A Robust Method for Arabic Car Plates Recognition and Matching Using Chain Code

R. F. Mansour
Department of Science and Mathematics, Faculty of Education, New Valley, Assiut University, EL kharaga, Egypt

Abstract This paper provides a new and fast method for matching and recognition of characters in Arabic license plate images. For this purpose, various methods have been proposed in literature. However, most of them suffer from: sensitivity to non-uniform illumination distribution, existence of shade in license plate, license plate color and the need for receiving an exact image of the license plate. The main contributions of our work include (1) chain code use to bounded the shape and distinguishing similar characters by local structural features. The moving window matching algorithm has been implemented. The distance measure (squared Euclidean distance) technique has been used for measuring the similarities between the moving window and the plate image. (2) Developing a system architecture combining statistical and structural recognition methods. We tested the method with 300 of plate images captured in different environments from real applications. The result yield 93.93% recognition accuracy.

Keywords License Plate Recognition, Template Matching, Moving Window, Segmentation and Chain Code

1. Introduction

Automatic license plate recognition (ALPR) is one of the most important aspects of applying computer techniques towards intelligent transportation systems. These systems attempt to facilitate the problem of identification of cars, via various techniques which mainly rely on automated (rather than manual) algorithms.

Accurate license plate locating is very important for post process, because sub-image drop off disturb from non-plate image region and the post process use sub-image as input to recognition. There are many papers[1-4] discussed the locating methods, but it need to be improved for better application.

Image processing is one of these techniques which deals with images and/or video sequences taken from vehicles. One unique property that can be taken into account for identifying all vehicles is their license plate number. Numerous applications, such as automatic toll collection[5], criminal pursuit[6] and traffic law enforcement[7], have been benefited from it[8-14]. Although some novel techniques, for example RFID (radio frequency identification), WSN (wireless sensor network), etc., have been proposed for car ID identification, LPR on image data is still an indispensable technique in current intelligent transportation systems for its convenience and low cost. Recognition algorithms reported in previous researches are generally composed of three major parts; license plate detection (LPD), character segmentation and character recognition. Segmentation and recognition of characters in Iranian license plates, which correspond to the second and third above mentioned stages, is the purpose of this research. Camera angle and different distances to the plate, non-uniform illumination in the image (such as light and shade), the plate different colors and availability of an inexact band, including the plate, are the major problems encountered. Some different methods are proposed for the segmentation of characters from license plates such as: global optimization procedure[8], image projection[15-17] and the Hough transform[18]. As most of the algorithms need a binary image of license plate, at first we explain the traditional methods of thresholding and binarization of an image and then review the segmentation and recognition techniques of the characters. The brightness distribution of various positions in a license plate image may vary because of the condition of the plate and the effect of lighting environment. Since the binarization with one global threshold cannot always produce useful results in such cases, adaptive local binarization methods are used[19,20]. In many local binarization methods, an image is divided into $m \times n$ blocks, and a threshold is chosen for each block.

In Arabic license plates has main features country name and numbers these features indicated in three rows or columns. Figure 1 show samples of plates from Egypt and kingdom of Saudi Arabia. Arabic license plates recognition based on the segmentation of plate and analyzing the segments. There are many factors that make the character segmentation task difficult such as image noise, inappropriate
plate frame, rivets, space mark, shapes, plate rotation, mixed
digitals and characters and various degrees of illumination.

The paper is organized as follows. In Section 2, the pro-
posed method is described step by step. Sections 3 and 4
using chain code for feature extraction and using chain code
and recognizing characters by statistical method respectively.
Using the moving window for matching car plates is discuss
in section 5. The experimental results and evaluation of the
algorithms are given in section 6. Finally, the paper is con-
ncluded in Section 7.

Figure 1. show samples of plates from Egypt and kingdom of Saudi
Arabia

In this research, we are proposing an efficient technique
that recognizes Arabic car plates which contain mixed
of characters and numbers will be used as testing images.

2. Preprocessing

In order to recognize characters accurately, preprocessing
the images, such as skew correction and normalization,
has to be performed. In this section, we briefly introduce
skew correction and normalization operations

2.1. Skew Correction

Character recognitions are generally very sensitive to
skew. Therefore, skew detection and correction are critical.
We propose here a least-square based skew detection method.
Suppose the binary image

\[ F = \{F(i, j), i = 1, \ldots , l, j = 1,2, \ldots , j\} \]

is defined as follows:

\[ f(i, j) = \begin{cases} 0 & \text{white} \\ 1 & \text{black} \end{cases} \]

Step 1: Find out all the connected regions. Let the con-
ected region sets be \( \{c_1, c_2, \ldots , c_n\} \), and \( c_i \) has a height
\( H_i \) and width \( W_i \)

Step 2: For each connected region, check if it is a "valid"
region. A connected region \( c_i \), is said to be "valid" if \( T_{\text{min}} < \frac{W_i}{H_i} < T_{\text{max}} \). Where \( T_{\text{min}} \) and \( T_{\text{max}} \) are predefined val-
ues. As for a standard, the rate between width and height of
each character ranges from 0.3 - 0.8 in a given Arabic car
plate. Therefore, in our implementation, \( T_{\text{min}} \) and \( T_{\text{max}} \)
are set to 0.3 and 1.0 respectively.

