B c_{R+} d_R}{1 - a_B c_R} \tag{21}$$

$$y_{BR} = \frac{a_B d_R + b_B}{1 - a_B c_R} \tag{22}$$

c. The equations for the bottom left corner :

ISSN 2229-5518

$$X_{BR} = \frac{b_B c_L + d_L}{1 - a_B c_L} \quad (23)$$

$$y_{BL} = \frac{a_B d_L + b_B}{1 - a_B c_L} \quad (24)$$

6. The position of the determined four corner points in the downsampled image should be mapped to allocate its position in the original (high resolution image). This could be done by scaling up the determined coordinates value by 3, then rotate them back by the same amount of the rotation angle, but with adverse manner. Thus, the negative value of estimated angle from the rotation step is applied. Then the new X andY coordinate of the four corner points will be calculated using equations (8) and (9).

2.2.5 Affine Transform

The affine transformation will be applied on the LP segment, extracted from the original image, to remove the shearing effect that is due to existing perspective effects in the input image.

The affine transform is applied through the following steps:

A. The length of each side of the rectangle is calculated using the Euclidean distance between any two neighboring corner points of the rectangle. So, the following four distances (denoted by d_1, d_2, d_3 and d_4) represent the length side of the LP are determined using the following equations:

$$d_1 = \sqrt{(x_{TL} - x_{TR})^2 + (y_{TL} - y_{TR})^2} \quad (25)$$

$$d_2 = \sqrt{(x_{BL} - x_{BR})^2 + (y_{BL} - y_{BR})^2} \quad (26)$$

$$d_3 = \sqrt{(x_{BL} - x_{TL})^2 + (y_{BL} - y_{TL})^2} \quad (27)$$

$$d_4 = \sqrt{(x_{BR} - x_{TR})^2 + (y_{BR} - y_{TR})^2} \quad (28)$$

B. Wherever the location of the extracted LP occurs in the input image, it will be affined transformed and resampled; to do that the new position of the LP corners will be mapped to form an exact rectangular shape because it will be treated as a new image that have a new width and new height. Its new width and height are denoted by nW and nH, respectively. They are calculated according to the following criteria:

If $d_1 > d_2$ then $nW = d_1$ otherwise $nW = d_2$
If $d_3 > d_4$ then $nW = d_3$ otherwise $nW = d_4$

To transform the position of the corners the following mapping conditions are applied:

$(x_{TL}, y_{TL}) \rightarrow (0,0)$
$(x_{TR}, y_{TR}) \rightarrow (nW - 1, 0)$
$(x_{BR}, y_{BR}) \rightarrow (nW - 1, nH - 1)$
$(x_{BL}, y_{BL}) \rightarrow (0, nH - 1)$

C. The affine transform is applied usig the following two linear mapping equations:

$$x`_i = a_0 + a_1 x_i + a_2 y_i \quad (29)$$

$$y`_i = b_0 + b_1 x_i + b_2 y_i \quad (30)$$

These two equations contain six unknowns they are a_0, a_1, a_2, b_0, b_1 and b_2. To obtain their values, the least squares error (χ^2_{min}) criteria is adopted; that is the partial derivative of the total error (χ^2) with respect to each of the unknown variables is taken and set equal to zero, then the two set of linear equations is solved using Cramer's rule. The result of applying affine transform on the adopted sample image is shown in figures (15) and (16) both images clarify the ability of affine transform to remove shearing from the LP segment.

Fig. 15. The image resulted from applying the affine transformation; it is free of shearing image

(a) The Input Image of another sample

(b) The affine transform for the LP shown in (a)

Fig. 16. The effect of affine transform

3 Results

The results of the tests sets conducted on the established proposed system are presented in this section. It contains the necessary information about the used software, hardware frame work, and the test materials. Also, the overall system behavior is clarified.

3.1 Software and Hardware Framework

All tests are done using a laptop with Intel(R) Core(TM) i5 CPU 2.53 GHz with 4.00 GB RAM. The programming language Visual Basic-6 is utilized to build the application program. The operating system is Windows 7 Ultimate, Service Pack1. The LP images snapshots have been taken using SONY camera, DSC-W230 Model.

3.2 Test Materials of the Proposed System

A set of 393 bitmap digital images for different LPs, fixed onto the car body, are taken. These images are classified into six groups, as shown in table (1). The LP

Table (1) Lists the taken car images classified into groups according to some of the LP attributes.

Name of group	Number of samples	Background color	Note
Hard	103	White	2001 Style
Soft	142	White	2001 Style
Frontal	30	White	2001 Style
Faded and blurred	11	White	2001 Style
Black-Group1	54	Black	Baghdad Province
Black-Group2	53	Black	Baghdad Province
Total	393	Black and White	

snapshots are taken from different distances, (between 3 meters to 60 centimeters), and with different angles, ranging from small angles close to zero degree (where the LP appears directly in front of the viewer) to difficult angles with degree more than 45˚ degrees; and under different illumination conditions, which cause a wide variation in the LP background color, beside to of the appearance of shadow in some parts of LP while the other parts are highlighted. The dimensions of most snapshot images are 1920×1080 pixels, (except 13 their dimensions are taken 1920×1440 pixels).

