DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

LICENSE PLATE SEGMENTATION IN IMAGES

by
Emre AKSOY

August, 2008
İZMİR

LICENSE PLATE SEGMENTATION IN IMAGES

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Master of Science
In Electrical and Electronics Engineering

by
Emre AKSOY

August, 2008
İZMİR

M.Sc THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"LICENSE PLATE SEGMENTATION IN IMAGES"** completed by **EMRE AKSOY** under supervision of **ASST. PROF. DR. HALDUN SARNEL** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Asst. Prof. Dr. Haldun SARNEL

Supervisor

Asst. Prof. Dr. Yavuz ŞENOL

(Jury Member)

Asst. Prof. Dr. Adil ALPKOÇAK

(Jury Member)

Prof. Dr. Cahit HELVACI
Director
Graduate School of Natural and Applied Sciences

ACKNOWLEDGMENTS

Initially, I would like to thank my supervisor, Asst. Prof. Dr. Haldun SARNEL who gave me an opportunity to do this research in which I am really interested, and make this thesis possible in the Department of Electrical and Electronics Engineering at Dokuz Eylül University, Izmir. His support, guidance and encouragement stimulated me throughout the duration of my Master of Science study.

I would also like to thanks to my family. To each of them I owe a great debt of gratitude for their patience, encouragement, support, motivation.

Finally, I would like to give my special thank to my wife, Nehir.

Emre AKSOY

LICENSE PLATE SEGMENTATION IN IMAGES

ABSTRACT

Vehicle License Plate Recognition Systems are used for automated recognition of vehicles. This system can be efficiently employed for traffic monitoring of roads, automated toll collection, entry surveillance of hospitals, military bases, official buildings, etc. The first and the important step in license plate recognition is the license plate segmentation. In this thesis, license plate segmentation systems for the vehicles which have license plate that conforms to Turkish License Plate Standards have been discussed.

The first method introduced in this thesis is based on The Optical Character Recognition (OCR) system. In this method, recognizing the segmented characters in the images are aimed and histogram equalization and smearing algorithms are used for implementation of this system.

The second method introduced in this thesis is based on determining high frequency coefficients within the image to find out license plate by wavelet transform. Performances of Haar and Daubechies filters are compared on that basis.

The last method introduced in this thesis is based on edge detection method. Edge detection is done by Sobel operator and a moving frame is used to determine the candidate areas with a dimension similar with license plate. Finally, to find out the license plate, the area having the highest edge pixels is taken.

Keywords: License Plate Recognition, License Plate Segmentation, Wavelet Transform, Smearing Algorithm

SAYISAL GÖRÜNTÜLERDEN PLAKA BÖLGESİNİN AYRIŞTIRILMASI

ÖZ

Motorlu araçlar için plaka tanıma sistemi araç tanıma otomasyonunda kullanılan bir tekniktir. Trafik denetleme, gişe otomasyonu ve denetimli saha giriş kontrolü (hastane, askeri tesis vb.) uygulamalarında verimli olarak kullanılır. Araç plaka tanımanın en önemli adımlarından biri plaka yerinin saptanmasıdır. Bu çalışmada, Türk Plaka Standartlarına uyan plakaya sahip araçlar için plaka yeri saptama sistemleri incelenmiştir.

Bu çalışma kapsamında bulunan yöntemlerden ilki Optik Karakter Tanıma' ya (OCR) dayanmaktadır. Bu yöntemde resimlerdeki karakterlerin belirlenmesini hedeflenmiş ve yöntem gerçeklenirken smearing algoritması, yerel ve genel eşikleme fonksiyonlarıyla kullanılmıştır.

İkinci sırada araç resimlerinde plakanın yerini belirleme amaçlı yüksek frekans bileşenlerinin dalgacık dönüşümü ile bulunması yöntemi incelenmiştir. Çalışmada bu amaçla Haar ve Daubechies filtreleri kullanılmış ve bu filtrelerin performansları karşılaştırılmıştır.

Bu çalışma kapsamında incelenen son teknik dikey kenar çıkarımına dayanmaktadır. Sobel operatörü ile dikey kenar çıkarımı yapıldıktan sonra resim içinde plaka boyutundaki bir çerçeve gezdirilerek aday bölgeler belirlenir. Bu bölgeler içinde en fazla dikey kenar ayrıtı içerenn bölgenin seçilmesiyle plaka resim içinde bulunmaktadır.

Anahtar sözcükler: Otomatik Plaka Tanıma, Plakanın Resimde Bulunması, Dalgacık Dönüşümü, Smearing Algoritması

CONTENTS

Page

CHAPTER ONE

INTRODUCTION

License plate segmentation is one of the modules of License Plate Recognition (LPR). LPR is an image-processing technology used to identify vehicles by their license plates (Hofman, 2004). Other names of LPR are NPR (Number Plate Recognition), AVI (Automatic Vehicle Identification), CPR (Car Plate Recognition), ALPR (Automatic License Plate Recognition), LAPI (Lecture Automatique de Plaques d'Immatriculation), and OCR (Optical Character Recognition) for Cars (Wikipedia).

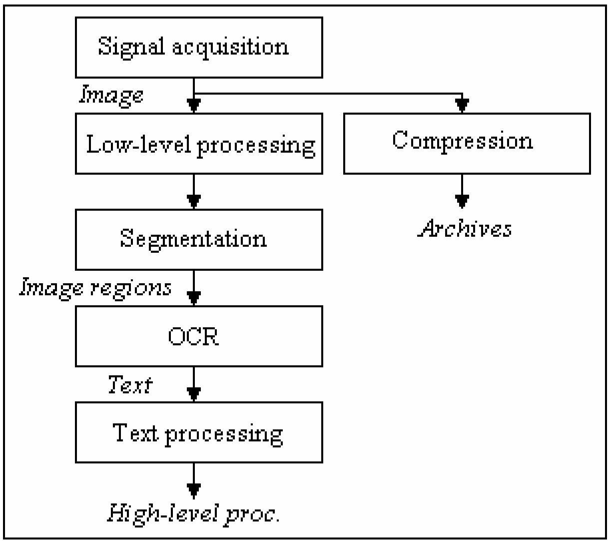


Figure 1.1 Global schema of an automated license plate recognition system.

License plate recognition system is mainly made up of a few distinct modules with specific signal-processing functions (Figure 1.1). A low-level processing module restores signal quality; the algorithms are usually selected according to the kind of sensor used as well as environmental conditions. As a second step, the location of interesting scene regions is attained by a segmentation process. Location results feed the actual vehicle identification module, which is generally supported by optical character recognition (OCR) methods. In some applications requiring archival facility for time-logging purposes, image-compression algorithms can be applied to reduce storage space (Zunino & Rovetta 2000).

1.1 Applications of License Plate Recognition

LPR applications have a wide range of applications, which use the extracted plate number and optional images to create automated solutions for various problems. Some of the basic applications will be introduced below.

1.1.1 Parking

In this application, a car is entering a car park. The license plate is recognized and stored by the system. When the car later exits the license plate will be recognize again. The driver will be charged for the duration of the parking time. Also, the system is used to automatically enter pre-paid members without any other fee (Hofman, 2004).

Some parking areas have security requirements. In these situations, such a system adds a further level of security by granting entrance only to registered vehicles. An example of using of LPR on parking is shown in figure 1.2 (Hofman, 2004).

Figure 1.2 Car inside a parking area (Hofman, 2004).

1.1.2 Access Control

In this application, a gate automatically opens for warranted members in a secured area, thus replacing or assisting the security. The events are logged on a database and could be used to search the history of events (Hofman, 2004).

The system can be used in restricted areas to identify the abuses in any situation. For instance, the historical centers of cities like Rome, Florence, etc. are closed to the public traffic. Nonetheless, many people just ignore this and transit the respective areas. Reinforcing the law is very difficult due to the great number of points of access in such areas (Draghici, 1997). An example of using of LPR on access control is shown in figure 1.3.

Figure 1.3 Gate automatically opens for authorized car (Hofman, 2004).

1.1.3 Toll Payment

A system able to recognize registration plates can be used to identify vehicles which transit through the toll gates. Such a system can be used to achieve two types of goals. Firstly, the system can be used in conjunction with a database containing registration data and owners' information in order to debit the amount due directly into the car owner's account. An example of using of LPR on toll payment is shown in figure 1.4.

Secondly, such a system can be used as a back-up system which deals only with fraudulent vehicles. In some countries, a remote sensing system is used to identify certain vehicles which are fitted with a special device. Those vehicles are allowed to transit without stopping through certain dedicated channels at the toll gates, thus eliminating queuing. However, fraudulent users can transit those dedicated channels without having the device fitted to their cars thus trying to avoid paying the toll. In such cases, the system can be triggered. The system would automatically identify the car and, in conjunction with a database, can identify the owner of the car and even issue a fine (Draghici, 1997; Hofman, 2004).

Figure 1.4 Car at the toll booths (Hofman, 2004).

1.1.4 Border Control

In this application, the car number is registered in the entry or exits to the country, and used to monitor the border crossings. It can short the border crossing turnaround time and cut short the typical long lines (Hofman, 2004).

1.1.5 Enforcement

In this application, a list of stolen cars or unpaid fines is used to alert on a passing cars. The LPR system is deployed on the roadside, and performs a real-time match between the passing cars and the list. When a match is found a siren or display is activated the police officer is notified with the detected car and the reasons for stopping the car (Hofman, 2004). An example of using of LPR on enforcement is shown in figure 1.5.

Figure 1.5 Speeding car (Hofman, 2004).

1.1.6 Travel

A number of LPR units are installed in different locations in the city and the passing vehicle plate numbers are matched between the points. The average speed and travel time between these points can be calculated and presented in order to monitor traffic loads. Additionally, the average speed may be used to issue a speeding ticket (Hofman, 2004).

1.2 Outline of the Thesis

The thesis presented here aims at the following aspects. Study on the existing LPR systems. Implement existing techniques for a license plate segmentation system. License plate segmentation with smearing algorithm, wavelet transforms and MVE (Maximizing Vertical Edges) methods have been implemented in this study.

1.3 Thesis Organization

This thesis presents alternative system for the segmentation of Turkish license plates. The thesis is structured as follows:

- **Chapter Two:** A review of the previous contributions to this work. A brief literature review of the techniques used in the license plate segmentation.
- **Chapter Three:** A description of the techniques which are used in this thesis for license plate segmentation.
- **Chapter Four:** Experimental analysis and the results of the techniques used in this thesis are discussed.
- **Chapter Five:** Conclusions about our work and future work are presented.

CHAPTER TWO
LITERATURE REVIEW

In the license plate segmentation, image is searched for the region that contains the license plate. The techniques for the solutions incorporate a number of features of the license plate like for example: "a license plate is a rectangular area on the car, which contains a number of characters".

There are many techniques which are used for image segmentation. Followings are the examples of image segmentation techniques which are used in previous LPR systems.

2.1 LP Extraction using Grayness and Texture

Two features, grayness and texture, are appointed to each pixel in the image. In fuzzy membership values between 0 and 1 are used for both features. In this technique DTCNNs (Discrete Time Cellular Neural Network) are used, which forces us to represent grayness and texture by crisp values -1 and 1. A grayness of 1 indicates that the pixel has a gray-value that corresponds to the color of a license plate. A grayness of -1 indicates that the color of the pixel is outside the range of gray-values that is common for license plates.

The ranges for grayness and texture are determined by a histogram based method. For the grayness feature, the gray-values of pixels taken from a large number of exemplary license plates are used to construct a frequency table (the number of occurrences of each gray-value in each license plate). Based on this table, the value range is derived. For the texture feature the histogram is constructed by applying a 3×3 Sobel operator to each license plate pixel.

The range for grayness has an upper bound and a lower bound. This requires a two step evaluation. In the first step all pixels of the input image (Figure 2.1a) with a gray-value larger than lower bound are extracted (Figure 2.1b). In the second step all pixels with a gray-value larger than upper bound are deleted from the output (Figure 2.1c). Morphologically, such an operation is described by dilation. A 7x7 block structuring element is chosen to dilate the grayness image (Figure 2.1d). The choice of the structuring element is based on the assumption that each character consists of a set of narrow lines that have width of less than 8 pixels.

All pixels with a Sobel-value larger than the license plate pixels' absolute Sobel-value (dark/light transitions) are extracted. In the second step all pixels with a Sobel-value smaller than the license plate pixels' negative Sobel-value (light/dark transitions) are added (Figure 2.1e). Similar to the initial grayness image, the texture image also needs to be dilated (Figure 2.1f). The reason for not mapping the last dilation is that it can be combined with the following operation. Remember that a part of the image is very likely to be license plate if it has the appropriate grayness and the appropriate texture. The combined feature is obtained by computing the intersection of the grayness and the texture image (Figure 2.1g). Notice that figure 2.1g still contains a number of objects that are not license plates. Apparently, objects like trees and parts of the crash barrier have the same characteristics as license plates.

