AN ALGORITHM FOR MULTISCALE LICENSE PLATE DETECTION AND
RULE-BASED CHARACTER SEGMENTATION

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ALİ ONUR KARALI

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

SEPTEMBER 2011

Approval of the thesis:

## AN ALGORITHM FOR MULTISCALE LICENSE PLATE DETECTION AND RULE-BASED CHARACTER SEGMENTATION

submitted by **ALİ ONUR KARALI** in partial fulfillment of the requirements for the degree of **Master of Science  in Electrical and Electronics Engineering  Department, Middle East Technical University** by,

Prof. Dr. Canan ÖZGEN
Dean, Graduate School of **Natural and Applied Sciences**　——————————

Prof. Dr. İsmet ERKMEN
Head of Department, **Electrical and Electronics Engineering**　——————————

Assist. Prof. Dr. İlkay ULUSOY
Supervisor, **Electrical and Electronics Engineering Dept., METU**　——————————

**Examining Committee Members:**

Prof. Dr. Gözde BOZDAĞI AKAR
Electrical and Electronics Engineering Dept., METU　——————————

Assist. Prof. Dr. İlkay ULUSOY
Electrical and Electronics Engineering Dept., METU　——————————

Prof. Dr. Aydın ALATAN
Electrical and Electronics Engineering Dept., METU　——————————

Prof. Dr. Nihan KESİM ÇİÇEKLİ
Computer Engineering Dept., METU　——————————

Dr. Tayfun AYTAÇ
TÜBİTAK BİLGEM UEKAE-İLTAREN　——————————

**Date:**　——————————

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name:    ALİ ONUR KARALI

Signature            :

# ABSTRACT

AN ALGORITHM FOR MULTISCALE LICENSE PLATE DETECTION AND
RULE-BASED CHARACTER SEGMENTATION

KARALI, Ali Onur

M.Sc., Department of Electrical and Electronics Engineering

Supervisor   : Assist. Prof. Dr. İlkay ULUSOY

September 2011, 56 pages

License plate recognition (LPR) technology has great importance for the development of Intelligent Transportation Systems by automatically identifying the vehicles using image processing and pattern recognition techniques. Conventional LPR systems consist of license plate detection (LPD), character segmentation (CS) and character recognition (CR) steps. Successful detection of license plate and character locations have vital role for proper LPR. Most LPD and CS techniques in the literature assume fixed distance and orientation from the vehicle to the imaging system. Hence, application areas of LPR systems using these techniques are limited to stationary platforms. However, installation of LPR systems on mobile platforms is required in many applications and algorithms that are invariant to distance, orientation, and illumination should be developed for this purpose. In this thesis work, a LPD algorithm that is based on multi-scale vertical edge density feature, and a character segmentation algorithm based on local thresholding and connected component analysis operations are proposed. Performance of the proposed algorithm is measured using ground truth positions of the license plate and characters. Algorithm parameters are optimized using recall and precision curves. Proposed techniques for each step give satisfying results for different license plate datasets and algorithm complexity is proper for real-time implementation if optimized.

# ÖZ

## ÇOK ÖLÇEKLİ PLAKA TESPİT VE KURAL TABANLI KARAKTER BÖLÜTLEME İÇİN ALGORİTMA

KARALI, Ali Onur

Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü

Tez Yöneticisi    : Yrd. Doç. Dr. İlkay ULUSOY

Eylül 2011, 56 sayfa

Akıllı trafik sistmelerinin geliştirilmesinde otomatik plaka tanıma (OPT) teknolojisi, araçların görüntü işleme ve örüntü tanıma teknikleri kullanılarak otomatik tanınmasını sağlaması bakımından büyük öneme sahiptir. OPT sistemleri genellikle plaka tespiti (PT), karakter bölütleme (KB) ve karakter tanıma (KT) basamaklarından oluşmaktadır. Plaka ve karakterlerin konumlarının başarılı tespitinin doğru OPT işlemi için rolü büyüktür. Literatürde yer alan birçok plaka tespit ve karakter bölütleme tekniği görüntüleme sisteminden araca sabit uzaklık ve yönelim varsaymaktadır. Dolayısıyla, bu teknikleri kullanan OPT sistemlerinin kullanım alanları sabit platformlarla sınırlıdır. Buna rağmen, birçok uygulamada OPT sistemlerinin hareketli platformlar üzerine kurulması gerekmektedir ve bu amaç doğrultusunda uzaklığa, yönelime ve aydınlanmaya karşı değişimsiz algoritmaların geliştirilmelidir. Bu tez çalışmasında, çok ölçekli dikey kenar yoğunluğu özelliğine dayalı bir plaka tespit algoritması ve yerel eşikleme ve bağlantılı bileşen analizine dayalı bir karakter bölütleme algoritması önerilmektedir. Önerilen algoritmanın başarımı plaka ve karakterlerin denektaşı pozisyonları kullanılarak ölçülmektedir. Algoritma parametreleri anımsama ve kesinlik eğrileri kullanılarak eniyilenmiştir. Her basamak için önerilen teknikler farklı plaka veritabanlarında tatmin edici sonuçlar vermektedirler ve algoritmanın karmaşıklığı gerçek zamanlı uygulama için uygundur.

Anahtar Kelimeler: plaka tespit, karakter bölütleme, görüntü işleme

*For Diren, who´s been making the world a better place for me since 2010.*

# ACKNOWLEDGMENTS

I would like to offer my deep and sincere gratitude to Asst. Prof. Dr. İlkay Ulusoy for her precious advice, guidance and the continuing support she has shown throughout this research, and throughout my study at METU. Without her encouragements, her enthusiasm, and her unselfish help, I could never finish my master work.

I am greatly indebted to Dr. Tayfun Aytaç for his wonderful supports throughout my thesis research. He has been a patient advisor and his expertise in the area of Image Processing, intellectual inputs, and ideas have helped shape my research. He has spent countless hours working with me and has inspired and motivated me to finish on time.

I would like to thank to the members of the committee, Prof. Dr. Gözde Bozdağı Akar, Prof. Dr. Aydın Alatan and Prof. Dr. Nihan Kesim Çiçekli. Their comments and suggestions have enhanced the quality of my thesis.

I am also thankful to my colleagues and friends, O. Erman Okman, Serdar Çakır, Özgür Aktekin, Betül Aydın, Alper Arıtaş, Selim Çağatay, M. Tufan Ayhan, Ümit N. Şen, Halil Uysal and Nurpınar Akdeniz, and for all the helpful discussions that I had with them over the years. My gratitude also goes to all the staffs of TÜBİTAK BİLGEM UEKAE/İLTAREN.

I owe a huge debt of gratitude to my parents for being very supportive of all my decisions in my life. They have always put my needs ahead of theirs and they are my inspiration for good work ethics and diligence.

# TABLE OF CONTENTS

# LIST OF TABLES

TABLES

# LIST OF FIGURES

FIGURES

# CHAPTER 1

# INTRODUCTION

Several issues in intelligent transportation systems (ITS) require development and implementation of automatic license plate recognition (LPR) systems as a solution of autometic vehicle identification and ongoing researches on more successful LPR systems have been conducted for decades. Continuous developments in image sensor and processor technologies should be taken into account in the algorithm development process and robust algorithms should be developed for mobile LPR systems as well as stationary ones to identify vehicles while the installed platform keeps moving.

LPR systems can be considered in two different groups with respect to their installation platforms: Stationary LPR systems and mobile LPR systems. Stationary LPR systems take advantage of the prior knowledge of the position and size of the license plate (LP) in the scene. Additionally, illumination can be controlled with the help of visible and/or infrared light sources. For this reason, accurate detection and recognition of LPs at electronic toll gates, parking lot entrances and highway speed control poles is accomplished and their implementations become widespread day to day.

Second group of LPR systems are mounted on mobile platforms, usually police cars, and the LPs of the around vehicles are recognized while the mounted vehicle keeps on driving. Mobile LPR systems still require performance enhancement and newly developed algorithms should take the advantage of ongoing technological advancements in the resolution and speed of the imaging systems. Successful LPR system mounted on a mobile platform should be invariant against the distance and the orientation to the target vehicles. Furthermore, actions should be taken in the algorithm against illumination change of the environment since effectiveness of the external light sources decrease due to the uncontrolled target vehicle distance

and orientation. For this purpose, edge-based techniques are preferred to the intensity and color based techniques for the detection and recognition of LPs when the system is planned to be installed on mobile platforms. Besides, multi-scale approaches are adopted to compensate distances variation from the imaging system to the vehicle, and also, high resolution imaging systems increase the performance of the multi-scale approaches. Another feature that a mobile LPR system should have is the detection of multiple license plates on a single image since most scenarios include street views of the roads and parking lots. However, stationary LPR algorithm scenarios can be restricted to include only a single LP on the image.

General structure of an LPR system consists of three based steps known as license plate detection (LPD), character segmentation (CS) and character recognition (CR). LPD step aims to find the exact location of the LP in the image and this step outputs the bounding box coordinates of them in the image. Second step of the LPR system is known as CS and this step feeds the CR algorithm with the exact positions of each character inside the LP region. In fact, if license plate detection and character segmentation algorithms are combined, binarized version of the license plate region can also be sent to the character segmentation step. Final step of the LPR system is the CR step and optical character recognition (OCR) algorithms are implemented in this step. Most algorithms in the literature consider each step as a separate problem and bring separate solutions to each one. Overall performance of such systems are calculated as the multiplication of the performance of each step. LPR systems with feedback between the steps give better results in some scenarios but still, false results obtained in one step result in false results in the next step and feedback information becomes useless.

## 1.1 Scope of this Work

In this thesis work, an algorithm is proposed for the LPD and CS steps of an LPR system. For LPD, a multiscale vertical edge based density technique is proposed. Nextly, a CS technique, which is based on connected component analysis operations by merging intra and inter character properties, is proposed. Results of each step is tested using images from different datasets which cover different scenario conditions.

In the Chapter 2, previous studies on LPR system are summarized. Implementation areas of the LPR systems are described and effectiveness of these systems are reported in the first part

of the chapter. Nextly, LPD algorithms proposed in the last two decades are briefly described in two groups as region-based and edge-based techniques and briefly described. Then, character segmentation algorithms existent in the literature are introduced and examined in two different groups, projection-based CS techniques and CCL-based CS techniques, for the ease of the reader.