The “Hard” class, in table (1), consists of images convey large degradation in LP area (Like, faded paint of LP characters, and/or very inclined angles). The “Soft” class consists of images have small distortions. Both of the “Hard” and “Soft” snapshots are taken for LPs fixed at the back side of the car. The class “Black-Group1” consists of the car images taken for the LPs have black background color and convey little degradations; while “Black-Group2” holds LP image suffer from big degradations (like, damaged LP and peeled off paint). The images of LP with black background are taken for the LP fixed on back and the front sides of the car. The “Frontal” class holds the snapshots taken for the LPs fixed at the front side of the vehicle. The “Faded and Blurred” class holds the LP images suffer from faded painting of characters/numerals and seriously blurred LPs images. Test results are presented in Table 2. The system gives %82.44 of success rate.

Table (2) LP Success Rates

Name of group	Number of samples	Success Rate Percentage
soft	142	93.66%
Hard	103	79.61%
Frontal	30	100%
Faded and blurred	11	100%
Black-Group1	54	64.81%
Black-Group2	53	56.60%
Total	393	82.44%

4 Conclusions and Future Work

So, in conclusion a robust method to find the location of the license plate in the digital image is presented. The proposed method includes preprocessing stage, LP localization and adjustment stage.

A method for image binarization depends on color closeness criteria have been proposed. This method costs less memory and leads to short time of execution.

The LP localization method depends on morphological (i.e., the rectangular shape of the LP) and statistical features extracted from the segments of the image, such as segment’s size, mean and the standard deviation for the differences between Y min and Y max values for every column in the image, in addition to the aspect ratio.

A method to find the position of the corners of the LP have been introduced, in which the corners represents the intersection points of the lines that represents the four sides of the LP rectangle, those lines are found through the linear regression calculations.

The license plate segment is rotated through rotation operation after the angle is estimated; also, the license plate image is adjusted through affine transform to remove shearing from LP images.

The presented method is tested using 393 LP images, classified into different groups based on LP attributes, with white or black background color, taking in consideration different distances, angles of view, and illumination conditions. Thus, the overall performance reached to %82.44.

The suggestion for future work includes: for preprocessing stage contrast stretching or histogram equalization may give higher performance for LP localization stage. Any style or color of LPs could be used as input for the proposed system. Unifying the distance, angle of view and illumination condition as much as possible will leads to better system performance. The numerals and the characters of the LP could be segmented and recognized through using any segmentation or recognition algorithm.

References

[1] Wang, Z., and Shaozi, L., "Research and Implement for Vehicle License Plate Recognition Based on improved BP Network", International Conference on Computer and Communication Technologies in Agriculture Engineering, IEEE, PP. 101-104, 2010.

[2] Yang, G., Zhu, Q., Chi, J., and Zhuang, X., "A License Plate Recognition System Based on GFNN and DSP”, IEEE Conference on Technologies for Homeland Security, IEEE, PP. 160-164, 2008.

[3] Mohsin, A., Hassin, A. H., and Abdul Jaleel, I. Q., "An Automatic Recognizer for Iraqi License Plates Using ELMAN Neural Network", Journal of Software Engineering and Applications, SciRes, Vol.3, No. 12, PP. 1163-1166, December, 2010.

[4] Yoon, Y., Ban, K., Yoon, H., and Kim, J., "Blob Extraction based Character Segmentation Method for Automatic License Plate Recognition System", International Conference on Systems, Man, and Cybernetics, IEEE, PP. 2192-2196, 2011.

[5] Sivanandan, S., Dhanait, A., Dhepale, Y., and Saiyyad, Y., "Automatic Vehicle Identification Using License Plate Recognition for Indian Vehicles", International Journal of Computer Applications, PP. 23-28, 2012.

[6] Deb, K., Lim, H., and Jo, K., "Vehicle License Plate Extraction Based on Color and Geometrical Features", IEEE International Symposium on Industrial Electronics,Seoul Olympic Parktel, Seoul, Korea, IEEE, PP. 1650-1655, July 5-8, 2009.

[7] Mai, V. D., Miao, D., Wang, R., and Zhang, H., "An Improved Method for Vietnam License Plate Location, Segmentation and Recognition", International Conference on Computational and Information Sciences, IEEE, PP. 212-215, 2011.

[8] Hai, L., and Shrestha, G., "Real Time License Plate Recognition at the Time of Red Light Violation", International Conference on Computer Application and System Modeling (ICCASM), IEEE, PP. V2-243-V2-247, 2010.

[9] Deb, K., Vavilin, A. and Jo, K., "An Efficient Method for Correcting Vehicle License Plate Tilt", IEEE International Conference on Granular Computing, IEEE, PP. 127-132, 2010.

[10] Du, Y., and Wu, Z., "An Approach to Horizontal Tilt Correction of Vehicle License Plate Based on Matchstick Model", The 1st International Conference on Information Science and Engineering (ICISE), IEEE, PP. 1124-1128, 2009.