The four weak size constraints that can be used to reduce the set of potential license plates are minimum/maximum height and minimum/maximum width. The rejection of smaller plates at this stage turns out to have no effect on the overall performance of the system, since these plates are unreadable anyway due to the limited character resolution. The size constraints can be computed in parallel by four DTCNNs. After applying these constraints to the image figure 2.1h is obtained. (Ter Brugge, Stevens, Nijhuis, & Spaanenburg, 1998)

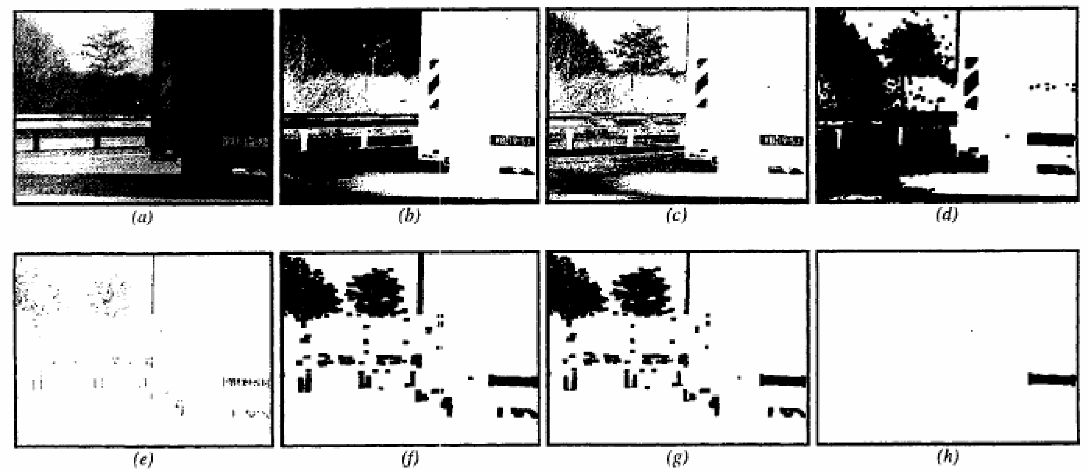


Figure 2.1 Extraction of potential license plates: (a) is the input image, (b) indicates the pixels that are bright enough, (c) is the grayness feature, (d) is the dilated grayness feature, (e) is the texture feature, (f) is the dilated texture feature, (g) is the combination of texture and grayness, (h) is the image that contains all plates after deleting undersized and oversized plates.

2.2 LP Extraction using Line Projection

In order to locate the number plate area, a search for character like shapes is conducted. When three or more such shapes are found in similar horizontal positions, the system looks in the neighborhood for other similar shapes. At a second stage, if the shapes are identified as number plate characters, the system assumes a number plate region has been found.

In order to improve the performance of the location process, this technique has been developed. This technique is based on the fact that the lines where the number plate is located in the image have a clear "signature" which makes it usually possible to distinguish them from other lines in the image, or at least to reselect some positions where to look further. Figure 2.2 shows two such lines. The top image shows the positions of the cross sections (the white lines). The "signature" of the number plate can be observed in the bottom cross section. It corresponds to strong grey level variations at somehow "regular" intervals.

The analysis of the image lines in order to identify which lines "cut" the number plate can be conducted both in the spatial and in the Fourier domain. In the Fourier domain the analysis proves to be very difficult and the work is continued using spatial information. An algorithm which analyses the maximum and minimum of the cross section must be developed. This algorithm searches for a set of continuous maximums and minimums with some predefined characteristics. These characteristics are dynamically chosen from a set of predefined values, using statistical information.

Once a horizontal line that crosses the number plate has been located, this information is used to define an area which should contain the number plate image. (Barosso, Rafael, Dagless, & Cruz, 1997; Parisi, Claudio, Lucarelli, & Orlandi, 1998)

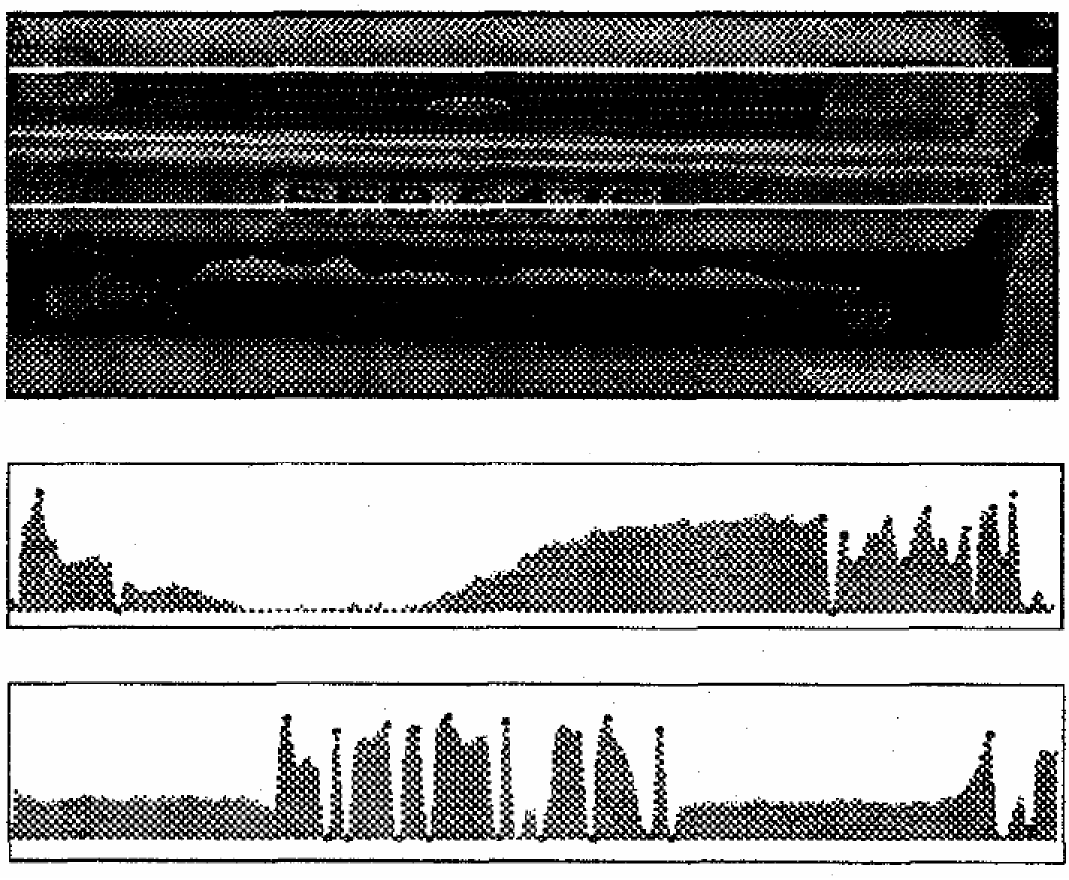


Figure 2.2 Cross section of a car image (the "signature" is shown in the bottom image).

2.3 LP Extraction using Vector Quantization

Most location methods disregard the actual contents of image regions in the search process and rather consider aspect features of candidate portions (edges, contrast, etc.). In some situations, however, peculiarities of acquired signals might mislead the search; for example, particular colors or contrast settings in critical image areas might result in poorly segmented regions and erroneous location. The drawback of such approaches is that the location module often ignores whether the regions considered contain plate-pertinent information; therefore, the crucial textual analysis is remitted to succeeding steps of the image-understanding process.

The quantization principle helps overcome such limitations just because the coding process involves an implicit analysis of image contents. As a codebook is defined in the same (pixel) space as the encoded blocks, associating each block with its best-matching codeword implies a classification of the block content. The classification result may give some hint about the block content itself; in particular, it may establish whether the block is likely to cover a license plate. The overall location approach can be formalized as follows: let the image-content function be defined as t (**p**) = 1 if pixel $p(x, y)$ belongs to a license plate, and 0 otherwise. Then, the general location problem is equivalent to locating the image region S^* such that: $S^* = \max_S \oint_S t(p)dp$. A license plate is realistically framed by a rectangular box; this assumption will be used throughout this section, and the associated rectangular regions will be denoted by the term “stripes”. Thus location is equivalent to finding:

$$S^* = \max_{x_0, y_0, x_1, y_1} \int_{x_0}^{x_1} \int_{y_0}^{y_1} t(x, y) dx\, dy. \tag{2.1}$$

The block-coding method prevents the location process from resolving the pixel level; hence a stripe's boundaries lie along the grid of blocks. The image-partitioning schema and the VQ-based approach impose a reformulation of the overall location problem for two basic reasons: 1) the integral in equation 2.1 becomes a sum of contributions from blocks rather than from single pixels and 2) after VQ block classification, the limited number of code words gives rise to an ambiguity problem:

the same codeword may encode both “interesting” and “insignificant” blocks. Thus, the contribution from each block must express the average probability that the block's pixels convey plate information. The VQ-based location problem in equation 2.1 can be restated as

$$S^* = \max_S \sum_{b \in S} p(t = 1|b) \tag{2.2}$$

where indexes the code words associated with the blocks in region S^*, and $p(t = 1|b)$ denotes the average probability that a block's pixels contain plate information, given that the block is coded by the codeword **b**. Each codeword has a “score” which is associated that estimates the corresponding probability of conveying plate information and this is the important fact related to VQ-based location. The probability of a region will result from the contributions of all the code words involved in the coding of the region itself. Location results directly from sorting out the highest-score region in the image (Zunino and Rovetta, 2000; Hermida, Rodrìguez, Lijò, Sande, & Iglesias 1997).

2.4 LP Extraction using Gabor Filters

Gabor filter, which is tuned for License Plate, is applied to image. Gabor filter response will be high on the regions that contain the license plate on the image (Kahraman and Gökmen 2003). Figure 2.3a shows the original image and figure 2.3b shows Gabor response of the original image. Next, Otsu thresholding is applied to the Gabor filter result. This step is needed to binarize and clarify the Gabor result. After that, morphological operations with an appropriate rectangle morphological operator are applied to the Otsu result. First dilation is applied to the Otsu result. This makes the license plate candidate region/blob to become wider, so erosion is applied afterwards with the same operator. This operation is called closing. To prevent unwanted regions to join together we first apply erosion, after this operation the regions will be narrowed. Thus, dilation is applied afterwards. This operation is called opening. The resulting image is a binary image that consists of some regions scattered around the image. Connected component analysis is applied to the image to gather information about these regions (Figure 2.3c). (Kahraman, Kurt & Gökmen, 2003)

The knowledge about license plate properties like: license plates width height ratio interval, the license plates and the original image's dimensions ratio are used. By checking these, the regions that cannot be license plate are eliminated (Figure 2.3d).

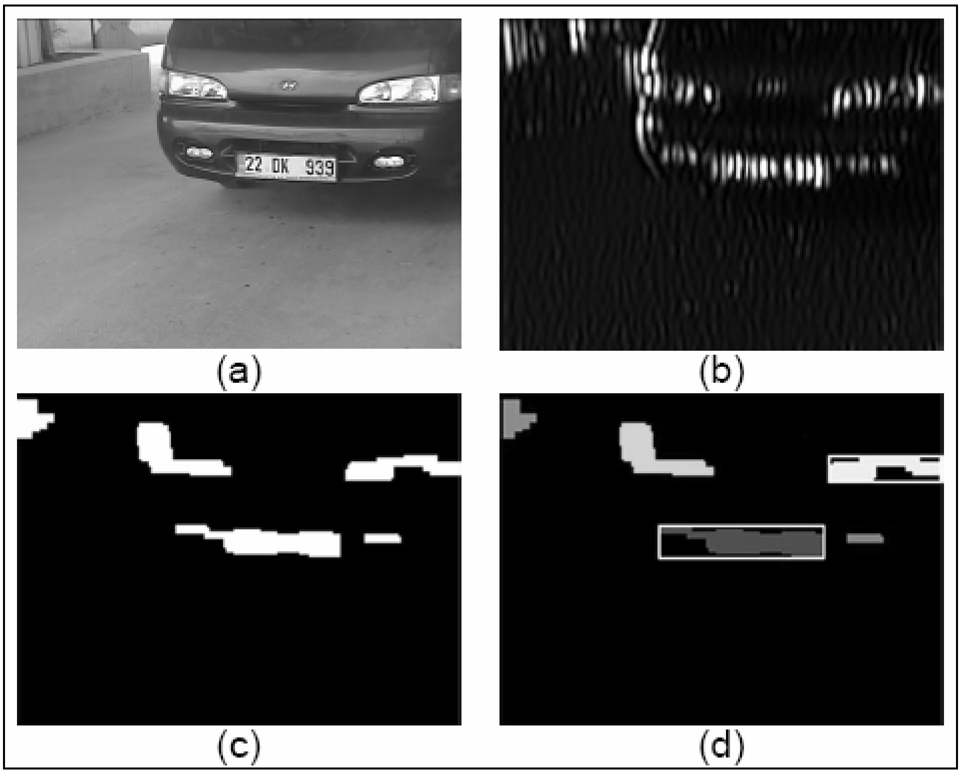


Figure 2.3 Extraction of potential license plates: (a) Input image, (b) Gabor response, (c) Otsu thresholding and morphological operations applied to gabor response, (d) License Plate Canditates.

CHAPTER THREE
IMAGE PROCESSING TECHNIQUES USED IN THE THESIS

Although, there are many LP (License Plate) segmentation techniques, only three of them have been implemented in this study. This chapter explains these techniques in detail.

3.1 LP Segmentation with Smearing Algorithm

OCR (Optical Character Recognition) is the mechanical or electronic translation of images of handwritten, typewritten or printed text (usually captured by a scanner) into machine-editable text.

Steps of OCR processing are character segmentation and character recognition. License plates can have seven or eight characters (numbers / letters). So we can use character segmentation techniques of OCR to find these characters.

Algorithm depends on 4 steps which are shown in figure 3.1; at first we apply histogram equalization to an image for increasing contrast of images. At a second step, threshold is applied to image; result of that step is a binary image. At the third step, smearing algorithm is applied to an image for character segmentation. At the last step, result image is searched for license plate.