In the Chapter 3, proposed algorithms for each step are explained step by step. Each step of the algorithm is described in detail, theoretical backgrounds are explained using equations and block diagrams, and results are illustrated using figures. Step results are implemented on sample images to illustrate scale, orientation and illumination invariance.

In the Chapter 4, firstly, test images from three different datasets are introduced, and type of the datasets, with reasons for selection, are explained. Secondly, simple GUI based software for ground truth construction is introduced. Then, performance metric used to measure the effectiveness of the algorithm is introduced. Finally, performance of the algorithm is evaluated and illustrated using precision and recall curves for different algorithm parameters. Overall performance of the system is discussed at the end of the chapter.

In the Chapter 5, performance of the algorithms for the image datasets are discussed, conclusions are done and future works are described.

# CHAPTER 2

# LICENCE PLATE DETECTION AND CHARACTER SEGMENTATION

## 2.1    Image Processing For Intelligent Transportation Systems

Recent advancements in the image sensor technology with the increasing number of manufacturer companies introduce the inexpensive digital cameras to our daily lives. Integration of these novel hardware technologies with the state of the art computer vision algorithms leads to the design and implementation of several intelligent systems for the good of the societies. ITS is one of the application areas of intelligent systems that increases the efficiency and safety of transportation [1]. Utilization of the visual information acquired from image sensors becomes widespread and vital for the management of transportation problems that arise from increasing number of vehicles. Traffic flow control, driver assistance and situational awareness are achieved with the emerging information. License plate recognition (LPR) is a crucial technology to deploy safety and control ability for ITS using visual information by providing automatic identification of the vehicles. For this reason, many LPR algorithms have been developed and still there is a need for further developments to cover different scenarios.

Today, LPR systems are used for many different purposes and they can be installed on both stationary and mobile platforms. Application areas of these systems can be seen in the below list:

- Speed control in highways,

- Identification of traffic light violations,

- Automated payment systems (billing and ticketing) for bridges, tunnel, highway and

parking lots,

- Identification and permission of vehicles in security areas such as government buildings and military zones,

- Analysis of traveling time and average speed,

- Observation of stolen vehicles,

- Situational awareness,

- Collection of large amount of data to be used in search and evidence gathering activities.

Above benefits increase the need for reliable LPR algorithms and numerous LPR techniques are proposed in the literature. Reliability and efficiency of these techniques are crucial and detailed surveys on these LPR systems can be found in the literature [2, 3]. Lum explains the usage of license plates in USA for security applications in [2]. With the help of the beneficial discussions made in this report, one can have a better insight into LPR technology and impact of it on social security area. Results of the Lum's work show that usage of LPR is an efficient solution for the detection of stolen vehicles, motor vehicle violations and further investigations by connecting the LPR output with databases, monitoring and recording vehicles in high crime locations, and monitoring security in high-risk locations. Lum investigated the deterrent effect of LPR systems on crimes but results show that there is not yet such a progress. Anagnostopoulos et al. [3] presents the most comprehensive survey on existent LPR algorithms in the literature. They divide an LPR system into license plate detection (LPD), character segmentation (CS) and character recognition steps and then performance of the proposed solutions for each step are presented. Anagnostopoulos's survey enables researchers to located their own work on LPR in the whole LPR technology and attempt to solve problems such as blurring effect of relative motion between the system and the target vehicles, existence of broken characters on the plate, inadequate resolution of the imaging system and complex backgrounds with many vertical edges, where many of them still remains unsolved.

LPR systems deployed in different platforms secure and simplify our daily lives. In fact, satisfying results are obtained, and benefits of these systems are proven. However, still there is a need for further improvements by incorporating the latest advances in signal processing and imaging technology areas. In the next section, structural analysis of license plates used in Turkey is given, and then detailed review of LPD techniques in the literature is given.

Figure 2.1: Vehicle license plate of Turkey [4].

## 2.2 License Plate Structure

License plates in Turkey are rectangular shape aluminium plates of size $11x52cm$ with a blue stripe on the left which shows that country is "European Customs Union" member. Figure 2.4 shows a sample LP [4].

Size of the blue stripe on the left is equal to $4x10cm$ and each character height is equal to 8cm. Character width varies between 0.2 to 0.7 times the height of the character. Black characters are used on white background, and for official vehicles, white characters are written on black background.

## 2.3 License Plate Detection

Detection of the boundaries of license plates in the scene is the first step of most LPR algorithms and known as license plate detection or localization in the literature. License plate detection (LPD) algorithms can be divided into two main categories : Region-based approaches and Edge-based approaches. In the following sections, we briefly describe the literature work for these two approaches.

### 2.3.1 Region-Based LPD Techniques

Earlier region based algorithms to detect license plate regions assume fixed size license plate region in the scene [5, 6, 7]. Naito proposes an adaptive thresholding technique to binarize the image, and utilizes the know character and license plate dimensions and orientations to extract candidate regions. Comelli et al. [6] finds the maximum contrast points over rectangular regions in the scene, which correspond to the license plate coordinates and defines that point as the center of license plate region. Barroso et al. [7] examines the vertical lines along

the image to locate the LP, and then the vertical projections of the binarized version of the image in the areas of interest. A peak-to-valley approach described in [8] is adopted to confirm licence plate location and character segmentation. Another earlier license plate detection algorithm based on region aspects is proposed by Zunino et al. [9] where image containing license plates is split into sub-blocks using local variance values and a set of stripes are obtained over these blocks. Final decision over license plate location is made using early training results and score assignment process. A license plate localization algorithm that depends on the statistical properties of the gray-level image is proposed by Anagnostopoulos et al. [10]. In the proposed method, two step binarization process is defined. In the first step, two sliding concentric windows are used to compute statistical values around each pixel and result is binarized using a user-defined threshold value. Result of the first binarization step is used as a mask for the original gray-level image and masked regions are binarized using the technique described in [11]. Obtained connected components (CC) are filtered using structural properties and remaining regions are detected as licence plate regions. Most region based techniques for LPD use local or global thresholding, followed by connected component labeling (CCL) operations as described above. Hmouz et al. [12] proposes a region based algorithm that combines the local thresholding result with the global thresholding results obtained from multiple threshold values. The algorithm has better results against illumination variance due to multiple thresholding technique. Hmouz models each CC feature as a random variable and assigns probabilities to the CCs obtained after global and local binarization operations. CC features used in this paper are width/height ratio of the region, character area, character/plate area density and number of edges in the plate center. A Gabor filter based license plate detection technique is proposed in [13]. Although, steps are not explained in detail, 12 Gabor filters are used that consist of three scales in four directions and result is binarized with a threshold value and morphological operations are applied to merge neighboring regions. Final CCL operation is used to segment each region separately.

Conci et al. [14] proposes a region-based algorithm to locate the plates in the scene which uses mathematical morphology and local Otsu thresholding technique [15]. In the first step of the algorithm, top-hat and bottom-hat morphological operations are applied to increase the contrast between the character and plate regions. Size of the structuring elements in the algorithm is selected with respect to the know LP character size. After binarizing the enhanced image, opening operations are applied to remove improper regions which have a width and

height outside the known plate width/height range.

### 2.3.2 Edge-Based LPD Techniques

A color edge based license plate region extraction technique is proposed in [16]. Authors define fuzzy maps using the hue, saturation- intensity and color edge components of the scene and aggregate the results by assigning weights for each fuzzy map to locate candidate regions. Guo et al. [17] proposes another edge-based technique, where $I$ component of the HSI color space is used to detect vertical and horizontal Sobel edges. Guo constructs the edge map of the image by computing the $l^2 - norm$ of the edges and then binarizes the edge map iteratively by selecting a threshold as the average values of the remaining edge map entries. Hung et al. [18] proposes a two step license plate detection algorithm, where in the first step, rough localization is achieved by analyzing the vertical detail coefficient of the Discrete Wavelet Transform of the gray scale image by using Haar Wavelet. In the second step of the algorithm, accurate location of the plate is found using horizontal detail coefficients and morphological operations. Gabor filter can also be used for license plate localization. Caner et al. [19] proposes a method that uses a predefined wavelength, orientation and threshold for binarization of Gabor filter output. Binarization operation is followed by CCL analysis to eliminate improper regions. Mahini et al. [21] defines a *bothat* operator as subtracting the original image from its morphological closing operation applied version to emphasize black regions on white backgrounds, and then applies vertical Sobel edge detection operation to detect vertical edges in the scene. Next, color information is merged with the vertical edge regions and result is binarized using a soft thresholding technique where threshold value is predetermined. Finally, some of the connected components are filtered using structural analysis. Yu et al [23] adopts Sobel edge detection and vertical and horizontal projections of the edge image to locate license plates. However, global projection based techniques are sensitive against background clutter and rotation of the license plate. Besides, size of the LP becomes extremely crucial in projection based localization techniques.

An edge based approach similar to the one proposed in this thesis work is proposed by Li et al. [22]. Li computes the vertical edges in the image after preprocessing operations and defines an energy image, where elements of the image are calculated as the average values of the vertical edge values in a region centered at a pixel. Size of the region is selected as the

assumed licence plate size. Energy image is segmented by checking neighborhood of the local extremas in the image. Sample energy image taken from this work can be seen in Figure 2.2. On the other hand, a multi-scale approach is adopted in this thesis work where edge density image is segmented using statistical values obtained from the multiscale processing of the vertical color edges in the scene.