Step 3: For each "valid" connected region, calculate its centered
\( (G_{ix}, G_{ij}) \):

\[ G_{ix} = \frac{\sum_{(x,y) \in c_i} x \cdot c_i(x, y)}{\sum_{(x,y) \in c_i} c_i(x, y)} \]
\[ G_{ij} = \frac{\sum_{(x,y) \in c_i} y \cdot c_i(x, y)}{\sum_{(x,y) \in c_i} c_i(x, y)} \]

Step 4: Perform the skew correction by least-squares
based on the centered \( (G_{ix}, G_{ij}) \). Approximate
sets \( (G_{ix}, G_{ij}) \) by least-square, and compute the skew an-
gle \( \theta \). Given that \( F(x, y) \) is the skew image and \( F(x', y') \)
the corrected image, the skew correction equation is defined
as following:

\[ \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \]

Figure 2 show image before and after skew correction. After skew
correction of the character images, characters are
segmented from the corrected images.

2.2. Size Normalization

Characters segmented from different car plates have dif-
ferent sizes. A linear normalization algorithm is applied to
the input image to adjust to a uniform size. Assume the
horizontal and vertical projections of the original image \( F \) be
\( h \) and \( v \), respectively. The normalization position \((m, n)\)
of \((i, j)\) is obtained by

\[ m = \frac{\sum_{k=1}^{i} h(k) \times M}{\sum_{k=1}^{m} h(k)} \]
\[ n = \frac{\sum_{k=1}^{j} v(k) \times M}{\sum_{k=1}^{n} v(k)} \]

where \( M, N \) is the height and width of normalized
Figure 3 shows some normalization results.

3. Feature Extraction Using Chain Code

The first step of the construction of the chain code is to
extract the boundary of the image. Chains can represent
the boundaries or contours of any discrete shape composed of
regular cells. In the context of this work, the length \( l \) of each
side of cells is considered equal to one. These chains rep-
resent closed boundaries. Thus, all chains are closed. Extract-
ing the contour depends on the connectivity. In the content
of this paper we use pixels with four-connectivity. The simplest
contour following algorithms were presented by Duda and
Hart[21]. Thus using these algorithms it is possible to rep-
resent shape contours by only two states: left turn (rep-
resented by “1”) and right turn (represented by “0”). The
The abovementioned process produces a chain composed of only binary elements. Figure 4 illustrates the contour following on an image composed of pixels. This contour was obtained according to the following algorithm:

**Figure 4.** Example of a contour following on a digital figure

![Diagram of figure 4](image)

Figure 5. Directions of the neighbors: (a) 4-connected; (b) 8-connected

Scan the image until a figure cell is encountered. Then: If you are in a figure cell turn left and take a step. If you are in a ground cell turn right and take a step. Terminate when you are within one cell of the starting point. In this paper we proposed a new algorithm to find the contour of a binary image and use this contour to obtain the chain code. Since we use pixels with 4-connectivity, the four neighbors of any point can be represented by directions as illustrated in figure (5a). To find the contour of a binary image we apply the following algorithm:

**Step 1.** For all pixels with value 0 (black) in the image, set the pixel that has the direction 2 in 4-connected to 0.

**Step 2.** In the new image (i.e., image obtained from Step 1, also, for all pixels with value 0, set the pixel that has the direction 1 in 4-connected to 0.

**Step 3.** Remove the old pixels (in the original image) that have 8-connected as shown in figure (2b) and do not satisfy the conditions shown in figure 6.

![Diagram of figure 6](image)

Figure 6. Four conditions to remove the old pixels.

we can apply this algorithm to real images to obtain their contours. Figure 7 shows a original plate and its contour.

**Figure 7.** show the applied chain code method to real plate

![Image of plate](image)

3.1. Segmentation

The technique applied in this phase will be the partial segmentation technique. The threshold images of car plates will be segmented to several regions depending on how many characters consist in the car plates. For example, car plate with 7 characters will be segmented into 7 regions.