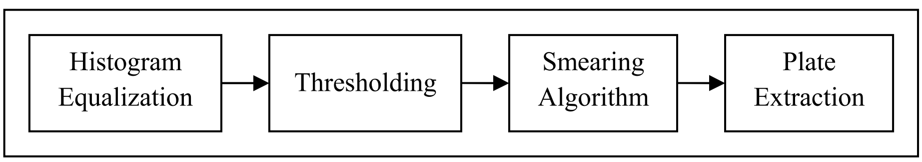


Figure 3.1 Flowchart of the algorithm.

3.1.1 Histogram Equalization

Histogram equalization is a method in image processing of contrast adjustment using the image's histogram. This method usually increases the local contrast of many images, especially when pixel values of the image are represented by close contrast values. Through this adjustment, the intensities can be better distributed on the histogram. This usually allows for areas of lower local contrast to gain a higher contrast. Histogram equalization accomplishes this by effectively spreading out the most frequent intensity values.

The method is useful in images with backgrounds and foregrounds that are both bright or both dark. A key advantage of the method is that it is a fairly straightforward technique. A disadvantage of the method is that it is indiscriminate. It may increase the contrast of background noise, while decreasing the usable signal.

Consider a discrete grayscale image, and let n_i be the number of occurrences of gray level i. The probability of an occurrence of a pixel of level i in the image is

$$p_r(r_i) = \frac{n_i}{n} \qquad i \in 0, \dots, L-1$$

n is the total number of pixels in the image, n_i is the number of pixels that have gray level r_i, L being the total number of gray levels in the image, n being the total number of pixels in the image, and p_r being in fact the image's histogram, normalized to [0,1].

Let us also define c as the cumulative distribution function corresponding to p_r, defined by:

$$c(i) = \sum_{j=0}^{i} p_r(r_i)$$
$$c(i) = \sum_{j=0}^{i} \frac{n_j}{n} \qquad i \in 0, \dots, L-1 \tag{3.1}$$

also known as the images accumulated normalized histogram. As indicated earlier, a plot of $p_r(r_i)$ versus r_i is called a histogram. The transformation (mapping) given in equation. (3.1) is called histogram equalization or histogram linearization. Original image, result of histogram equalization and histogram of these images are shown in figure 3.2. (Gonzalez & Woods, 2002)

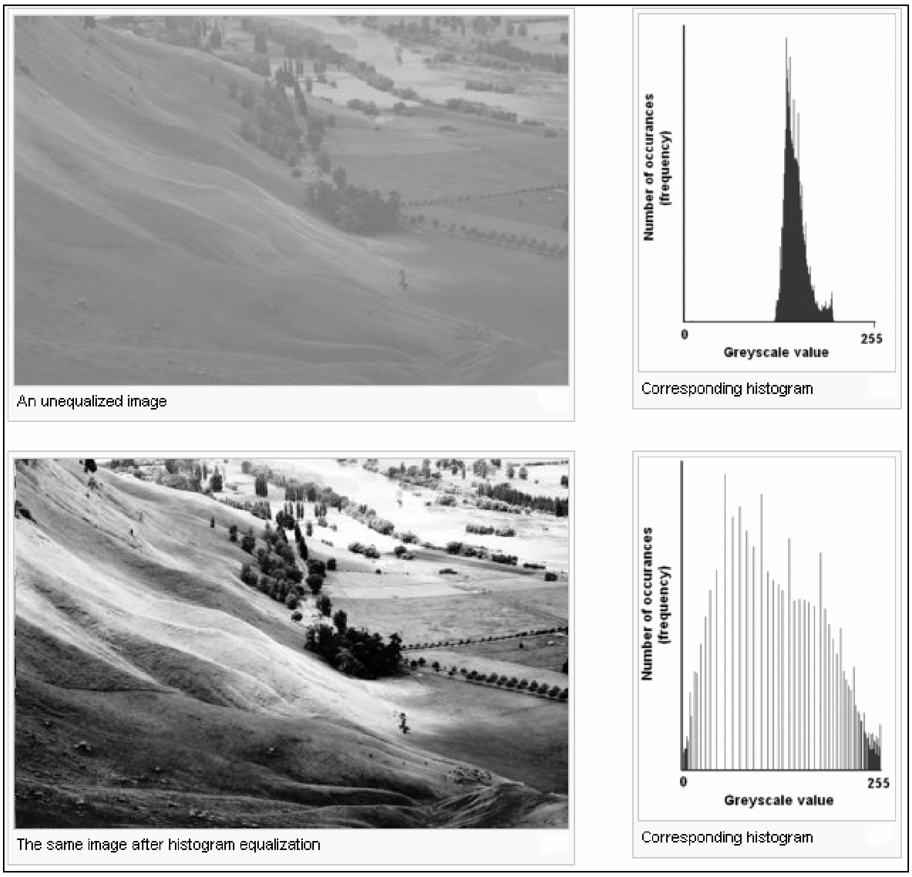


Figure 3.2 Image, results of histogram equalization and corresponding histograms.

3.1.2 Thresholding

Thresholding transforms a gray scale image into binary image which has only two possible values of each pixel. Thresholding is the simplest method using for image segmentation. Thresholding creates binary images from grey-level ones by turning all pixels below some threshold to zero and all pixels above that threshold to one. If *g(x, y)* is a thresholded version of *f(x, y)* at some threshold *T*,

$$g(x,y)=\begin{cases}1 & f(x,y)\geq T\\ 0 & f(x,y)<T\end{cases}.$$

The key parameter in thresholding is obviously the choice of the threshold. Several different methods for choosing a threshold exist. When *T* is constant, this approach is called global thresholding. (Gonzalez, Woods & Eddins, 2004)

3.1.2.1 Global Thresholding

The simplest off all thresholding techniques is to partition the image histogram by using a single global threshold, T. The following algorithm can be used to obtain T automatically:

1. Select an initial estimate for T
2. Separate the pixels into two clusters according to the T. This will produce two groups of pixels: G_1 consisting of all pixels with gray level values $>T$ and G_2 consisting of all pixels with gray level values $\leq T$.
3. Compute the average gray level values μ_1 and μ_2 for the pixels in regions G_1 and G_2.
4. Compute a new threshold value:

$$T = \frac{1}{2}(\mu_1 + \mu_2)$$

5. Repeat steps 2 through 4 until the difference in T in successive iterations is smaller than a predefined parameter T_0. (Gonzalez, Woods & Eddins, 2004)

3.1.2.2 Adaptive Thresholding

The original image is divided into sub images and then a different threshold is used to segment each sub image. (Gonzalez & Woods, 2002) But, every sub images may not contain a boundary between object and background, so variance of a sub image is calculated to decide whether, or not, a sub image contains a boundary between object and background. The second moment, $\mu_2(z)$, is the variance. The expression for the 2nd moment about the mean is given by

$$\mu_2 = \sum_{i=0}^{L-1} (z_i - m)^2 p(z_i)$$

Where z_i is a random variable indicating intensity, $p(z)$ is the histogram of the intensity levels in a region, L is the number of possible intensity levels, and

$$m = \sum_{i=0}^{L-1} z_i p(z_i)$$

is the mean intensity. (Gonzalez, Woods & Eddins, 2004)

3.1.3 Smearing Algorithm

The Smearing Algorithm (SA) is a useful algorithm for detecting regions of text on a background. Smearing algorithm works on binary images where white pixels are represented by 0's and black pixels by 1's. The algorithm transforms a binary sequence *x* into *y* according to the following rules:

- 0's in *x* are changed to 1's in *y* if the number of adjacent 0's is less than or equal to a predefined threshold *C*.
- 1's in x are unchanged in y.

These steps have the effect of linking together neighboring black areas that are separated by less than *C* pixels. The SA is applied row-wise to the document using a threshold C_h, and column-wise using threshold C_v, yielding two distinct bitmaps. These two bitmaps are combined in a logical OR operation. An example of smearing algorithm shows in figure 3.3. (Shafait, Keysers & Breuel)

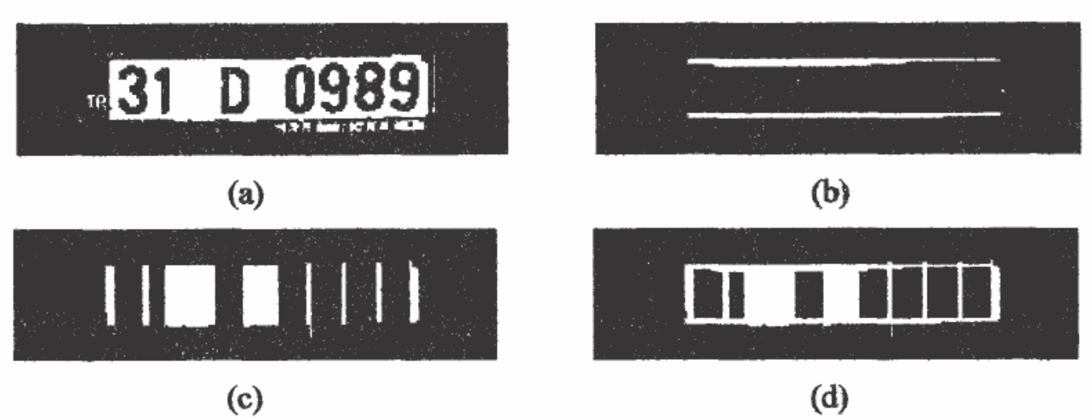


Figure 3.3 Smearing algorithm applied to binary image. (a) Binary image; (b) Horizontal smearing; (c) Vertical smearing; (d) Vertical and horizontal smearing binary images are combined in a logical OR operation.

3.1.4 Plate Extraction

After applying “Smearing Algorithm” many possible license plate positions come out. The exact plate position must be found among the possible plate positions in the next step.

There are two types of license plates used in Turkey. They are single line and double line license plates. License plates have seven or eight character on it. Single line license plates’ height is 110 millimeters and length is 520 millimeters. Double line license plates’ height is 210 millimeters and length is 320 millimeters.

Therefore, all possible license plate’s dimensions and their ratios of height to length can be used to decide whether the tested component is plate or not.

3.2 LP Segmentation with Wavelet Transform

License plates consist of many edges because of their construction material and their painted characters on the background. The following procedure eliminates the image sub regions which do not have so many edges by using Wavelet Transform.

The number of sub regions to be searched for license plate is therefore decreased using this Wavelet Transform based edge detection technique.

3.2.1 Wavelet Transform

Wavelet analysis is a set of mathematical functions for analyzing any type of signals by sorting data by frequency, and then study each component with a resolution matched to its scale. Small features and large features are observable since they are studied separately. Wavelet transform has advantages over traditional Fourier methods in analyzing physical situations where the signal contains discontinuities and sharp spikes.

Think about an image *f(x, y)* of size $M \times N$ whose forward, discrete transform, *T (u, v)*;

$$T(u,v)=\sum_{x,y} f(x,y)g_{u,v}(x,y)$$

x and *y* are spatial variables and *u, v* are transform domain variables. Generalized inverse transform of *T (u, v)*;

$$f(x,y)=\sum_{u,v} T(u,v)h_{u,v}(x,y)$$

The $g_{u,v}$ and $h_{u,v}$ are called forward and inverse transformations kernels. In this case

$$h_{u,v}(x,y)=g^{*}_{u,v}(x,y)=\frac{1}{\sqrt{MN}}e^{j2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)}$$

$$u=0,1,\ldots,M-1 \quad v=0,1,\ldots,N-1 \quad j=\sqrt{-1}$$

Transform domain variables *v* and *u* represents horizontal and vertical frequency. The kernels are separable

$$h_{u,v}(x,y)=h_u(x)h_v(y)$$

$$h_u(x)=\frac{1}{\sqrt{M}}e^{\frac{j2\pi ux}{M}} \quad and \quad h_v(y)=\frac{1}{\sqrt{N}}e^{\frac{j2\pi vy}{N}}$$

The kernels can be represented as three separable 2-D wavelets

$$\psi^{H}(x,y)=\psi(x)\varphi(y) \qquad horizontal\ wavelets$$

$$\psi^{V}(x,y)=\varphi(x)\psi(y) \qquad vertical\ wavelets$$

$$\psi^{D}(x,y)=\psi(x)\psi(y) \qquad diagonal\ wavelets$$

And one separable 2-D scaling function

$$\varphi(x,y)=\varphi(x)\varphi(y)$$

Each of these 2-D functions is the product of two 1-D functions

$$\varphi_{j,k}(x)=2^{j/2}\varphi\left(2^{j}x-k\right)$$

$$\psi_{j,k}(x)=2^{j/2}\psi\left(2^{j}x-k\right)$$

k is the position of 1-D functions along *x*-axis, scale *j* determines their width and $2^{j/2}$ controls their amplitude. Both $\varphi(x)$ and $\psi(x)$ can be expressed as

$$\varphi(x)=\sum_{n} h_{\varphi}(n)\sqrt{2}\varphi(2x-n)$$

$$\psi(x)=\sum_{n} h_{\psi}(n)\sqrt{2}\varphi(2x-n)$$

h_φ and h_ψ are called scaling and wavelet vectors.(Bow, S.T., 2002; Gonzalez, Woods & Eddins, 2002; Gonzalez & Woods, 2002)

3.2.2 Morphology

The word morphology is the context of mathematical morphology as a tool for extracting image components that are useful in the representation of region shape. Morphology based method is performed to extract important contrast features as guides to search desired license plates. It has some operations as dilate (grow image regions), erose (shrink image regions), opening and closing (removal and filling of image boundary pixels), thinning, thickening, etc. These operations are used to extract the contrast features as the important cues to extract license plates.