<div align="center">(a)          (b)</div>
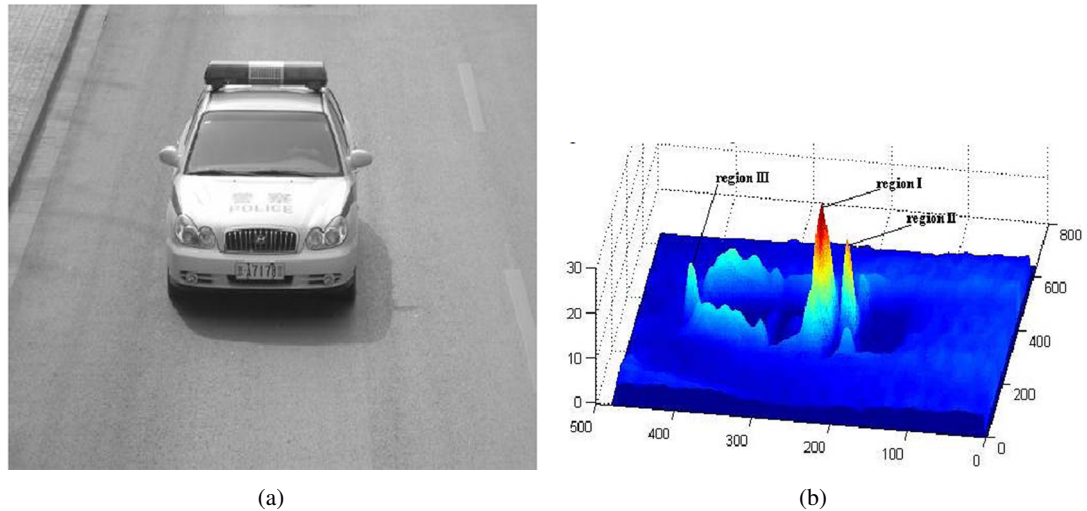
Figure 2.2: (a) Sample image, (b) Energy image (Source: Li et al. [22]).

## 2.4 Character Segmentation

CS is the second step of LPR systems and operation applied in this step outputs the exact location of each character inside the previously detected licence plate region. In most of the approaches brought to CS, initial binarization operation is applied to the license plate region, and then, vertical projections or CC aspects are used to segment characters.

In general, CS techniques can be divided into two different categories which will be described in the following sections: Projection-Based CS techniques, and CCL-Based CS techniques.

### 2.4.1 Projection-Based CS Techniques

Barroso et al. [7] proposes a vertical projection based character segmentation technique, which uses binary image of the plate region for vertical projection and seeks peak-to-valley

<div align="center">9</div>

structures in the projection. It's mentioned that the threshold value chosen is scene dependent and peak-to-valley parameters depend on the distance between the car and the imaging system. Cheng et al. [25] also uses the technique proposed by Barroso, and additionally applies an orientation adjustment operation and skew correction to compensate rotational effects. Rahman et al. [24] proposes a similar character segmentation technique to the previous one. Vertical projection of the LP region is used to find character upper and lower boundaries and horizontal projection is used to segment characters as seen on Figure 2.3. Although not mentioned, results show that binarization operation is applied before projection. The result of projection is also used in character recognition step by using a template matching operation between the license plate character projection and the previously defined character projection templates.



(a)                                        (b)
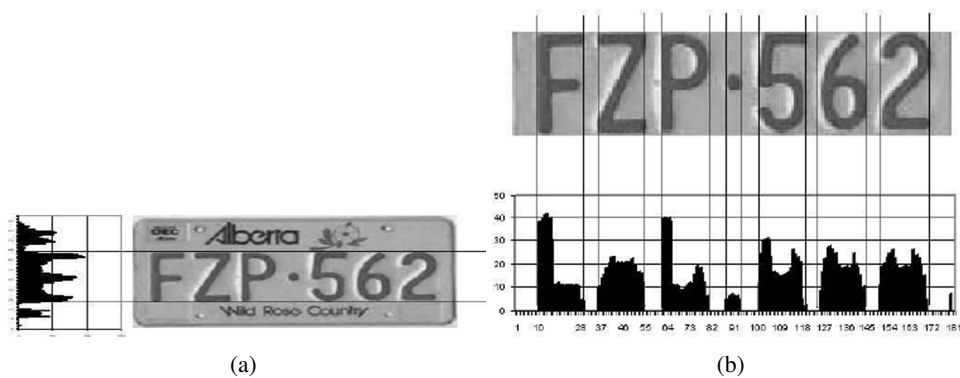
Figure 2.3: (a)Upper and lower boundary extraction through horizontal projection, (b) Character segmentation using vertical projection( Source : Rahman et al. [24] ).

Although projection based techniques are simple and fast for images as shown in Figure 2.3, in most of the cases plates are obtained rotated with broken characters and/or merged characters due to dirt or stickers and possibly have a lower resolution than the sample image shown in Figure 2.3. Projection based techniques suffer from rotation, motion-blur and dirt effects seriously and can cause erroneous segmentation results due to weak connections that exist for characters such as $\{'L','T','H'\}$. To avoid these errors, Shi et al. [27] proposes a detailed correction procedure that consists removal of non-character objects, top-bottom correction, tilt correction and merging unconnected characters after projection based segmentation. Shi also uses horizontal projection of the LP to decide whether it is single or double line. Another projection based method that adopts tilt correction and iterative CS is proposed by Tyan et

al. [26]. Tyan defines a cost function as sum of squared values of the projected edge values and finds the tilt angle which maximizes the cost function. Later, the characters are split and merged back with respect to the result of character recognition operation.

## 2.4.2 CCL-Based CS Techniques

Segmentation of an image enables us to group image pixels that have similar features and treat the whole group as a single object. Inside LP regions, the most distinguishing property between the LP characters and the background is the intensity difference between these two regions. Usually, although colors may be different due to regulations around the countries, there exists a high contrast between the characters and the LP background, and this difference is the mostly used cue to segment LP characters. Quantization, binarization and region growing operations are different ways of segmentation that are frequently used in LPR. Result of segmentation should be labeled to identify each region, and, for this reason, connected component labeling (CCL) operations are used. Features of the identified regions at the end of the CCL operation can be used to decide correct character regions and to remove erroneous regions appeared due to noise, dirt or other disruptive effects on the LP. Connected component analysis is adopted for many CS algorithms in the literature and rest of the section gives brief information about the earlier works that depend on CCL based CS.

Naito et al. [5] proposes a technique where, exact size of the characters are known and binarization result is looked for that regions. If missing characters exist, using the known license plate character placement, the region where missing character exists is found and sent to the recognition step. Chang et al. [16] binarizes the plate region using a local adaptive algorithm proposed by Nakagawa et al. [29] and eliminates the regions due to noise, by using region placement and region aspect ratio. Character alignment is determined using Hough transform and divergent characters are removed. Finally, total number of characters are limited to eight and if more characters are obtained, the ones with less area are eliminated. Guo et al. [17] proposes a hybrid binarization technique, which adopts two global thresholds that depends on the peak and valley of the licence plate region histogram and one local thresholding technique. Through top-down and left-right search processes, using LP size dependent kernels, LP frames and noise are partially eliminated. Hamey et al. [28] applies CS based on color, size and position properties of the characters. Their algorithm merges the information of the

expected placement and fixed width of each character. These information are used to separate merged characters and to merge broken characters. Giannoukos et al. [30] proposes an adaptive thresholding technique that is similar to the approach adopted in this thesis work. Mean and variance inside a *kxk* region centered at the examined pixel are found, and then, threshold value is defined as the sum of the mean and the variance values if the variance is more than the minimum variance in the scene. Then, they use minimum height, maximum height and aspect ratio properties to eliminate non-character regions. Initial character segmentation results are used as a mask and further CC analysis are conducted adopting LP template format. Steps of this algorithm can be seen in Figure 2.4.
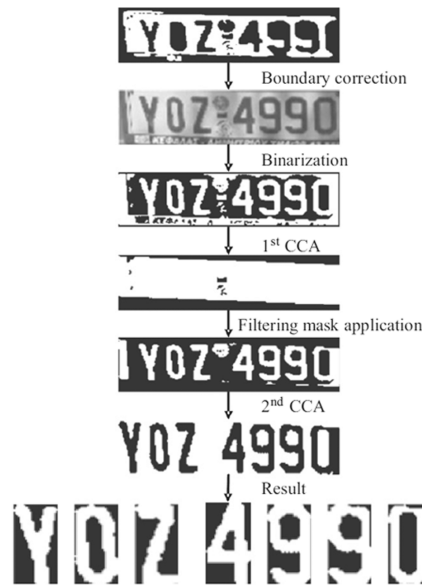


Figure 2.4: Sample CS operation sequence (Source: Giannoukos et al. [30]).

12

# CHAPTER 3

# PROPOSED ALGORITHM

LPR algorithms mostly consist of separate detection, CS and recognition steps as shown in Figure 3.1. Some adaptive and iterative versions of this structure are proposed in [20], but still performance of each step significantly affects the overall success of the whole system. In this chapter, details of the proposed algorithm for the first two steps of a LPR system, which are LPD and CS steps, are explained.

First part of this chapter explains an LPD algorithm which uses multi-scale vertical color edge features to detect candidate LP locations. Then, some of the candidates are eliminated using structural region analysis. In the second part of this chapter, a rule-based character segmentation algorithm which counts for the Turkish LP regulations, is proposed and algorithm steps are illustrated on sample images. Steps of each part can be seen in Figure 3.2.

## 3.1 License Plate Localization

In this thesis work, a multi-scale, vertical color edge density based LPD algorithm is presented. Number of pixels that a LP region spans shows variation due to the resolution of the imaging system, the range from the car to the imaging system, and optical lens characteristics of the system. For this reason, a multi-scale approach should be carried out to avoid strict



Figure 3.1: License plate recognition steps.