The first step in segmentation process is to cutoff the background from each character and number from the license plate. We use vertical scanning to detect first and last columns for each character and number before horizontal scanning as explained in algorithm. Vertical scanning is done before horizontal scanning because if skewness is present in the input image then its effect will be minimized. This improvement will reduce the error of first and last columns and rows. Vertical scanning (column by column) will be done to detect the first and last columns or each component and cut the area in between to separate the license information from background. The vertical segmentation and horizontal segmentation outputs are shown in Figure 8.

**Figure 8.** shown the output segmentation (a) vertical segmentation and (b) horizontal segmentation

![Image of segmentation](image)

3.2. Thinning

In this phase, each region will undergo a thin line formation in order to find the most successful thin line of characters. The output from this process will be a standard size of thinning image regardless of its various sizes and font types and known as template. A traditional thinning technique, the Hilditch technique is proposed to be applied in this phase because of its capability in reducing processing time. Hilditch is a skeletonization method that used non-recursive, recursive and partially recursive neighborhood[22]. A process of resizing an image will be applied to the thinning image and here, the nearest neighbor method to make each character’s size constant will be used because it is the fastest technique compared to other.

4. Recognizing Characters by Statistical Method

After preprocessing, the input character image is first recognized by statistical methods. In our approach, four
sub-classifiers recognize the character independently, and
their recognition results are combined using the Bayes
method[23,24]. To recognize similar characters in our car
plate character recognition system, it is important to extract
stable and representative structure features. Fortunately,
different similarity sets have different structural features.
Taken for example, we will discuss in this section how to
distinguish most frequently occurring similarities: "8" and
"B" by using left edge contour feature. First, we give the
definition of edge point. A point \((i,j)\) with \(F(i,j)\) satisfies
the following conditions: \(F(i,j)|\{0\leq i<1,0\leq j<1\}\) = 1 is called an edge point.
The left edge sequence of the input image \(F\) is defined,
which is a left edge point set \(\{F(i,k)|i=1,2,\ldots,M\}\). For
point \(F(i,k_i)\), the value \(k_i\) can be obtained by the following
process: In the \(i\)-th row, the column index \(j\) moves from left
to right until \(F(i,j)\) is a left edge point, and \(k_i=j\) (the last
value). Then the curve direction of edge point \((i,j)\) is defined
as follows:
\[
dir_i = \begin{cases} 
1 & f(i,j) - f(i-1,j) > 0 \\
0 & f(i,j) - f(i-1,j) = 0 \\
-1 & f(i,j) - f(i-1,j) < 0 
\end{cases}
\]
Let the curve direction sequence be \(\{dir_i|i=1,2,\ldots,M\}\), if \(dir_i\) and \(dir_j\) satisfy the following conditions:
\[
dir_i \times dir_j > 0, \\
dir_i = 0 \quad \forall i = 1,2,\ldots,k-1, \\
\|k\| < W
\]
Then the sequence is said to contain a curve point. The
left edge contour feature is calculated as follows:
Step 1: Obtain the left edge sequence \(\{F(i,k_i)|i=1,2,\ldots,M\}\) of the input
image \(F\).
Step 2: Compute the curve direction of the left edge se-
quence. \(\{dir_i|i=1,2,\ldots,M\}\)
Step 3: Compute the total of the curve point set (denoted
by total curve ) from the direction of the left edge sequence.
Step 4: Approximate the left edge sequence by using a
least-square method. Compute the approximate error (de-
noted by error )
The two types of structural features are feed into a binary
decision tree to distinguish "8" and "B". The decision tree
doesn’t always give the precise result. If the decision rejects
the character, the final recognize result is set back to pre-
processing stage. In our system, several parameters(such as
\(W\) and decision parameters used in binary decision tree need
to be predefined, and they can be obtained by some optimi-
ization algorithm, and we use genetic algorithm for optimi-
ization parameters.

5. Moving Window with Template
Matching
Moving window using the template matching method
(sum of squared differences) is a common and practical
 technique utilized in many pattern recognition applica-
tions[25,26]. The template matching method gives high
recognition accuracy and reduces the processing time com-
pared to other methods such as cross-correlation. The
applied method computes the sum of squared differences in
each position while the word image we want to recognize
moves over the background template. The point where the
sum of squared difference is less than a preset threshold will
be considered as the point of matching. The proposed mov-
ing window template matching scheme is illustrated in figure
5. First a window containing an object with size smaller than
that of the main image is defined. Only a portion of the image
is visible through this window. The template matching
function is performed between the object in the window and
the corresponding area of the image. Then the window is
shifted and the template matching function is carried out
between the object in the window and the new part of the
image visible through the window. Thus, the window is
moved left to right and top to bottom in single pixel dis-
placement steps until the entire image is covered and tem-
plate matching is carried out for all different window posi-
tions. Mathematically, distance measure is a measure of the
similarities or shared properties between two signals. The
distance metric commonly used is the Minkowski metric
\(d(x,y)\) as follows:
\[
d(x,y) = \left( \sum_{i=1}^{N} |x_i - y_i|^r \right)^{1/r}
\]
where \(x, y\) are two \(N\) dimensional feature vectors, and \(r\) is
a Minkowski factor. And when \(r = 2\), it is actually Euclidean
distance.
In our case there are two discrete signals \(f, t\) represent two
images denoting the object to be searched and the template
respectively. The object is of dimension \(I \times J\) pixels and the
template is of dimension \(M \times N\) see Figure 9.
\[
d^2(x,y) = \sum_{i=0}^{I-1} \sum_{j=0}^{J-1} \left( f(i,j) - t(x+i,y+j) \right)^2,
\]
x=0,1, … , M, y=0,1, … , N-J
where the sum is over \(i, j\) under the window containing the
feature positioned at \(x, y\). To reduce the computing time, the
above equation can be simplified to Manhattan distance:
\[
d(x,y) = \sum_{i=0}^{I-1} \sum_{j=0}^{J-1} |f(i,j) - t(x+i,y+j)|
\]
6. Experiments