Dilation and Erosion are two basic operators in the area of mathematical morphology. The basic effect of the dilation on a binary image is to gradually enlarge the boundaries of regions of foreground pixels. An example of dilation shows in figure 3.4. Thus areas of foreground pixels grow in size while holes within those regions become smaller. Dilation is the dual of erosion dilating foreground pixels is equivalent to eroding the background pixels. The basic effect of the erosion on a binary image is to erode away the boundaries of regions of foreground pixels. An example of erosion shows in figure 3.5. Thus areas of foreground pixels shrink in size, and holes within those areas become larger.

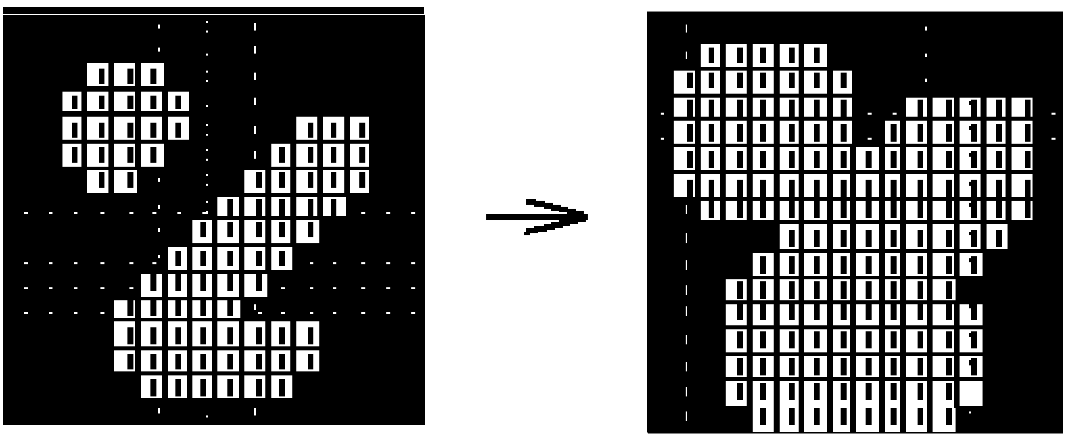


Figure 3.4 Effect of dilation using a 3×3 square structuring element.

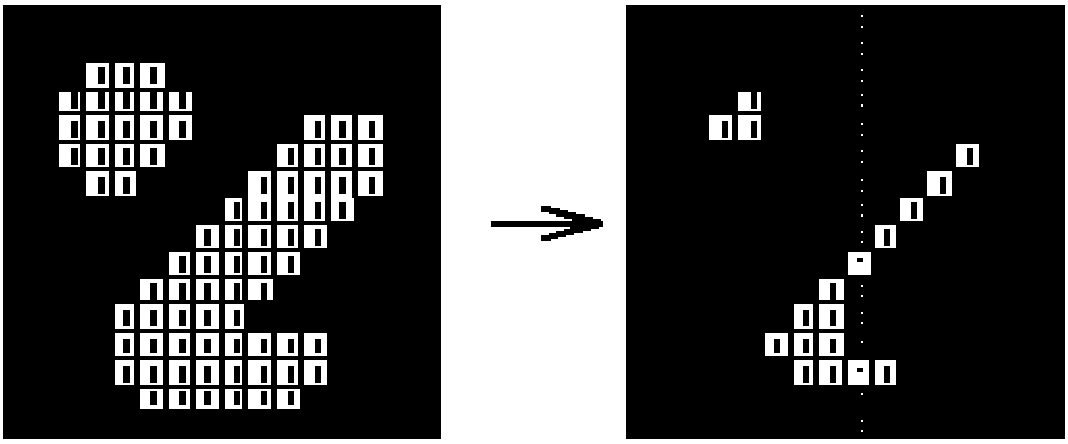


Figure 3.5 Effect of erosion using a 3×3 square structuring element.

3.2.3 Procedure

The first part of this algorithm is three level Wavelet Transform. Starting from the highest level (lowest frequencies);

(1) High frequency coefficients are combined in a single edge map. Absolute values of horizontal high frequency coefficients, vertical high frequency coefficients, and diagonal high frequency coefficients are summed after multiplying by weighting numbers. Vertical high frequency coefficients are multiplied by the highest weighting number because of high incidence of vertical line segments of characters in a license plate.

(2) Thresholding is applied.

(3) Dilation is performed on edge map.

(4) Increase the dimension (by two) of the map which is obtained at the end of third step.

(5) 0 is assigned to coefficients of the wavelet transform which are out of the resultant map for preparation to the next level.

Steps from 1 to 5 are repeated up to the lowest level, where an edge map of the same size as the original image is obtained.

(6) For the last step of this part, erosion and dilation operators are applied to the final edge map in order to eliminate noise and smooth edges.

After step (6), we take the each component in the edge map individually then remove the components which have few pixels. Then we calculate the ratio of row and column sizes of the components. If the component yields a ratio similar to that of license plate then we marke this component as a license plate. Figure 3.6 shows the flowchart of the algorithm. (Duman, Öktem & Çetin 2005)

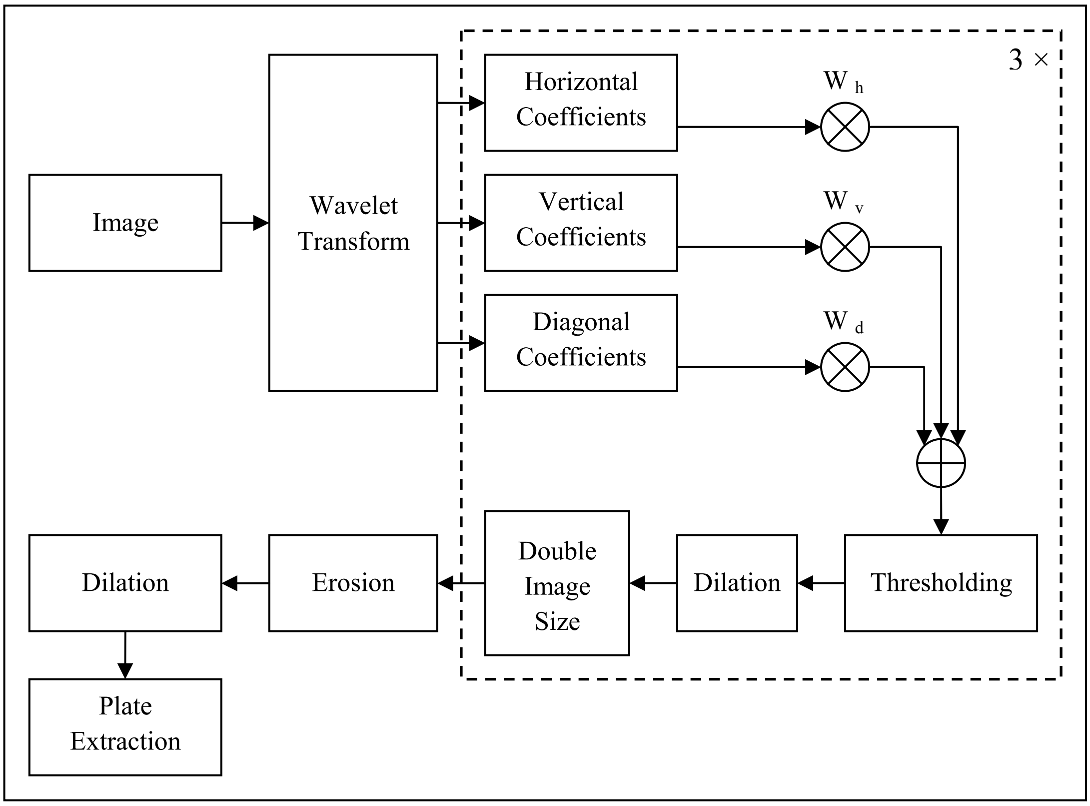


Figure 3.6 Flowchart of the algorithm.

3.3 LP Segmentation with MVE

Cinsdikici (2004) developed MVE (Maximizing Vertical Edges) method, for his Ph.D. thesis. The method is based on simple edge detection. After edge detection, image is scanned for license plate size sub images which have maximum vertical edge pixels. Flowchart of the algorithm is shown in figure 3.7.

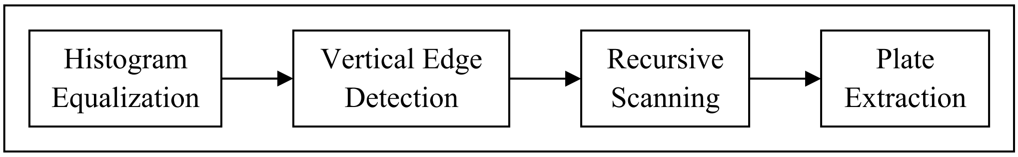


Figure 3.7 Flowchart of the MVE algorithm.

3.3.1 Edge Detection

Edges characterize boundaries therefore their detection is a problem of fundamental importance in image processing. Edges in images are areas with strong intensity contrasts – a jump in intensity from one pixel to the next. Edge detecting an image significantly reduces the amount of data and filters out useless information, while preserving the important structural properties in an image. There are many ways to perform edge detection. The gradient method is one of these techniques which detect the edges by looking for the maximum and minimum in the first derivative of the image (Gonzalez & Woods, 2002). Based on this one-dimensional analysis, the theory can be carried over to two-dimensions as long as there is an accurate approximation to calculate the derivative of a two-dimensional image. The Sobel operator performs a 2-D spatial gradient measurement on an image. Typically it is used to find the approximate absolute gradient magnitude at each point in an input grayscale image. The Sobel edge detector uses a pair of 3x3 convolution masks, one estimating the gradient in the x-direction (columns) and the other estimating the gradient in the y-direction (rows). A convolution mask is usually much smaller than the actual image. As a result, the mask is slid over the image, manipulating a square of pixels at a time. The actual Sobel masks are shown in figure 3.8.

These masks are designed to respond maximally to edges running vertically and horizontally relative to the pixel grid, one kernel for each of the two perpendicular orientations. The masks can be applied separately to the input image, to produce separate measurements of the gradient component in each orientation

$G_x = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)$

and

$G_y = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)$

An approximate magnitude can be calculated using:

$|G| = |G_x| + |G_y|$

-1	0	+1
-2	0	+2
-1	0	+1

+1	+2	+1
0	0	0
-1	-2	-1

Figure 3.8 Sobel masks.

Figure 3.9a shows original image. Figure 3.9b shows horizontal edges of the original image which is shown in figure 3.9a. Vertical edges of the original image are shown in figure 3.9c. Combination of vertical and horizontal edges gives the edges of the image. Edges of the original image are shown in figure 3.9d.

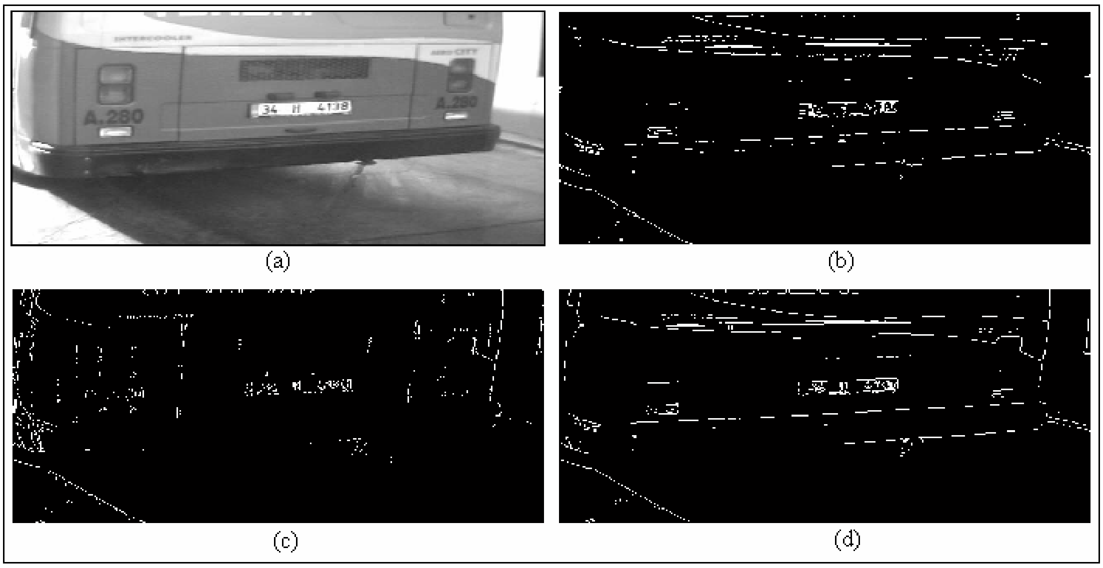


Figure 3.9 Edge detection: (a) Original image; (b) Result of horizontal Sobel mask; (c) Result of vertical Sobel mask;(d) Result of determining both vertical and horizontal edges.

3.3.2 Recursive Scanning

In recursive scanning method, a window having same dimensions with license plate is used. The window is moved pixel by pixel over the image starting from upper left corner of the picture. During this process, numbers of vertical edge pixels within the window are calculated for all possible locations of window. The area which has the most vertical edge pixels is determined and marked as a candidate for license plate and the coordinates of the window is saved.

We run the process eight more times. Each time the vertical edges in the area of candidate license plate are erased for the next process. End of the whole process we have nine possible license plate areas.