Figure 3.2: Flow chart of the algorithm.

scenario constraints on these three scenario parameters. Hence, as a first step of the detection algorithm, a scale space pyramid is constructed using original and down-sampled versions of the original color image. Then, a color edge detection technique that uses vector operations is applied to each level of the color image pyramid. But, in some cases, grayscale images are obtained from the imaging system and in these cases, first derivative of the i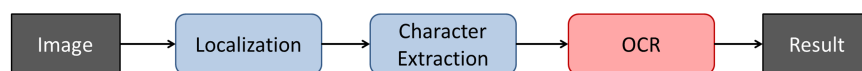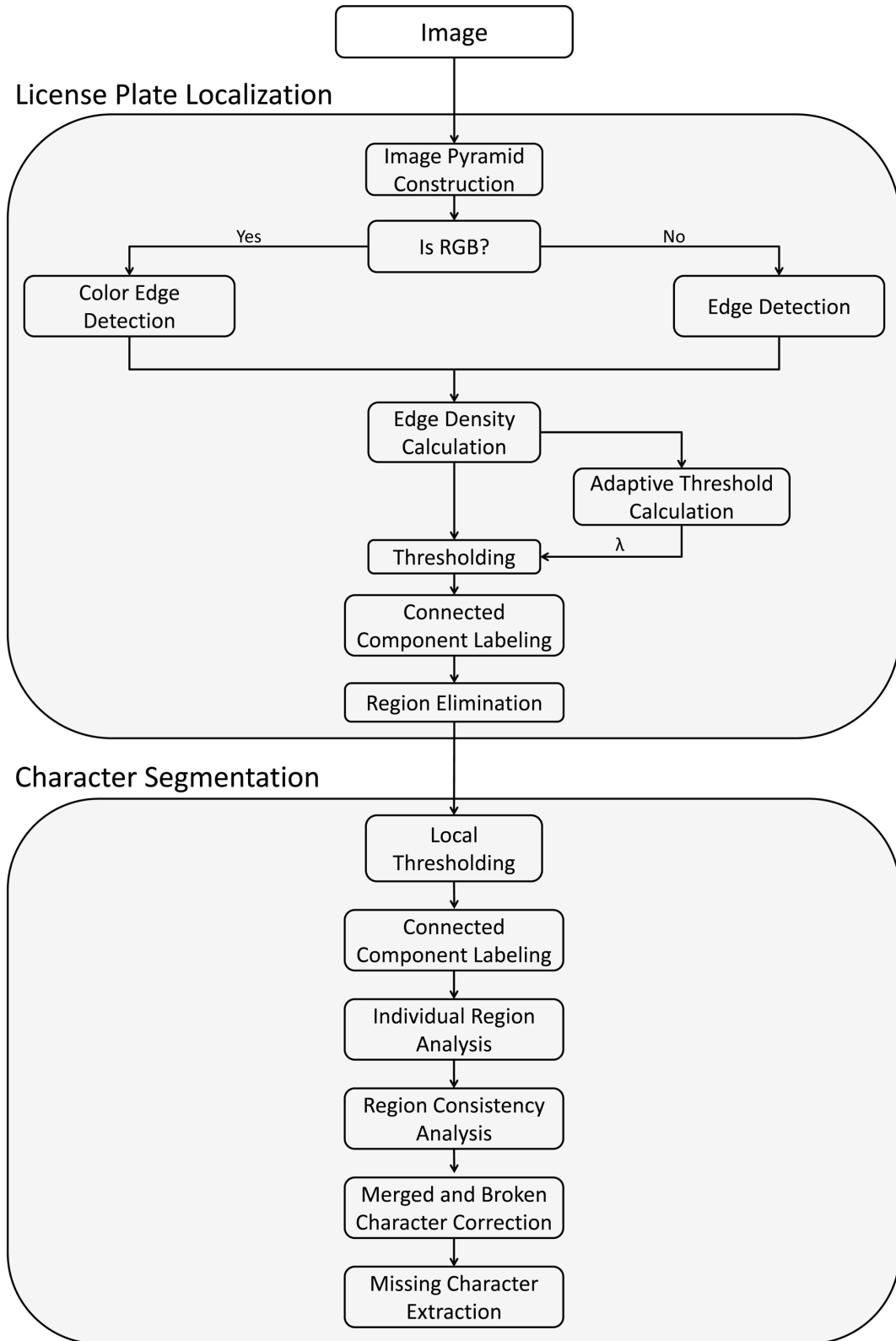mage in the horizontal direction is used to detect vertical edges. Next, vertical edge densities are calculated and noisy edge responses are eliminated using a rectangular shape averaging filter. Filtering results of the edge responses at different scales are fused by resizing and multiplication operations. Single channel, high dynamic range image acquired at the end of the multiplication operation is segmented using an adaptive thresholding technique. Candidate LP regions are obtained and refined using morphological operations. Finally, structural analysis of the candidate regions are held and improper regions are eliminated. Details of these operations are given in the following section.

### 3.1.1  Image Pyramid Construction

In this work, a multi-scale algorithm is developed because the size of the LP is assumed to be unknown. In other words resolution, field of view of the imaging system and the variable distance of the LP from the camera can change in a limited range by using this multi-scale technique. Multi-scale approaches are used in many robust feature detection techniques [31, 32]. For this purpose, an image pyramid is constructed using the down-sampled versions of the original color image. In this work, a 4-level image pyramid is constructed. $1^{st}$ level of the pyramid, denoted by $I^1$, is defined as the original color image. $2^{nd}$, $3^{rd}$ and $4^{th}$ levels of the pyramid ($I^2$, $I^3$ and $I^4$) are the resized versions of the original image by ratios 0.75, 0.5 and 0.25, respectively, and they are obtained by subsampling and linear interpolation operation. Number of image pyramid levels can be selected depending on the time and performance constraints, as well as recognition system parameters such as distance and resolution. Sample image pyramid constructed using this method is shown in Figure 3.3.

LP regions contain strong and frequent vertical edges due to the spatial ordering of the characters and plate background and foreground color difference. Considering this, vertical color edges are used as a cue for LPD and are extracted from each of the pyramid images. Detected vertical edges are used in region vertical edge density calculation. Results obtained from the

15

Figure 3.3: Color image pyramid ($I^1$, $I^2$, $I^3$ and $I^4$).

pyramid images are fused in the subsequent operations. Since a single edge detection filter is used in the next step, edges with different thickness values can be detected over this pyramid. Next section describes the color edge detection method applied in this work.

### 3.1.2 Color Edge Detection

Existence of the dense vertical edges is one of the cues used in the LPD. While detecting edges of the LPs, often grayscale edge detection techniques are preferred as explained in Section 2.3.2. Conversion from RGB to grayscale decreases the computational complexity, however, due to loss of color information, these techniques sometimes give erroneous results. To have a more robust technique, a color edge detection is used in this work. Each channel of an image pixel is represented as a component in a three dimensional vector space. Then, image pixels, which are represented as three dimensional vectors, are used to find edge responses ($E^n(x, y)$). $E^n$ is calculated as the magnitude of the difference vector calculated from adjacent image pixels in horizontal dimension where $n$ denotes the image pyramid level. In Figure 3.4, step by step color edge detection procedure is shown on a sample image $I$, which is separated into its R, G and B channels ($I_R$, $I_G$ and $I_B$).

First derivative of the image in horizontal direction is used to detect vertical edges. So, a filter, $f_h$, which is also known as Haar wavelet is used. This filter avoids thickness included by edge filters such as Sobel or Prewitt, but its noise filtering characteristic is worse than these. However, the averaging filter applied after edge detection removes noisy responses. By filtering the channels of the image $I$ with $f_h$, each component of the difference vector is obtained as seen in the below equation:

16
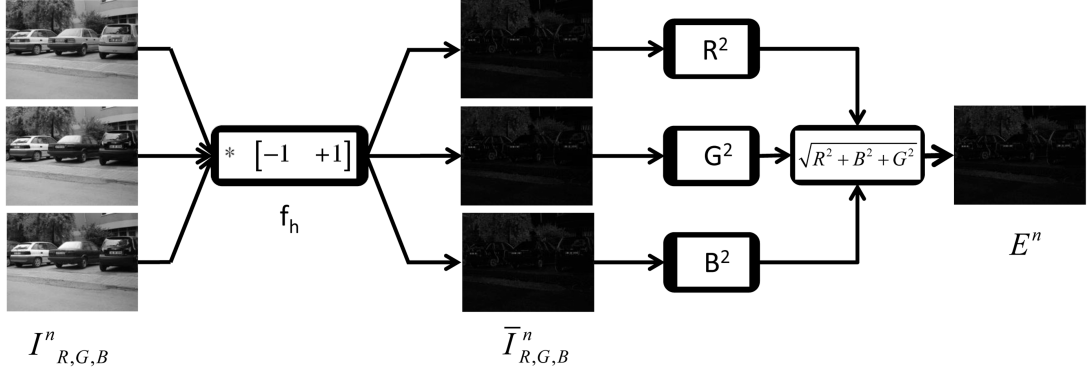
Figure 3.4: Color edge detection steps.

$$\bar{I}^n_{R,G,B} = I^n_{R,G,B} * f_h \tag{3.1}$$

First derivatives of the channel images are denoted as $\bar{I}^n_R$, $\bar{I}^n_G$ and $\bar{I}^n_B$ and $n$ is the level of the image in the pyramid. Elements of edge images $E^n$ are computed as seen in the Equation 3.2.

$$E^n(x,y) = \sqrt{(\bar{I}^n_R(x,y))^2 + (\bar{I}^n_G(x,y))^2 + (\bar{I}^n_B(x,y))^2} \tag{3.2}$$

LP regions in $E^n$ for all pyramid levels show vertical edge distribution. Color edge detection procedure is visualized in Figure 3.4 on a sample image taken from the image pyramid. But still there can be some other regions which include more frequent vertical edges when each level image is considered individually. To avoid false detections, information obtained in each pyramid level is fused and regions with stable edge characteristics along the pyramid are found. In the next step, vertical edge density images are calculated by further data fusion operations.

In the literature, some datasets consist of monochrome images and a grayscale edge detection technique is needed in that cases. In this thesis work, if a grayscale image is given as an input to the LPD step, $E^n$ is simply computed as the first derivative of the image $\bar{I}$ in the horizontal direction.