We apply the above algorithm to our database of car license plates, all of which are real scene images acquired by CCD cameras. They contain cars in different conditions, such as different illumination and different visual angle. Figure 1 shows some test images in our experiment. Table 1 shows the result of our experiment. From it we can see that, in most cases the car license plates can be detected effectively. Our algorithm failed in 16% cases. The failure region there is a dark shade. the algorithm can be applied in a certain range of the size of license plate which is according to the concrete situations. In different situations, we can adjust the size of window to coincide with it. The time spent to run the algorithm depends on the size of windows and the size of the image processed. Our experiment proposed algorithm has been implemented in Matlab software, A variety of country names, characters and numbers are used through this primary test. Figure 1 shows some of the characters and numbers images included in the database with two different fonts to match the fonts used in Egypt and Saudi Arabia license plates. The software program has been improved several times to reduce the processing time to the minimum value.

<table>
<thead>
<tr>
<th>Character District</th>
<th>Letters District</th>
<th>Letter Numeral Mixing Zone</th>
</tr>
</thead>
<tbody>
<tr>
<td>300</td>
<td>300</td>
<td>1500</td>
</tr>
<tr>
<td>The number of correctly identified Recognition Ratio</td>
<td>290</td>
<td>287</td>
</tr>
</tbody>
</table>

A large number of Egyptian license plates acquired in different environments have been used in the test phase to determine the most suitable threshold for similarity as shown in Figure 10(a). Another number of Saudi license plates have been acquired and processed in the test phase Figure 10(b). It can be easily noted that the distance measures for the Saudi plates have higher peaks than that for Egyptian ones. It can be referred to the fact that the Egyptian plates have colored background while the Saudi plates have white background. The size of the moving window is an important criterion in determining the system performance.

The threshold corresponding to minimal error has been determined. The relation between the minimum error (minimum distance between the object in the moving window and corresponding area in plat's image) and the standard deviation of the distance measure is control factor for determining the threshold. It has been found that $R = 0.4$ is safe threshold.

$$d_{\min} = \frac{\sum_{x=0}^{W-1} \sum_{y=0}^{H-1} (d(x, y) - \bar{d})^2}{((M - 1)(N - 1) - 1)}$$

Where $d_{\min}$ and $\bar{d}$ are the minimal error and the average error respectively.

![Figure 10. Illustration of the threshold determination](a)

![Figure 10. Illustration of the threshold determination](b)

Figure 11. The comparison of the average ROC curves for our method and others
We evaluate the proposed method and compare with different methods [27,28] with respect to the criteria of the matching accuracy and efficiency at plat car image. Therefore, we report the ROC in Figure 11. Clearly, our method that based on features extraction using chain code and matching is observed to perform better than the other techniques.

7. Conclusions

In this paper, we propose a robust method for car plate character recognition. The main contributions of our work include (1) chain code used to bounded the shapes and distinguishing similar characters by local structural features. The moving window matching algorithm has been implemented. The distance measure (squared Euclidean distance) technique has been used for measuring the similarities between the moving window and the plate image. (2) Developing a system architecture combining statistical and structural recognition methods. We tested the method with huge number of plate images captured in different environments from real applications, and proven to be successfully in commercial car plate recognition. Compared with other methods, our method is more effective and robust. The method is applied on a test database of 300 samples of extracted license plate images captured in outdoor environment. The result yield 93.93% recognition accuracy. We believe that our method can be extended to other OCR application fields. The mixed characters of Arabic car plates is difficult, which sometimes may be very illegible even by human beings. How to recognize it is a challenging research topic.

REFERENCES


[22] Lucas J. Van and Ben J.H. Verwer, "A Contour Processing