CHAPTER FOUR
LICENCE PLATE SEGMENTATION EXPERIMENTAL WORK

MATLAB version 7 (R14) is used in the simulations during this study. “Graphical User Interface Tools”, “Image Processing Toolbox” and sample set which was taken from ASELSAN are use to implement algorithms. Sample set has 1000 images which are 640×312 pixels.

4.1 License Plate Segmentation with Smearing Algorithm

This method is described at subsection 3.1. Second step of this method is thresholding. Two thresholding techniques are used for this method. Results and comparison of two techniques are described in the following subsections.

4.1.1 Global Thresholding

Following figures show the steps of the method. Original image is shown in figure 4.1. Figure 4.2 shows the result of performing histogram equalization on the original image. Global thresholding is performed on image whose histogram is equalized. Result of performing global thresholding is shown in figure 4.3. Figure 4.4 shows the result of smearing algorithm on the thresholded image on. Figure 4.5 shows the successful result of plate extraction.

Figure 4.1 Original image.

Figure 4.2 Result of performing histogram equalization on original image.

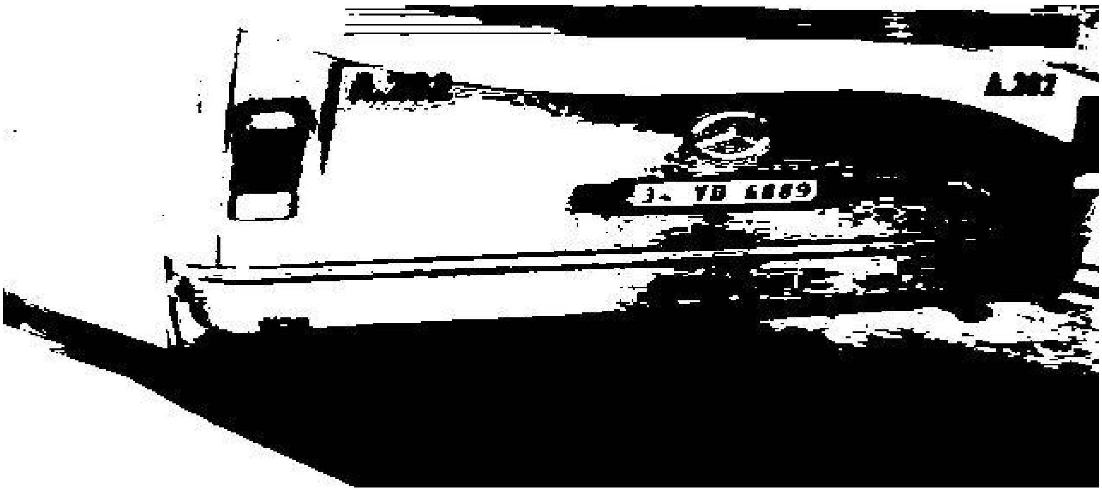


Figure 4.3 Result of performing global thresholding on histogram equalized image.

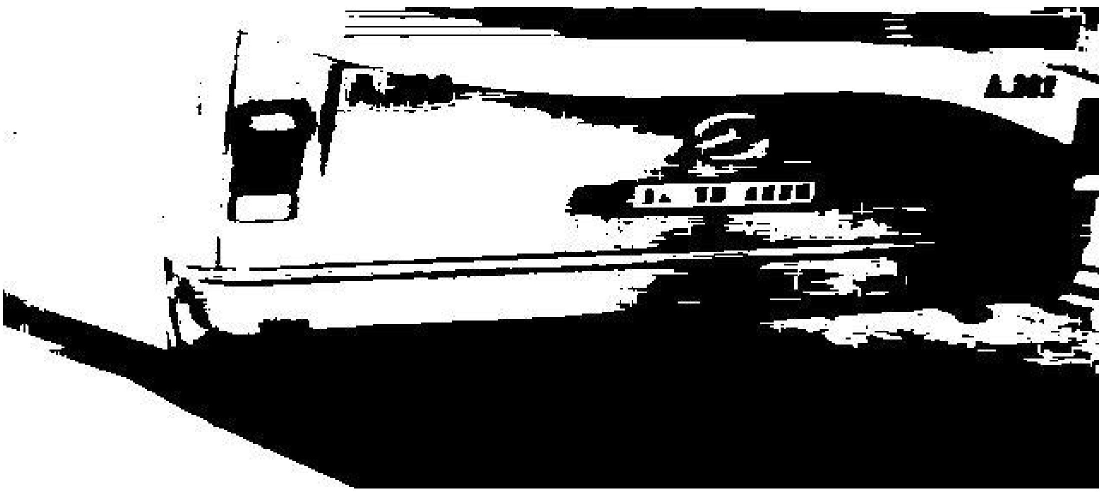


Figure 4.4 Result of performing smearing algorithm on thresholded image.

Figure 4.5 Result of performing plate extraction.

Smearing algorithm is used for segmenting dark objects from light backgrounds. Most of the license plates have dark objects (characters) on light ground. But according to Highway Traffic Law of Turkey license plates have different colors of background numbers and texts. Plates of official and law enforcement cars have light objects (characters) on dark background. Figure 4.6 shows the image of a car which has official car plate. Figure 4.7 shows the result of performing smearing algorithm on an image which belongs to an official car. So this method fails on images which belong to official cars.

Figure 4.6 Image of an official car.

Figure 4.7 Result of smearing algorithm on image of an official car.

Segmentation with histogram equalization and global thresholding is not successful on all images. Too dark and too bright images dose not give successful results with this method. Figure 4.8 shows a dark image which was taken on a night. Figure 4.9 shows the result of performing histogram equalization on this image. Global thresholding is performed on the image shown in figure 4.9. The result of performing global thresholding is shown in figure 4.10.

Figure 4.8 Dark image.

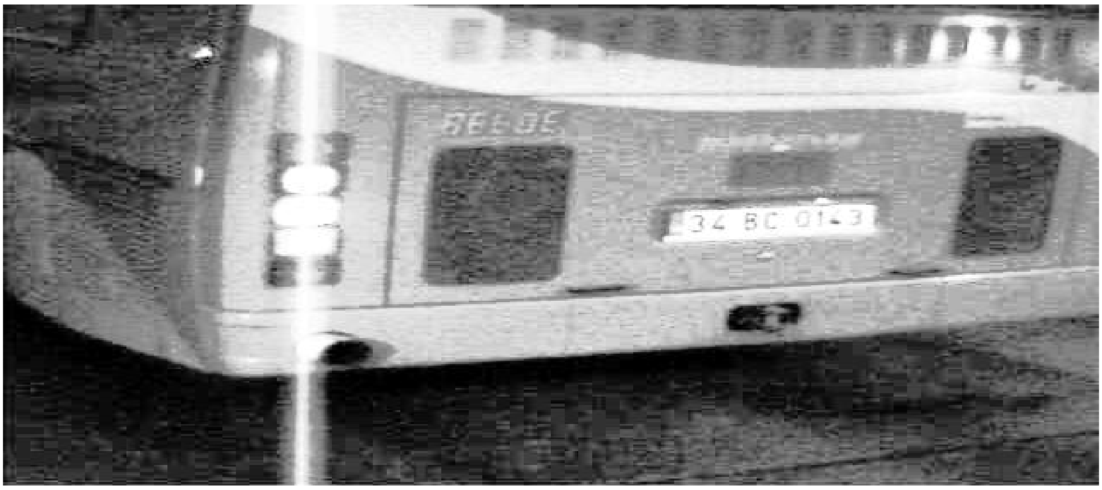


Figure 4.9 Result of performing histogram equalization on dark image.

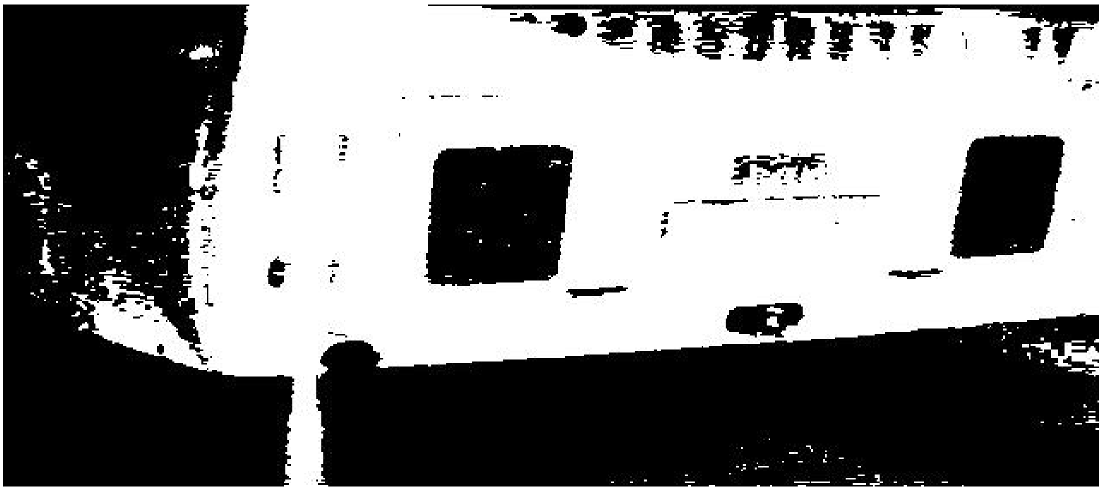


Figure 4.10 Result of performing global thresholding on histogram equalized dark image.

Figure 4.11 Bright image.

Figure 4.11 shows a bright image which was taken on a night. Figure 4.12 shows the result of performing histogram equalization on this image. Global thresholding is performed on the image shown in figure 4.12. Result of performing global thresholding is shown in figure 4.13.

Figure 4.12 Result of performing histogram equalization on bright image.

Figure 4.13 Result of performing global thresholding on histogram equalized bright image.

Table 4.1 Performance of smearing algorithm with global thresholding

Correct license plate location;	%30.2
False or none license plate location;	%61.4
Official license plate (all unsuccessful);	%8.4

General performance results of the method are shown in table 4.1. Because of the failure of the method, adaptive thresholding is used instead of global thresholding in this method.

4.1.2 Adaptive Thresholding

The following figures show the steps of the method. Original image is shown in figure 4.14. Figure 4.15 shows the result of performing histogram equalization on the original image. Basic adaptive thresholding is performed on the image whose histogram is equalized. Result of performing basic adaptive thresholding is shown in figure 4.16. Figure 4.17 shows the result of smearing algorithm on the thresholded image. Figure 4.18 shows the successful result of plate extraction.

Figure 4.14 Original image.

Figure 4.15 Result of performing histogram equalization on original image.

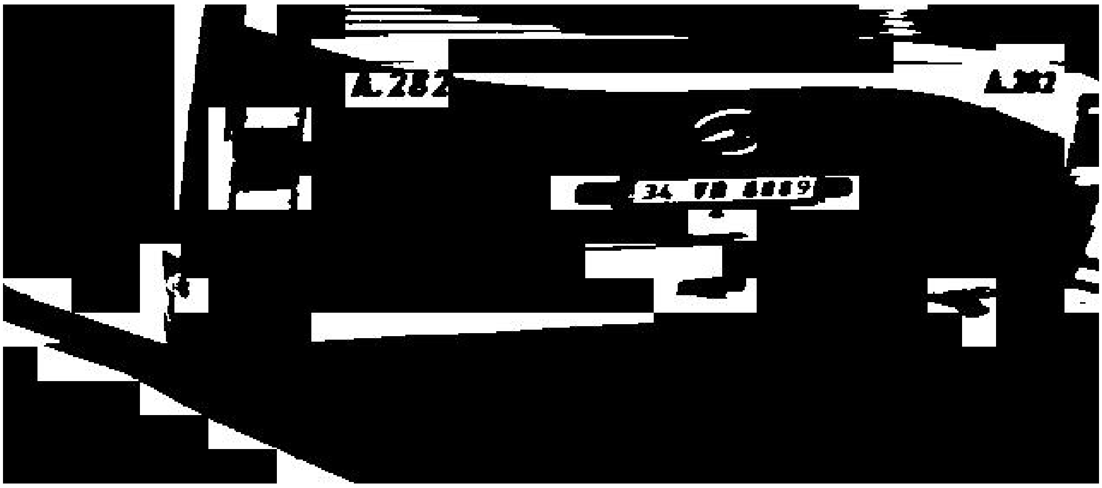


Figure 4.16 Result of performing basic adaptive thresholding on histogram equalized image.

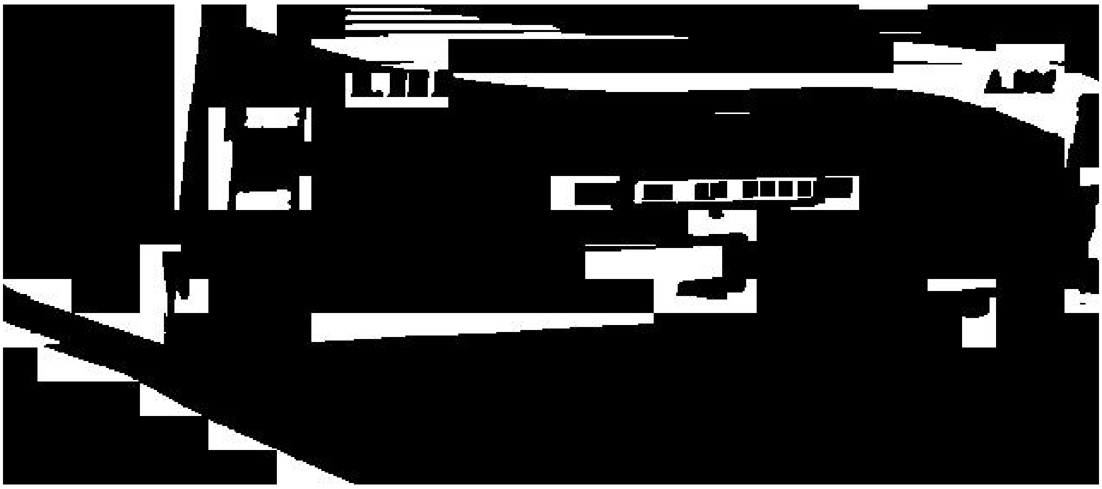


Figure 4.17 Result of performing smearing algorithm on thresholded image.