17

### 3.1.3 Vertical Edge Density Calculation

LP regions can be differentiated from other components of the scene using strong and frequent vertical edges. Some techniques count the number of edges or project edges along the vertical direction to find LP regions. However, these techniques are not robust against to low contrast and rotated plates. For this reason, edge response averaging approach is adopted in this study. An averaging filter $f_a$ is applied to the edge images to find vertical edge densities. Vertical edge density value is calculated for each pixel in the pyramid images as an average value of the edge responses in a rectangular area centered at that pixel. Size of the averaging filter $f_a$ depends on the resolution of the image and expected size of the LP region. So, a rectangular filter of size $k_{f_a}$ by $f_{f_a}$ is used in vertical edge density calculation to get maximum response around license plate regions. $k_{f_a}$ by $f_{f_a}$ are defined as given in Equation 3.3 where $W$ and $H$ correspond to image horizontal and vertical resolutions, respectively. Filter size is simply adapted to the expected resolution of the original input image using the multiplication coefficient $\tau$. $\tau$ is defined as in Equation 3.3, because, nominal horizontal resolution of the images is assumed to be 640 and values of the parameters are simply adapted to the changes of the image resolution. With the help of this filter, impulsive edge responses are filtered out which are added with the usage of filter $f_h$. After filtering with $f_a$, results are resized to the original image size ($W$ by $H$) to fuse edge density information obtained from the pyramid images. Finally, resized edge density images are element-wise multiplied to combine vertical edge density results obtained from the whole pyramid. Above operations are illustrated in Figure 3.5 for a sample image that contains LPs.

$$
\begin{aligned}
\tau &= \frac{W}{640} \\
k_{f_a} &= 5 * \tau \\
l_{f_a} &= 30 * \tau
\end{aligned}
\tag{3.3}
$$

$$
f_a = \frac{1}{k_{f_a}.l_{f_a}} \cdot \begin{bmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1 \end{bmatrix}_{k_{f_a} x l_{f_a}}
$$

18

$$\bar{E}^n = resize(E^n * f_a) \tag{3.4}$$

$$R = \prod_n \bar{E}^n \tag{3.5}$$

where $\bar{E}^n$ denotes the $n^{th}$ level, resized vertical edge density image as n goes from 1 to 4 and $R$ denotes the resultant edge density image. Multiplication is used to fuse edge density information since we expect to obtain strong vertical edges around LP regions in each of the pyramid images. As seen in Figure 3.5, in the LP regions $R$ takes high values. Segmentation of the image $R$ using statistical values of the element values is explained in the following section.

### 3.1.4 Candidate Region Segmentation and Correction

In the previous sections, vertical edge density responses are fused and $R$ is obtained as a result. LP region appears as a peak in $R$ and segmentation process is needed to differentiate LP regions from the background. The most basic segmentation method is thresholding against a constant threshold value. However, such a technique depends on scene characteristics and fails in images with different backgrounds and illumination conditions. In this thesis work, an adaptive thresholding technique that depends on the statistical properties of the scene is applied. A threshold value for the scene, $\lambda$, is computed by multiplying the standard deviation of the elements of $R$, $\sigma$, by a coefficient $k$ and adding the result to the mean value of the elements of $R$, $\mu$. Calculation of mean $\mu$, standard deviation $\sigma$ and the overall threshold value $\lambda$ is seen below:

$$\mu = \frac{1}{W \cdot H} \sum_x \sum_y R(x, y) \tag{3.6}$$

$$\sigma = \left( \frac{1}{(W \cdot H) - 1} \sum_x \sum_y (R(x, y) - \mu)^2 \right)^{\frac{1}{2}} \tag{3.7}$$

$$\lambda = \mu + k \cdot \sigma \tag{3.8}$$

After calculating the threshold value, binary image $R_b$ is obtained by thresholding $R$ with $\lambda$. Sample image segmentation result for $k = 2.5$ can be seen in Figure 3.6.
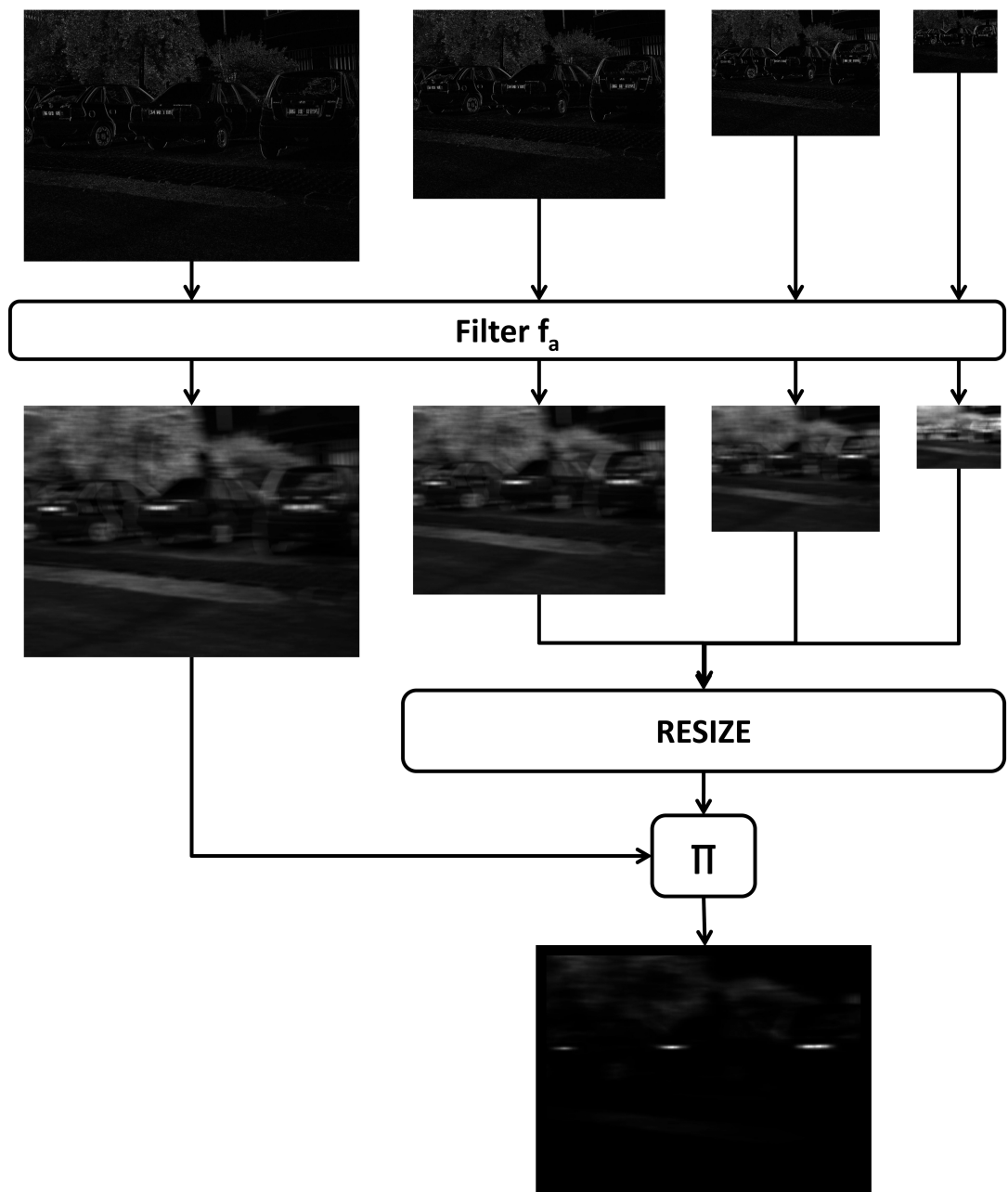
19

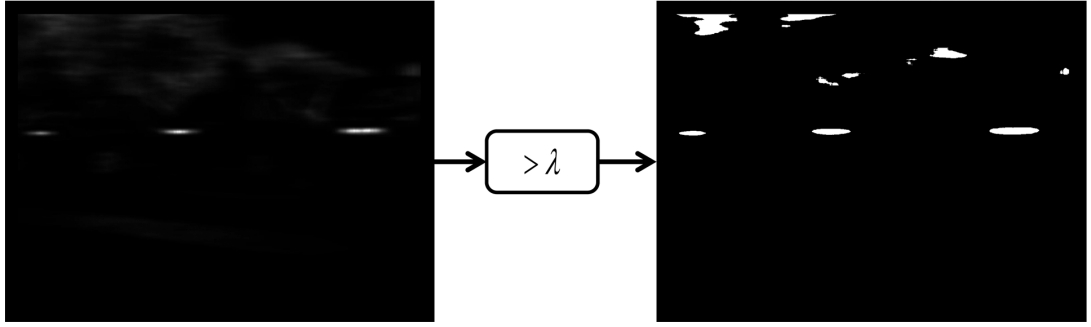Figure 3.5: Vertical edge density image calculation steps.

Figure 3.6: Binary segmentation of candidate license plate regions.

Adaptive selection of threshold $\lambda$ gives better results for the images taken under nonuniform illumination and localization conditions. However, existence of highly cluttered background objects such as tree branches, guard rails or wire meshes result in a dense vertical edge density regions. These regions increases the threshold level and LP regions can not be detected.

After thresholding operation, correction of the segmentation result $R_b$ is needed using morphological operations. For this purpose, binary erosion and dilation operations are applied sequentially using different sized structuring elements. Firstly, a binary erosion operation using a 3-by-3 square structuring element eliminates the small regions and then binary dilation operation with an adaptive size $m$ by $n$ is applied to merge neighboring candidate LP regions. After detection and correction steps, candidate regions are analyzed and eliminated.These operations are explained in detail in the next section. Effect of binary morphological operations is shown in Figure 3.7.

$$
\begin{aligned}
m &= 10 * \tau, \\
n &= 40 * \tau
\end{aligned}
\tag{3.9}
$$

### 3.1.5 Candidate Region Labeling and Elimination

After morphological operations, each connected component of $R_b$ is structurally analyzed and final candidates are obtained. After segmentation operation, some regions are obtained that are not suitable to be an LP and one more elimination step is required to remove false detec-

21
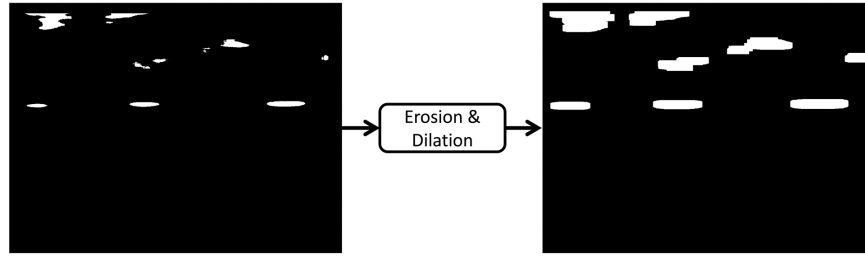
Figure 3.7: Binary erosion and dilation operations on candidate license plate regions.
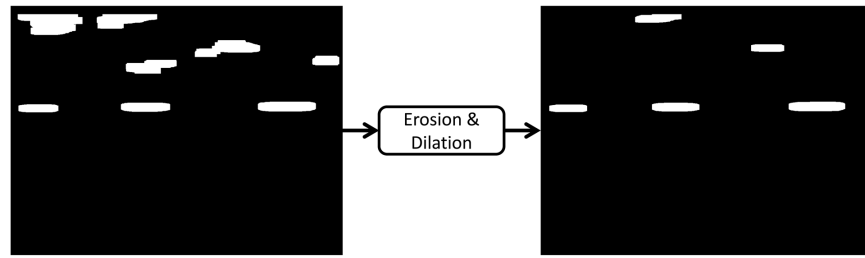


Figure 3.8: Region elimination result.

tions. For this purpose, connected components in $R_b$ are found using a connected component labeling operation. Then, firstly, area of the labeled regions are examined and the ones that have a pixel area less than 200 pixels are removed. 200 pixels of LP area should have approximately a width/height ratio of 6 for an approximate LP region, hence, have an height of approximately 6 pixels. In experimental trials, characters are poorly segmented in LP regions that have less than 10 pixel height and this explains why 200 is selected as minimum area condition. Secondly, width/height ratio of each connected component bounding box is checked to be greater than 2. In turkish regulations, this ratio is approximately equal to 5 and can decrease down to 2 due to the orientation of the imaging system and the car. After elimination operation, regions shown in Figure 3.9 are obtained.