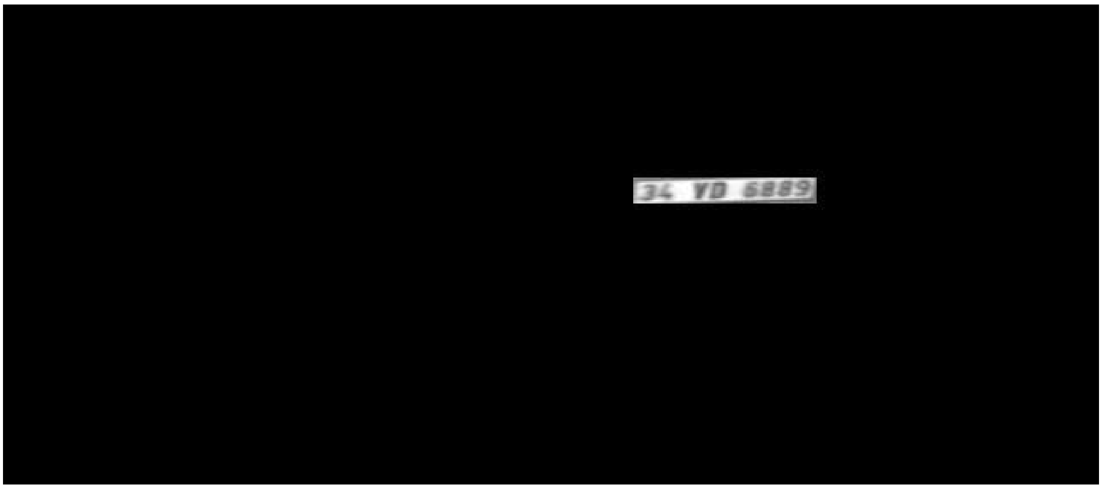


Figure 4.18 Result of performing plate extraction.

Segmentation with basic adaptive thresholding is more successful than global thresolding. Too dark and too bright images cause failures of the algorithm with global thresholding. But basic thresholding is successful on many images. Figure 4.19 shows a dark image which is taken on a night. Figure 4.20 shows the result of performing histogram equalization on this image. Basic adaptive thresholding is performed on the image shown in figure 4.20. The result of performing adaptive thresholding is shown in figure 4.21. Figure 4.22 shows the result of the smearing algorithm on the thresholded image. Figure 4.23 shows the successful result of plate extraction.

Figure 4.19 Dark image.

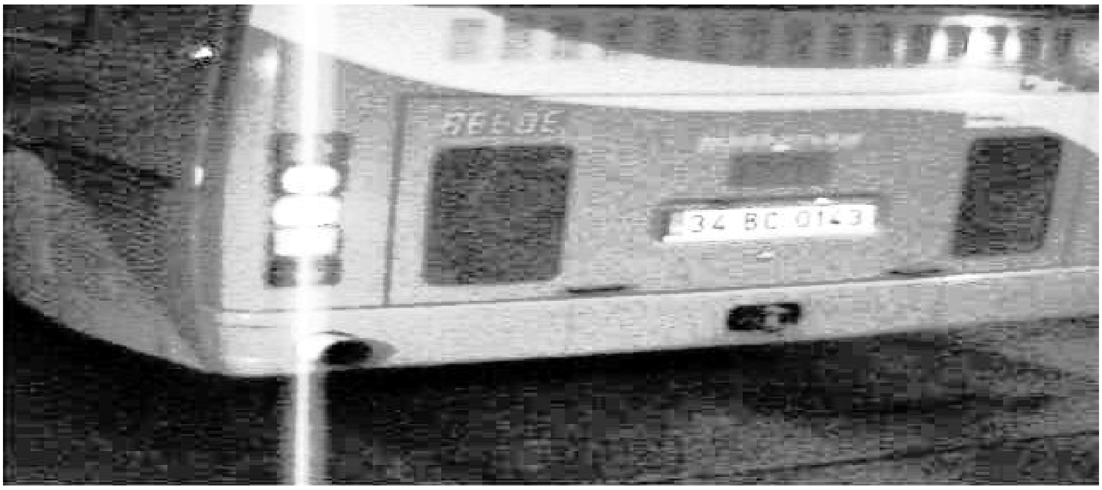


Figure 4.20 Result of performing histogram equalization on dark image.

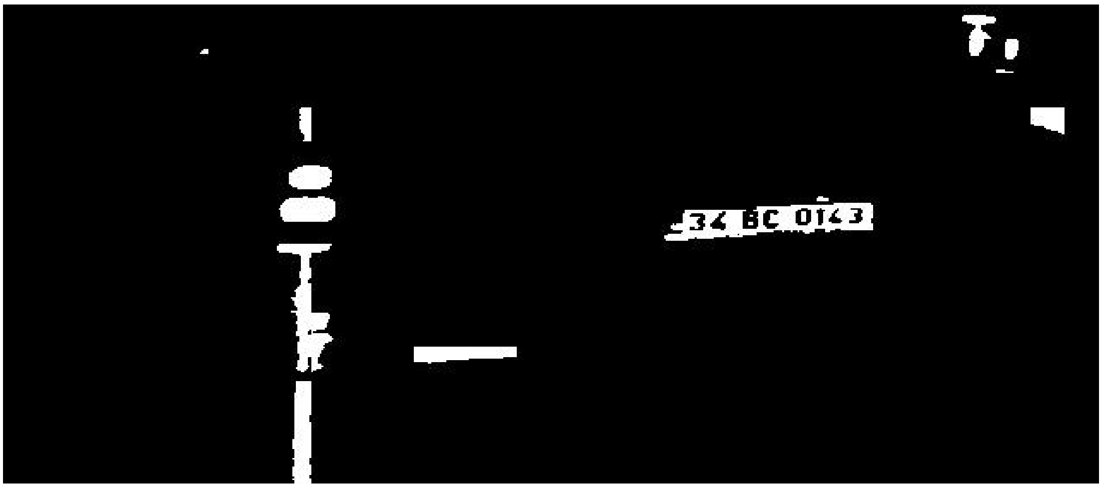


Figure 4.21 Result of performing basic adaptive thresholding on histogram equalized dark image.

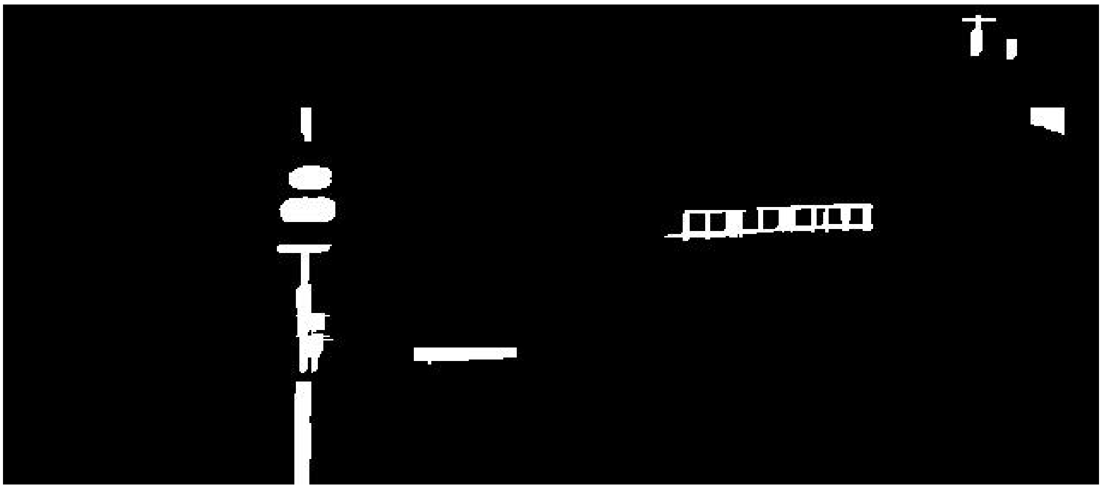


Figure 4.22 Result of performing smearing algorithm on thresholded image.

Figure 4.23 Result of performing plate extraction.

Figure 4.24 shows a bright image which was taken on a night. Figure 4.25 shows the result of performing histogram equalization on this image. Adaptive thresholding is performed on the image shown in figure 4.25. The result of performing adaptive thresholding is shown in figure 4.26.

Figure 4.24 Bright image.

Figure 4.25 Result of performing histogram equalization on bright image.

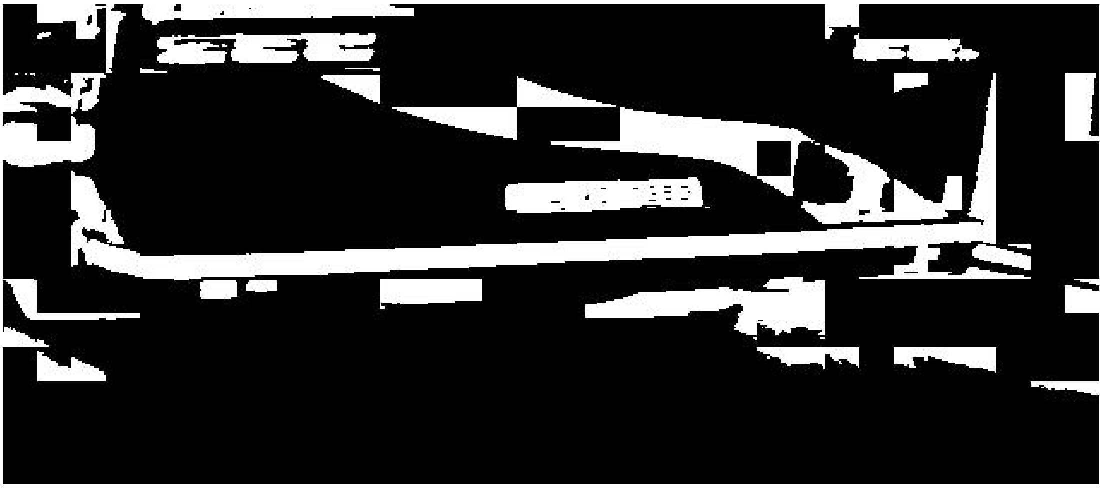


Figure 4.26 Result of performing adaptive thresholding on histogram equalized bright image.

Figure 4.27 shows the result of the smearing algorithm on the thresholded image. Figure 4.28 shows the result of plate extraction.

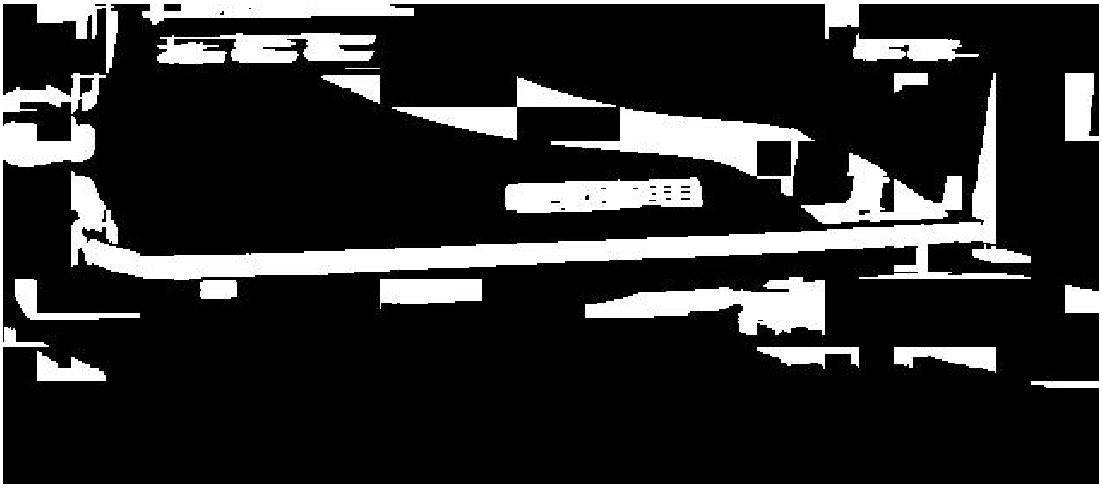


Figure 4.27 Result of performing smearing algorithm on thresholded image.

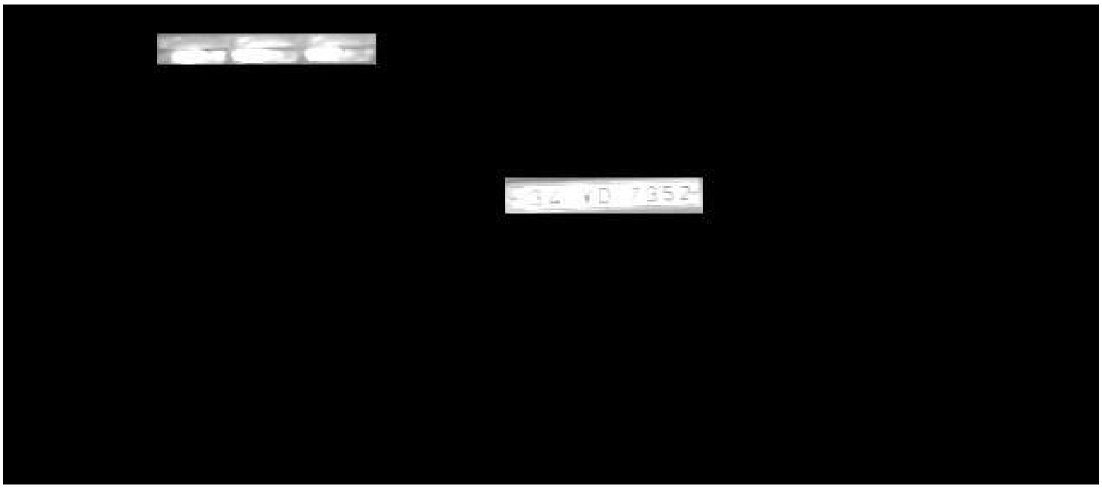


Figure 4.28 Result of performing plate extraction.

General results of the method are shown in table 4.2.

Table 4.2 Perfomance of the smearing algorithm with adaptive thresholding

Correct license plate location;	%63.2
False or none license plate location;	%29.4
Official license plate (all unsuccessful);	%8.4

4.1.3 Comparison of Two Thresholding Techniques

Table 4.3 shows the comparison of the two thresholding techniques. Adaptive thresholding is more successful than global thresholding. The method with global thresholding requires 2.4 seconds for the operation. The method with adaptive thresholding requires 2.6 seconds for the operation.

Table 4.3 Comparison of the two thresholding techniques

	Global Thresholding	Adaptive Thresholding
Correct license plate location;	%30.2	%63.2
False or none license plate location;	%61.4	%29.4
Official license plate (all unsuccessful);	%8.4	%8.4

4.2 License Plate Segmentation with Wavelet Transform

This method is described in subsection 3.2. Two filters, Haar and Daubechies, are used in the wavelet transform for this method. The results and comparison of two filters are described in the following subsections.

4.2.1 Haar Filter

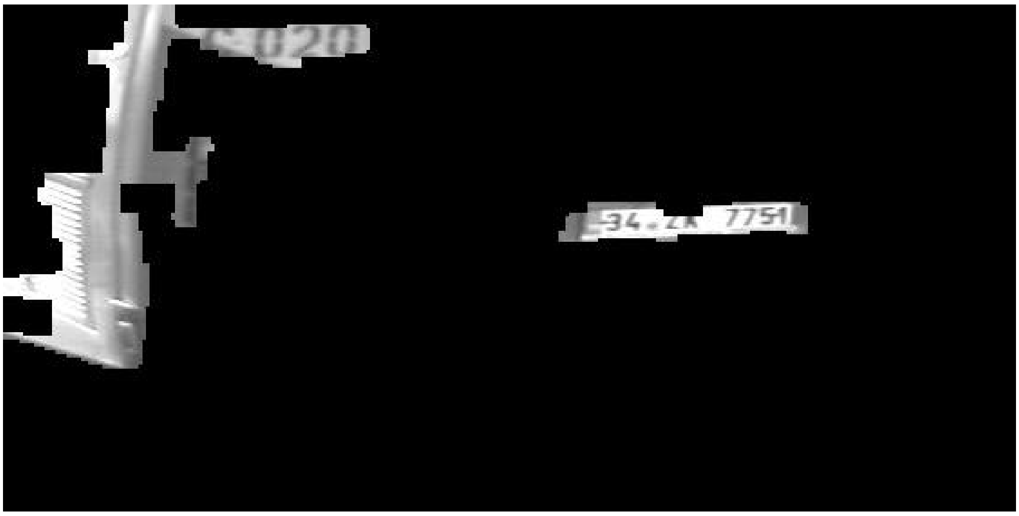
The following figures show the main steps of the method. The original image is shown in figure 4.29a. Figure 4.29b shows the result of performing the wavelet transform with Haar filter on the original image. Figure 4.29c shows the result of plate extraction.

The results of the method with Haar filter are shown in table 4.4. The method sometimes finds more than one candidate including the correct license plate location. Table 4.5 shows the number of candidates for successful results.

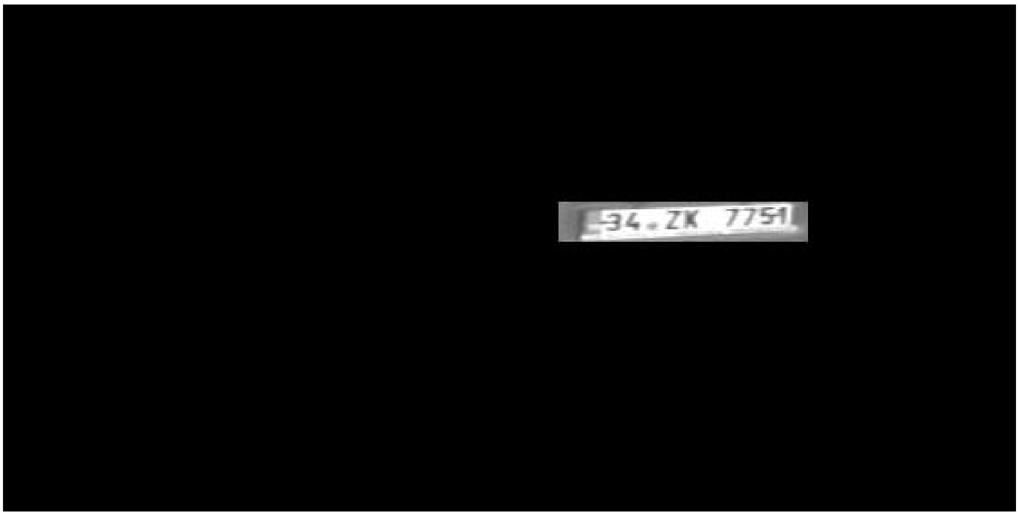
Table 4.4 Performance of the algorithm with Haar filter

Correct license plate location;	%90.8
False or none license plate location;	%9.2

(a)



(b)



(c)

Figure 4.29 LPS with wavelet transform: (a) Original image; (b) Result of wavelet transform by Haar filter; (c) Result of performing plate extraction.

Table 4.5 The number of candidates for successful results of the algorithm with Haar filter

1 Candidate	%75.3
2 Candidates	%11.8
3 or more Candidates	%3.7

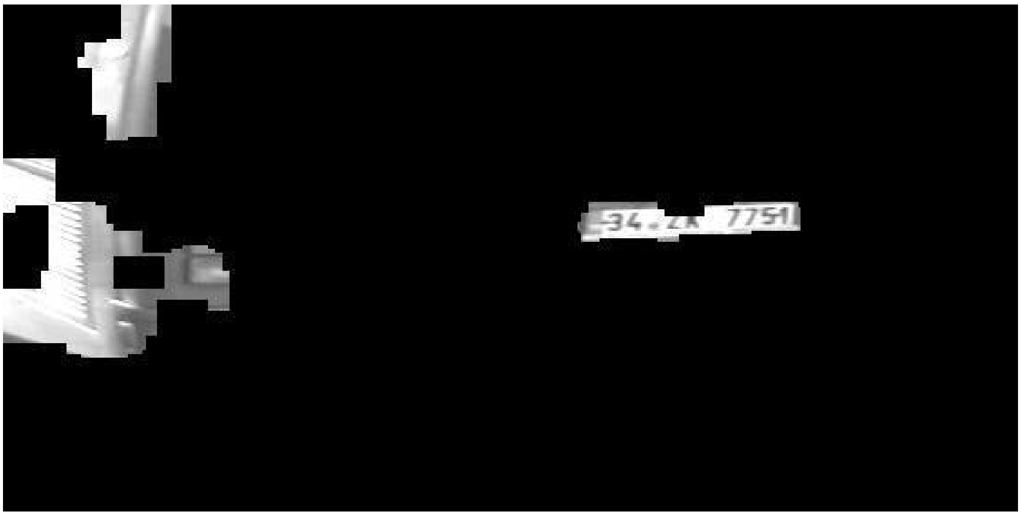
4.2.2 Daubechies Filter

The following figures show the main steps of the method. The original image is shown in figure 4.30a. Figure 4.30b shows the result of performing the wavelet transform with Daubechies filter on the original image. Figure 4.30c shows the result of plate extraction.

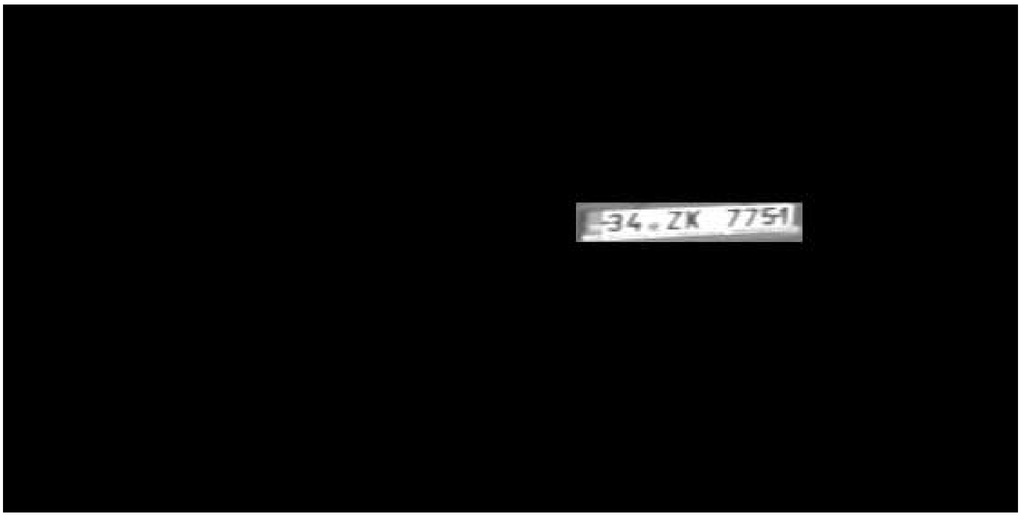
The results of the method with Daubechies filter are shown in table 4.6. the method also sometimes finds more than one candidate including the correct license plate location. Table 4.7 shows the number of candidates for the successful results.

(a)

Figure 4.30 LPS with wavelet transform: (a) Original image; (b) Result of the wavelet transform by Daubechies filter; (c) Result of performing plate extraction.



(b)



(c)

Figure 4.30 LPS with wavelet transform: (a) Original image; (b) Result of the wavelet transform by Daubechies filter; (c) Result of performing plate extraction.

Table 4.6 Performance of the algorithm with Daubechies filter

Correct license plate location;	%93.1
False or none license plate location;	%6.9

Table 4.7 The number of candidates for successful results of the algorithm with Daubechies filter

1 Candidate	%80.7
2 Candidates	%10.2
3 or more Candidates	%2.2

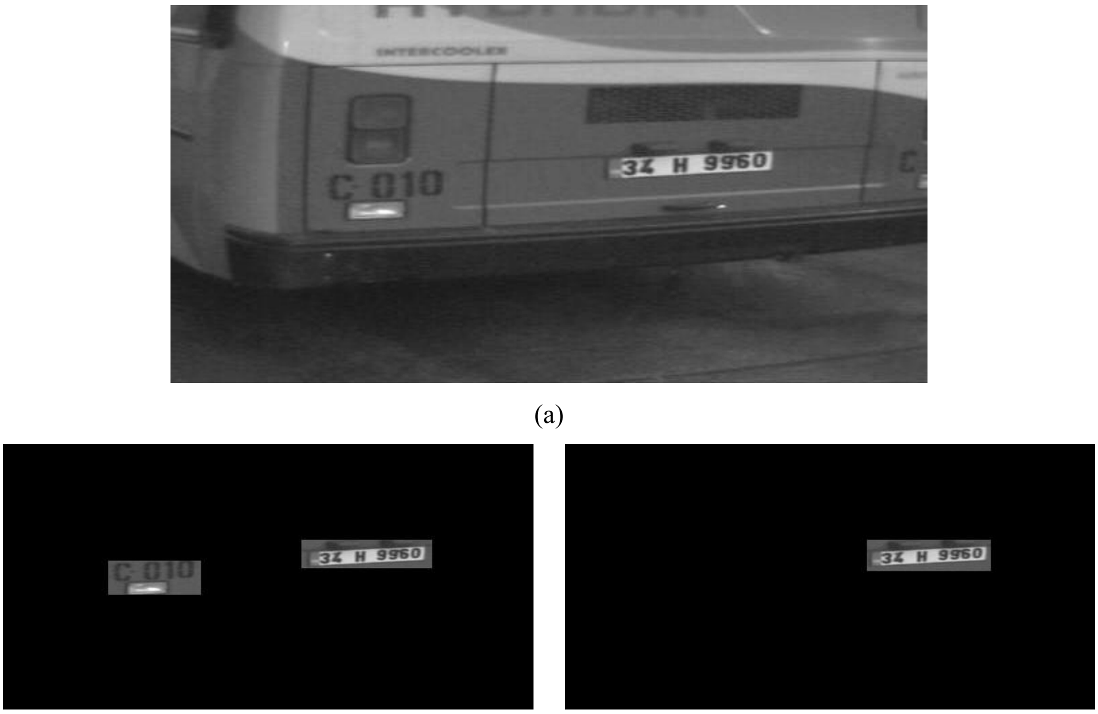
4.2.3 Comparison of the Two Filters

Table 4.8 shows the comparison of the two filters (Haar & Daubechies). The method with Haar filter requires 0.5 seconds for the operation and the method with Daubechies 3 filter requires 0.7 seconds for the operation.