## 3.2 Character Segmentation

Next step after detection of the LP regions is the segmentation of the LP characters. In this step of the algorithm, each candidate LP region is segmented and using CCL and an elimination process, character-like CCs are obtained. While obtaining characters, rule-based elimination

Figure 3.9: License plate region detection results.

process is defined and further consistency check of remaining characters are applied. Then, broken characters are corrected and merged characters are divided. Finally, least squares algorithm [33] is used to fit characters' centers and further missing character extraction and connected character division operations are performed over this line. This 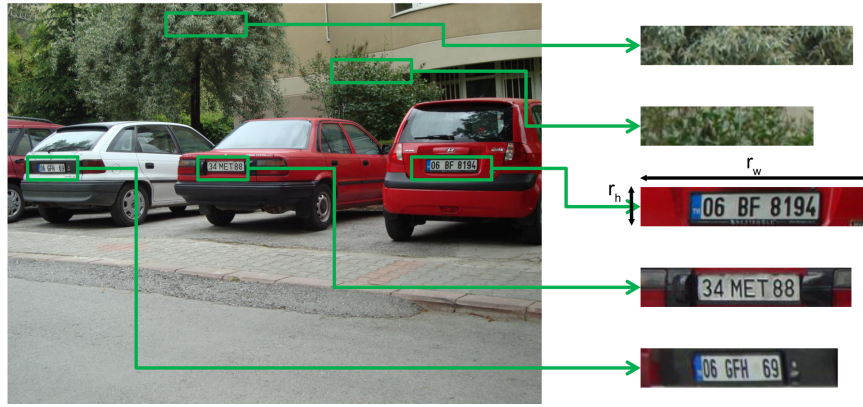part of the chapter requires insertion of each country's regulations over LPs for better character extraction results and some parameters are defined to adapt the algorithm for better performance on different LP types. LP character extraction steps are explained in detail in the following sections. Sample detected LP regions can be seen in Figure 3.9 where $r_w$ and $r_h$ denotes width and height of the detected region.

As seen in Figure 3.9, both LP and non-LP regions are obtained at the end of the LPD step. Proposed algorithms is robust against false LP region inputs and most of them are discarded at the end of the character character elimination of consistency check steps. Steps of CS technique is explained in detail in the following sections.

### 3.2.1 Segmentation of License Plate Region

Most of the LPs in Turkey consist of black numbers and letters on a white background. Background and foreground differentiation is needed to find each character region inside LP region. In this thesis work, intensity values inside the LP region are used to segment characters from the background of the plate. Global thresholding technique is used in most characters segmentation algorithms described at Section 2.4. Global thresholding techniques generally have

the following disadvantages:

- Constant thresholds are not robust against illumination,

- Otsu thresholding in some cases results in false segmentation due to rough detection of plate region and contribution of vehicle's body color in threshold computation,

- Application of a global threshold can result in incorrect segmentation of the plate characters due to nonuniform illumination such as darkening of a plate region as a result of shadow.

To avoid these disadvantages, an adaptive thresholding technique that uses local intensity values for background and foreground differentiation is adopted. For this purpose, firstly, original color image $I^1$ is cropped using the plate region boundary values obtained by the LPD step. Then, a grayscale conversion operation given in Equation 3.10is applied to this color image to be used in binarization operation. Weights of grayscale conversion operation can be seen in Equation 3.10. At the end of this conversion, $I^{gray}_{roi}$ is obtained.

$$I^{gray}(x, y) = 0.2989 * I^1_R(x, y) + 0.5870 * I^1_G(x, y) + 0.1140 * I^1_B(x, y) \qquad (3.10)$$

A threshold value denoted as $\gamma(x, y)$ is computed for each pixel in the $I^{gray}_{roi}$. Average value of the intensities around a square region centered at position $(x, y)$ is found and added to a constant $K$, to compute $\gamma(x, y)$ as seen in the below equations:

$$d = \frac{r_h}{4} \qquad (3.11)$$

$$\gamma(x, y) = \left( \frac{1}{d^2} \sum_{x'=x-\frac{d}{2}}^{x+\frac{d}{2}-1} \sum_{y'=y-\frac{d}{2}}^{y'=y+\frac{d}{2}-1} I^{gray}_{roi}(x', y') \right) + K \qquad (3.12)$$

Selection of parameter K depends on the scenario properties, and discussed in detail in the experimental results. $d$ denotes a single dimension of the square region. Sample results of local thresholding technique can be seen in Figure 3.10 with the values of $K = 10$ and $d = 15$
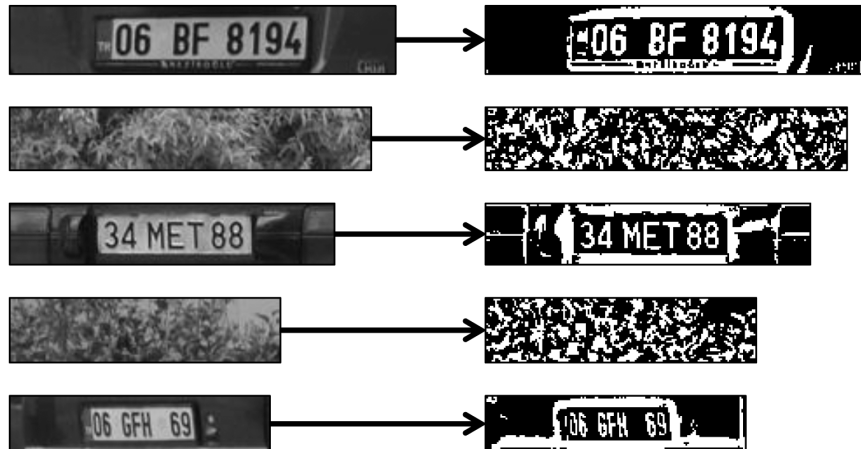
Figure 3.10: Segmentation result of the license plate regions using adaptive thresholding (K=10).

Results obtained at the end of the local thresholding operation are used to segment license plate characters, since most of them are obtained as a single CC after binarization. However, there are still lots of non-character regions, and merged and broken characters. In this thesis work LP characters are obtained using CC analysis. Through CC analysis, improper LP regions are eliminated, merged characters are divided and broken characters are detected. Next section describes the elimination process non-character regions.

### 3.2.2 Elimination of Non-License Plate Characters

At the end of the segmentation operation, CCL is applied to binary plate region and CCs with different sizes and shapes are obtained. Since, most of the character regions obtained at the end of the localization step include LP frame and its neighborhood, some of the obtained CCs belong to non-character regions. These non-character regions include LP frame, stickers on the plate, brand and model logo of the car and dirt. Hence, further elimination process is required to extract characters. To eliminate these non-character regions, features extracted from each CC are used. In this thesis work, the features for a proper LP character to meet are listed below;

- Minimum character area,

- Minimum character height,

25

- Minimum character area to character region area ratio,

- Maximum character region area to plate region area ratio,

- Maximum character region width/height ratio.

An LP character which does not meet any of these conditions is removed and if at the end of the elimination process, more than three characters are remaining, the process is continued. Most of the conditions meet the scale invariance requirements and some conditions are defined for giving a proper output to the recognition phase in terms of minimum size and area that can be recognized. By analyzing each CC with respect to these conditions, elimination of the false detections is achieved.

### 3.2.2.1 Minimum Character Area

Character area is one of the mostly used property to segment characters and most of the algorithms use this property to decide whether a CC is a character or not. Since most algorithms define an exact range for character area, successful results can be obtained only in a limited distance between the camera and the vehicle. In this thesis work, only the minimum value it can have is defined and CCs obtained after segmentation are checked whether they hold this condition or not. Selected minimum number is equal to 30 pixels and most of the undesired responses obtained after segmentation due to noise, dirt or stickers are removed. Characters that have less than 30 pixels are also removed because further difficulties are met at the recognition step for such small characters. Effectiveness of this condition at removing noisy responses in and around plate region can be seen in Figure 3.11.

### 3.2.2.2 Minimum Character Height

Similar to the minimum character area condition, to be robust against distance parameter, instead of a character height range, only its minimum value is defined as a condition and characters are eliminated by checking whether this condition is met or not. Characters that have a height less than 10 pixels are more prone to be a non character structure such as dirt and stickers, broken character segments or vertical LP frame edges. Even if the detected character is a real LP character, most of the recognition algorithms can still not detect such small
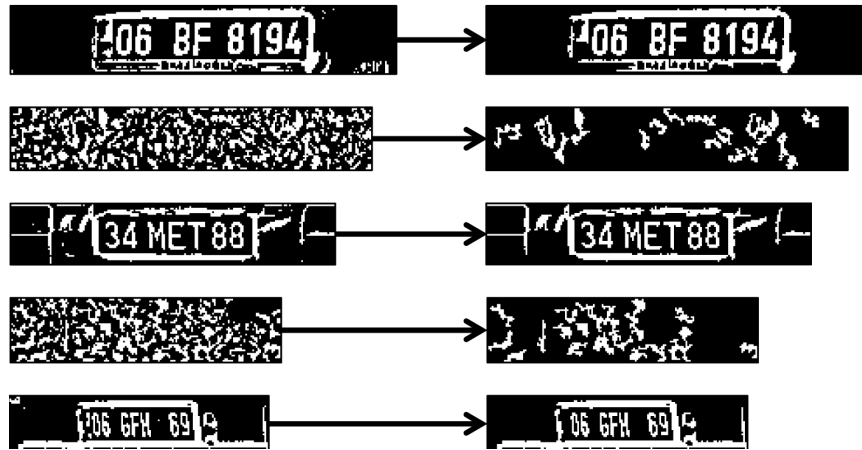
26

Figure 3.11: Elimination with respect to character area condition, CCs that have less than 30 pixels area are removed.

characters. On the sample LP segmentation result given in Figure 3.10, result of elimination process for characters that have less than 10 pixels of height can be seen in Figure 3.12.