Table 4.8 Comparison of the two filters

	Haar Filter	Daubechies 3 Filter
Correct license plate location;	%90.8	%93.1
False or none license plate location;	%9.2	%6.9
1 Candidate	%75.3	%80.7
2 Candidates	%11.8	%10.2
3 Candidates	%3.7	%2.2

Figure 4.31 and figure 4.32 show the results of the two filters, on two images for a comparison.

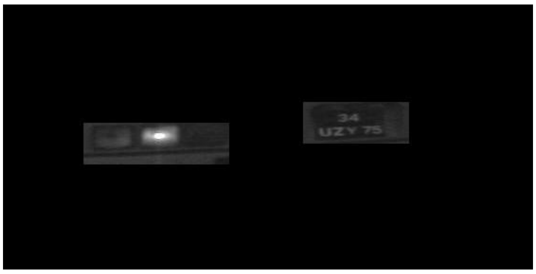


(a)

(b) (c)

Figure 4.31 Comparison of the two filters: (a) Original image; (b) Result of LPS with Haar filter; (c) Result of LPS with Daubechies filter.

(a)



(b) (c)

Figure 4.32 Comparison of the two filters: (a) Original image; (b) Result of LPS with Haar filter; (c) Result of LPS with Daubechies filter.

4.3 License Plate Segmentation with MVE

This method is described in subsection 3.3. The following figures show the main steps of the method. The original image is shown in figure 4.33. Figure 4.34 shows the result of performing vertical edge detection.

Figure 4.33 Original image.

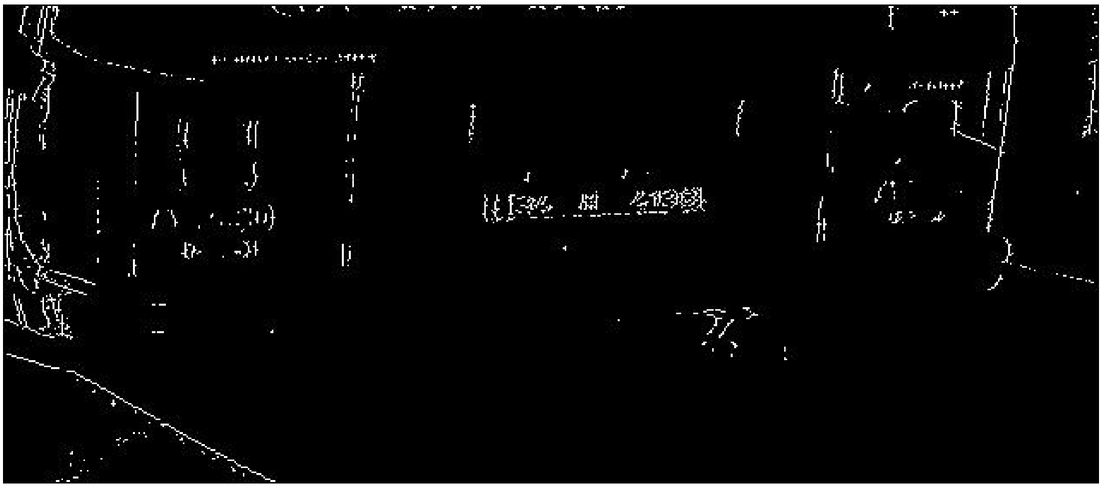


Figure 4.34 Result of performing vertical edge detection.

Figure 4.35 First candidate in the image.

Figure 4.36 First three candidates in the image.

Figure 4.35 shows the strongest candidate in the image. Figure 4.36 shows the first three candidates ordered by the strength in the image. And finally, figure 4.37 shows the first nine candidates in the image.

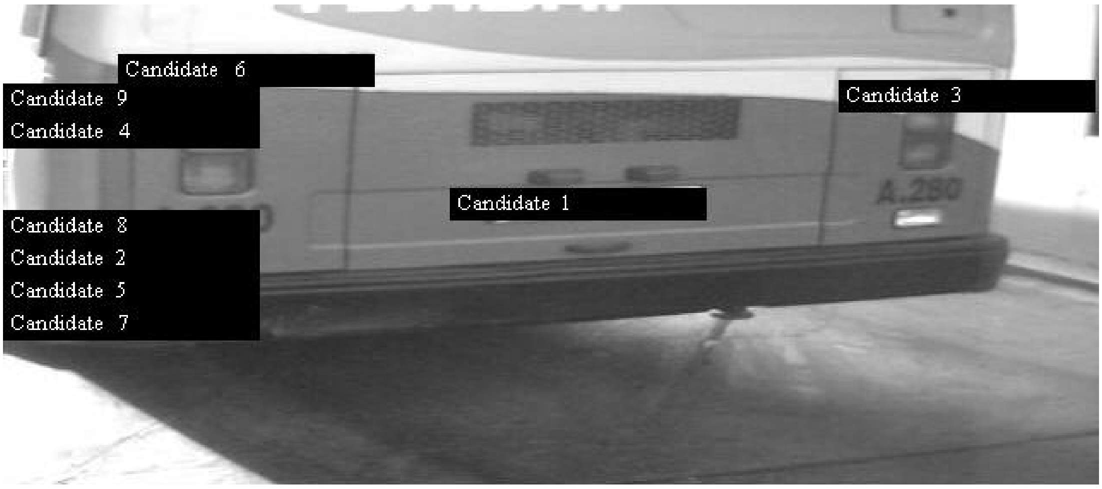


Figure 4.37 First nine candidates in the image.

The results of the method are shown in table 4.9. The license plates which have two lines can not be found by the MVE method cause of the shape of window used in method. Only %5.8 of the sample set have double lines. Table 4.10 shows the success ratio and required time of the algorithm for different number of candidates.

Table 4.9 The performance of LPS with MVE algorithm

Correct license plate location;	%92.5
False or none license plate location;	%1.7
License plates with double line	%5,8

Table 4.10 Success ratio and required times for different number of candidates

	Success ratio	Required seconds for algorithm
1 Candidate	%82.6	12 seconds
3 Candidates	%91.2	35 seconds
9 Candidates	%92.5	102 seconds

CHAPTER FIVE
CONCLUSION

In this study, three different LPS algorithms are applied to the test set which was taken from ASELSAN. These are real images taken on the Istanbul bridges. There is no any image which was taken out of the test set for any reason. So any LPS method is expected to be successful on both unofficial and official license plates and both single and double line license plates.

License plate segmentation with the technique in OCR has many disadvantages which cause to unsuccessfulness. There are some images on which histogram equalization and local thresholding cause loss of the information in license plate regions. Histogram equalization is becomes unsuccessful when background of a license plate and the characters on license plate have adjacent gray levels and images are taken on night yielding too many dark pixels. Also there are some problems when determining the threshold value automatically.

The Smearing algorithm used in this method is actually a page segmentation technique. Because of that the smearing algorithm is useful for segmenting dark characters on light background. Due to the Highway Traffic Law of Turkey which determines the colors of license plates, only license plates of unofficial cars conform to this case. There are also license plates with dark background and light characters for which this algorithm is not useful to determine them in the image.

The Smearing algorithm detects horizontal lines so that license plates must be aligned with the horizontal edges of the images. But usually the license plates in the images are rotated by an angle with respect to the edges. The angle between license plate and the horizontal edge of image is not constant at all time and rotating the images by a fixed angle before processing is not a solution. The slope of the license plates decreases the success of the algorithm. Detection of slope and then rotation of the image before smearing algorithm will increase the success of algorithm.

The algorithm with global thresholding is successful only on one third of the images. So this algorithm is not useful for the license plate segmentation. But some alterations increase the success of the algorithm. The algorithm is useless on night pictures. If the illumination of the images is increased, the success of the algorithm will increase.

The algorithm does not work for license plates of official vehicles. For the official vehicles the algorithm may be implemented twice. At the second implementation the reverse of binary image is used. So the dark backgrounds of plates transform to light, light characters on license plates transform to dark. But this increases the operation time of the implementation.

The algorithm takes time between 2.1 and 2.5 seconds. Because of vertical and horizontal smearings, there are two loops. These loops increase the operation time because of Matlab's slow response to loops.

Basic adaptive thresholding is used to increase the success of LPS with the Smearing algorithm. The success of the method is doubled with this. The required time for the algorithm increases by only 0.2 seconds.

Table 4.8 shows comparison of the two filters (Haar and Daubechies) in the wavelet transform method. Daubechies filter provides more successful results than Haar filter. But Daubechies filter takes more time than Haar filter. For both of the filters, the algorithm takes less than a second.

The disadvantage of the wavelet transform method is finding more than one region as a license plate. We need a second algorithm to eliminate the fake regions and find the correct region.

License Plate Segmentation with MVE has the best success ratio. Main disadvantage of LPS with MVE method is it's useless for images of a car with double line license plate. To overcome this problem a second window with a suitable size for double line license plates can be also run on the image during the recursive scanning process. The candidate areas for both type of windows are found and the order off all candidates are performed on the basis of highest vertical edge pixel numbers in any window.

REFERENCES

Barosso, J., Rafae,l A., Dagless, E.L., & Cruz, J.B., (1997) Number plate reading using computer vision, *IEEE – International Symposium on Industrial Electronics ISIE'97- GuimarSes*, p. 761-766

Bow, S.T. (2002). *Pattern Recognition Image Preprocessing: Second Edition, Revised and Expanded*. New York: Marcel Dekker Incorporated

Burdurlu, H. & Ertürk, S. (2004). Dalgacık Uzayında Blok Bazlı İmge Damgalama. *IEEE 12. Sinyal İşleme. ve. Uygulamaları Kurultayı*, p. 327-330

Cinsdikici, M.G. (2004) *Akıllı Trafik Sistemleri İçin Araç Plakası Tanıma Modülü Tasarımı ve Geliştirilmesi*. Ege Üniversitesi Fen Bilimleri Enstitüsü

Çapar, A., Beratoğlu, M.S., Taşdemir, K., Kılıç, Ö. & Gökmen, M. (2003). İTÜ Araç Plaka Tanıma Sistemi. *IEEE 11. Sinyal İşleme ve İletişim Uygulamaları Kurultayı,* p. 371-374

Draghici, S. (1997). A neural network based artificial vision system for licence plate recognition. 5 December 2007, http://vortex.cs.wayne.edu/papers/ijns1997.pdf

Duman, Ş., Öktem, R. & Çetin, A.E. (2005). Plaka Tanıma Amaçlı Alternatif Öznitelik Çıkarma Yaklaşımları. *IEEE 13. Sinyal İşleme ve İletişim Uygulamaları Kurultayı, SIU05*

Gonzalez, R.C. & Woods, R.E. (2002). *Digital Image Processing* (2nd ed.). New Jersey: Prentice-Hall

Gonzalez, R.C., Woods, R.E. & Eddins, S.L. (2004). *Digital Image Processing Using MATLAB*. New Jersey: Pearson Prentice Hall

Hermida, F., Rodrìguez, F.M., Lijò, F., Sande, F.P., & Iglesias, M.P., (1997) A System for the Automatic and Real Time Recognition of V.L.P.'s, *Lecture Notes on Computer Science, Vol. 1311*, p. 552-558

Hofman, Y. (May 2, 2004). *License Plate Recognition - A Tutorial*. Retrieved January 29, 2008, from http://www.licenseplaterecognition.com/#WhatIs

Kahraman F., Gökmen M., (2003) Gabor Süzgeçler Kullanılarak Taşıt Plakalarının Yerinin Saptanması, *Sinyal İşleme ve Uygulamaları Kurultayı*, p 317-322

Kahraman, F., Kurt, B., Gökmen, M., (2003) License Plate Character Segmentation Based on the Gabor Transform and Vector Quantization, *ISCIS 2003*, p. 381-388

Morse, B.S., (January 12, 2000). *Lecture 4: Thresholding*. Retrieved October 16, 2007, from, http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/MORSE/threshold.pdf

Parisi, R., Claudio, E.D.D., Lucarelli, G., & Orlandi, G., (1998) Car Plate Recognition by Neural Networks and Image Processing, *IEEE Proceeding of Int. Symposium on Circuits and Systems*, p. 195-198

Shafait, F., Keysers, D. & Breuel, T.M., (n.d.). *Performance Comparison of Six Algorithms for Page Segmentation*. Retrieved October 23, 2007, from, http://pubs.iupr.org/DATA/2005-IUPR-28Oct_1101.pdf

Ter Brugge, M.H., Stevens, J.H., Nijhuis, J.A.G., & Spaanenburg, L., (1998) License Plate Recognition Using DTCNNs, *1998 Fifth IEEE international Workshop on Cellular Neural Networks and their Applications*, England

Wikipedia. (January 29, 2008) *Automatic number plate recognition*. Retrieved January 30, 2008, http://en.wikipedia.org/wiki/Automatic_number_plate_recognition

Zunino, R., & Rovetta, S., (2000) Vector Quantization for License-Plate Location and Image Coding. *IEEE Transactions on Industrial Electronics*, *47*, 159-167