### 3.2.2.3 Minimum Character Area to Character Region Area Ratio

Another scale independent character feature, that is defined as a condition, is the ratio between the character area and the area of the bounding box that surrounds the character. This condition is mostly used to eliminate thin lines inside the LP such as license plate frame. Although, next condition defined is used to eliminate LP frames, this condition helps with the removal of the broken LP frame parts. Hence, CCs which have a ratio between the character area (number of white pixels inside character bounding box) and the area of the bounding box of the character less than 0.2 are removed. Results obtained after elimination process with respect to this constraint can be seen in Figure 3.13.

### 3.2.2.4 Maximum Character Region Area to Plate Region Area Ratio

In properly segmented LP regions, character bounding box area is approximately equal to 5% of the licence plate region and this value can not be more than 20%. At the end of the segmentation results, some CCs from the body of the vehicle or frame of the license plate are obtained which do not meet above condition. Therefore, this condition is especially defined to
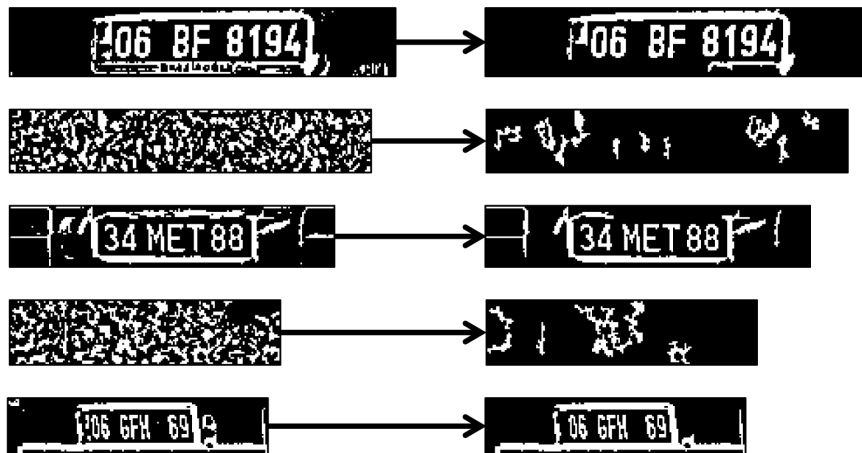
27

Figure 3.12: Elimination with respect to character height condition.
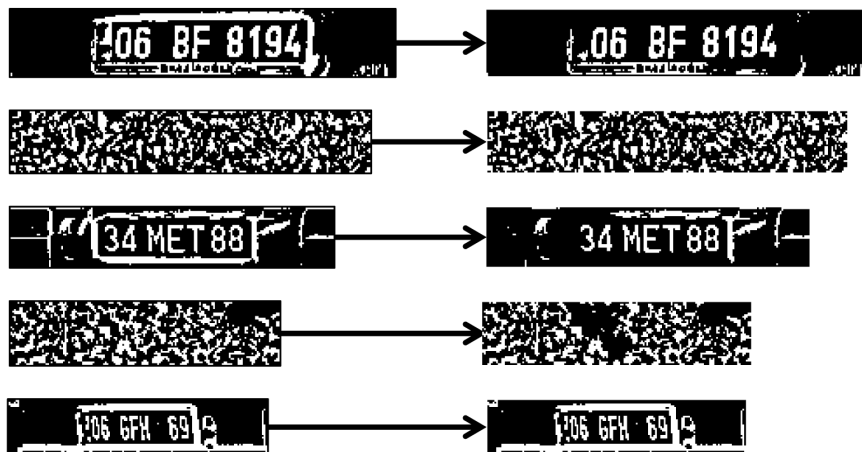


Figure 3.13: Elimination with respect to character area to character bounding box area ratio.
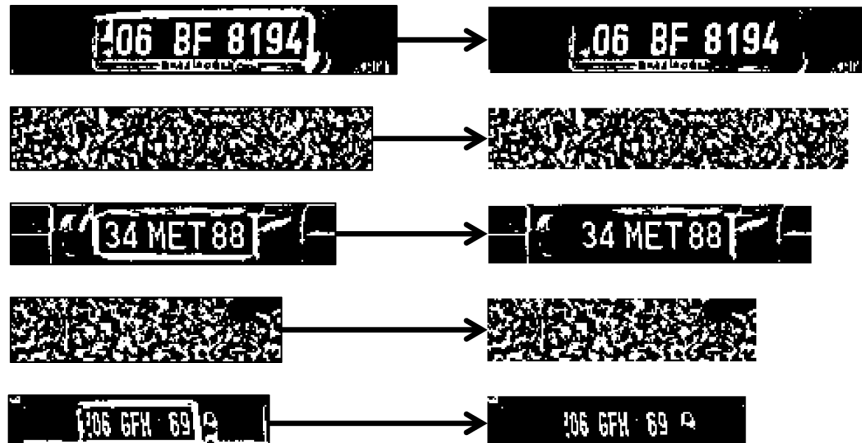
Figure 3.14: Elimination with respect to character bounding box area to plate region area ratio.

eliminate license plate frames because in most of the scenarios, license plate frame is extracted as a unique structure and it has a bounding box area which is approximately equal to licence plate region area. Results obtained after elimination process with respect to this constraint can be seen in Figure 3.14. Results show that licence plate frames are removed completely or partially using this condition. Remaining parts of the frames that can not be removed can be eliminated using one of the other conditions explained in this section.

### 3.2.2.5  Maximum Character Region Width/Height Ratio

In Turkish license plates, each character has a width/height ratio of approximately 0.6. However, due to some effects discussed previously, license plate characters can be merged at the end of the segmentation operation. Turkish license plates consist of groups of numbers and characters and a group can include at most 4 characters. Considering these four characters are all merged and obtained as a single character, it can have a width/height ratio of at most three. So, using this constraint, the license plate character regions are eliminated and result obtained on samples can be seen on Figure 3.15.

### 3.2.3  Consistency Check of License Plate Characters

In the previous section, components inside license plate region are eliminated with respect to their individual CC properties. However, individual character properties is not adequate for
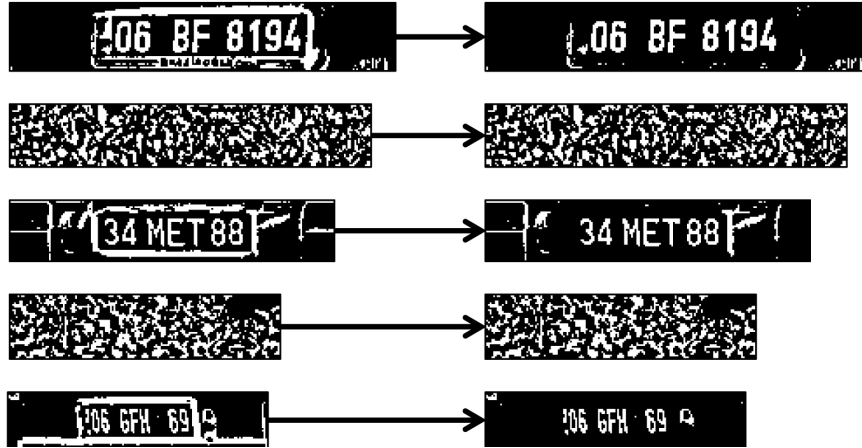
Figure 3.15: Elimination with respect to character width/height ratio.

final character decisions. In this section, one more elimination step is applied to check consistency of inter character properties. LP characters are all lined up and have equal height so consistency of these properties give strong cues about LP existence and characters' positions. Consistency is searched using median value of the corresponding property. If more than three characters remain at the end of the consistency check operations, then process is continued, otherwise, LP region is discarded and CS algorithm does not return a result. In the next two sections, consistency check operations are explained in detail.

### 3.2.3.1 Consistency of Character Heights

Regulations of most country LPs requires that all characters should have equal height. Therefore, after individual blob analysis, inter-character analysis are carried out to remove CCs that have inconsistent height. To eliminate divergent characters with respect to their heights, median value of the character heights which is denoted as $H_{med}$ is calculated. If absolute difference from each characters height to $H_{mean}$ is more than $\lambda_H$, that character is removed from the candidate character list. $H_{mean}$ and $\lambda_H$ are calculated as seen in Equations 3.14 and 3.13.

$$H_{med} = Median(H_{char}(i)), \quad 3 < i \leq p \tag{3.13}$$

$$\lambda_H = \frac{H_{med}}{8} \tag{3.14}$$

30

Figure 3.16: Results of inconsistent height elimination step. $1^{st}$ column: Segmentation output, $2^{nd}$ column: Individual blob analysis output, and $3^{rd}$ column: Height consistency elimination results.

where $H_{char}$ denotes the height of remaining characters and $p$ is equal to remaining number of characters. Result of height consistency elimination can be seen on sample LP regions in Figure 3.16. If remaining character number is less than 3 at the end of this elimination process, region is regarded as a non-LP region and CS operation is invoked for the next region.

### 3.2.3.2 Consistency of Character Placement

All characters in LPs are lined up along a linear line. Therefore, we can define a line as given in Equation 3.15 which passes through the bounding box (BB) centers of the characters. Hence, by checking their divergence in position with respect to this line, inconsistent characters are removed. Divergent characters are found by checking the perpendicular distance from the character BB center to the line. Using a procedure similar to described in the previous section , median value of the character height $H_{med}$ is updated by discarding the values removed in the previous section and if absolute distance from each character's center to line is more than $\lambda_V$, that character is removed from the candidate character list.

$$y = a + b * x \tag{3.15}$$

$$\lambda_V = \frac{H_{med}}{8} \tag{3.16}$$

Coefficient $a$ and $b$ of the line given in Equation 3.15 are calculated using least square fitting [33] as seen below:

31

Figure 3.17: Results of inconsistent character placement elimination step. $1^{st}$ column: Segmentation output, $2^{nd}$ column: Individual blob analysis + character height consistency elimination output, and $3^{rd}$ column: Character placement consistency elimination results.

$$a = \frac{\bar{C}_y \left( \sum_{i=1}^{m} C_x(i)^2 \right) - \bar{C}_x \sum_{i=1}^{m} C_x(i) C_y(i)}{\sum_{i=1}^{m} C_x(i)^2 - m\bar{C}_x^2} \tag{3.17}$$

$$b = \frac{\left( \sum_{i=1}^{m} C_x(i) C_y(i) \right) - m\bar{C}_x \bar{C}_y}{\sum_{i=1}^{m} C_x(i)^2 - m\bar{C}_x^2} \tag{3.18}$$

where $C_x$ and $C_y$ are defined as arrays that contain characters' BB center positions in terms of horizontal and vertical coordinates, and $\bar{C}_x$ and $\bar{C}_y$ are average values of the array entries.

This step of the algorithm includes two iterations. In the first iteration, divergent characters are found but not removed. In the second iteration, line coefficients are computed excluding the previous divergent characters center positions and distance from all characters to this line is computed again. This time, divergent characters are removed and if remaining number of characters is greater than three, process is continued, otherwise, region is regarded as a non-LP region and CS operation is invoked for the next region. Result of character placement consistency elimination can be seen on sample LP regions in Figure 3.17.

### 3.2.4 Merged and Broken Character Correction

After LP characters are eliminated using inter and intra character feature analysis, some characters can be merged with the neighboring ones and can be segmented as a single character. For this reason, further operations are required to separate merged characters and to merge broken characters. For this purpose, median height of the characters is used. If $p$ characters are obtained at the end of the previous section, median height of these characters are defined
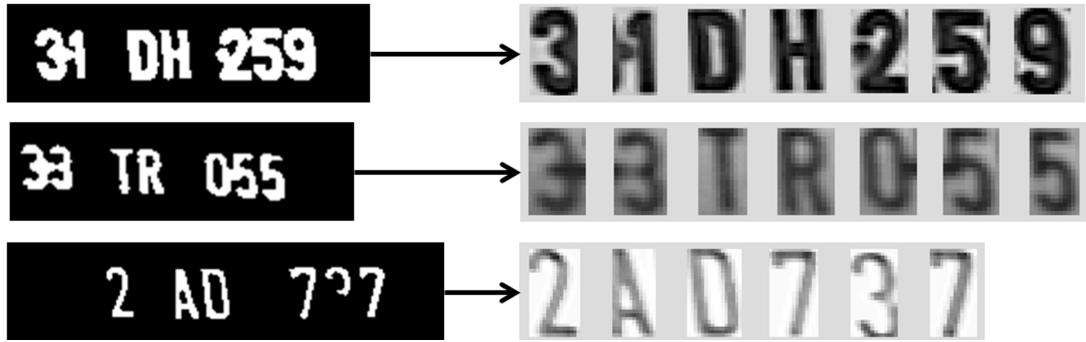
Figure 3.18: Examples of merged and broken character correction.

as $H_{med}$ and median width, $W_{med}$, of the characters are obtained using $H_{med}$ as in equation 3.13 and 3.17.

$$W_{med} = H_{med} * 0.6 \qquad (3.19)$$

If any of the characters has a height less than 0.6 times $H_{med}$, its size is updated to be $H_{med}$ and its vertical center position is re-calculated using its horizontal center position and the Equation 3.15. Nextly, if any of the characters have a width of more than 1.6 times the $W_{med}$,then character is divided by 2 in horizontal direction with two equal parts. Characters which have width/height ratio more than 2.6 are divided into 3 equal parts. Merged and broken character correction operations are shown on sample images in Figure 3.18.

At the end of the merged and broken character correction step, the values of $H_{med}$ and $W_{med}$ are updated and newly defined characters are replaced with the old ones in the LP character list.

### 3.2.5 Missing Character Extraction

Final step of the CS process is extraction of missing characters. Missing characters occur at the end of the character elimination process because of the following reasons:

- Merging with the license plate frame,
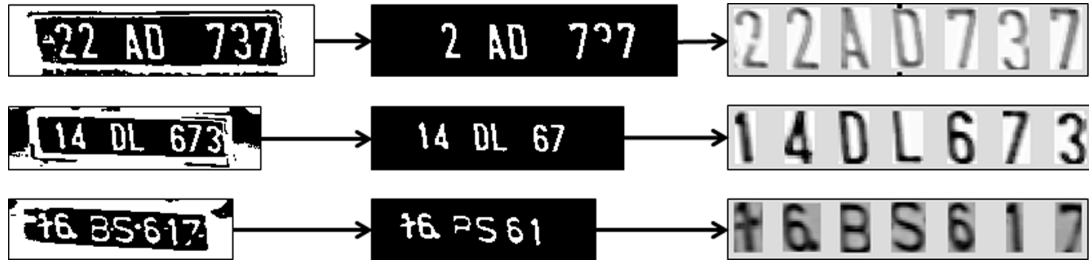
- Broken character parts,

33

Figure 3.19: Missing character extraction at boundaries.

- Insufficient pixel area.

### 3.2.5.1 Missing Character Extraction at the Boundaries

Rightmost and leftmost characters are more prone to be merged with the LP frame. For this reason, left region of the leftmost and right region of the rightmost characters are searched for proper character existence. Median width and height of the characters including the ones obtained at the end of the correction step are used in the search process. Minimum character distance is calculated as 0.1 times the median character height. License plate frames at both sides can appear like the letter $I$ or number 1 and to avoid false detections, search area is enlarged vertically and horizontal projection of the binary image $I_{roi}^{bin}$ in this sub-region is examined to discard frame edges. If the projection result is more than or equal to height threshold value $t_h$, that region is regarded as the LP frame and not added to the character region list. Calculation of $t_h$ can be seen in Equation 3.20

$$t_h = H_{med} * 1.2 \tag{3.20}$$

### 3.2.5.2 Missing Character Extraction in-between Characters

The gaps between the characters which are longer than the median character width are searched for the existence of characters. Lengths of the gaps are defined as the distances between sequent characters. Each gap is divided into $s$ sub-regions and number of sub-regions are calculated using Equation 3.21. $s$ is defined as an integer and fractional results obtained at the end of the operation are rounded towards zeros to the nearest integer value.
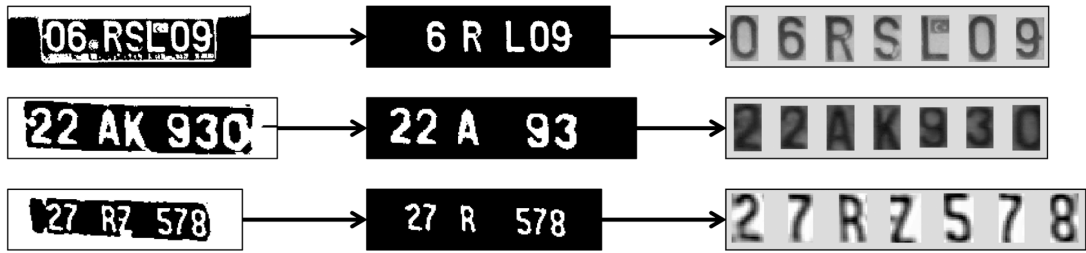
Figure 3.20: Missing character extraction in-between detected characters.

$$s = \frac{dist}{w_{med} + d_{min}} \qquad (3.21)$$

where $dist$ is the distance between the consequent characters, and $d_{min}$ is the minimum distance between the characters. If $s$ is greater than zero, then the following $s$ regions on the segmentation result image $I_{roi}^{bin}$ which have a height $h_{med}$ and width $w_{med}$ next to the character where the gap begins are checked for character existence. This regions are added to the character list if the sum of the white pixels in the examined area is more than $12,5\%$ and less than $80\%$ of the total character bounding box area.

# CHAPTER 4

# EXPERIMENTAL RESULTS

To examine the performance of the proposed algorithm, LP images from three different datasets are used that cover different scenario conditions for a mobile LPR systems. In the following sections of this chapter, datasets are introduced and reasons of selection are explained. Nextly, works done for ground truth extraction is explained and ground truth structure with developed GUI-based software is described. Finally, performance analysis of the system is done for different system variables using recall and precision rates. Performance analysis for LPD and CS are conducted separately since no feedback exists between these two stages.

## 4.1  Image Datasets

In this thesis work, 3 datasets are used for performance analysis of the proposed algorithm. These image datasets are labeled as Dataset-1, Dataset-2 and Dataset-3. Below you can find the detailed descriptions for each dataset.

### 4.1.1  Dataset-1

This dataset includes 64 images taken from parking lots during daytime with a camera of resolution $1280x960$. Distance between the camera and vehicles varies between 3 to 15 meters and most of the images contain more than one LPs. This scenario simulates the parking lot check usage of an LPR system and images are taken from the street. Image backgrounds consist of highly cluttered textures such as tree branches and leaves, or guard rails of the apartment gardens. Mentioned structures in the image create high vertical edge density regions and sometimes cause serious difficulties at localization of LPs. Some of the images have motion

Figure 4.1: Sample images from Dataset-1

blur effect due to vibration of the camera while capturing operation, noise due to insufficient illumination and again blur due to de-focus. LPs with dirt, broken characters, stickers and even without frames (same color with the body) are included in the dataset. Sample images from this dataset can be seen in Figure 4.1.

### 4.1.2 Dataset-2

This dataset includes 100 grayscale images taken from a border gate by "ITU Computer Vision and Image Processing (CVIP) Laboratory" researchers, during day and night conditions from a fixed distance and orientation. Resolution of the images is 768$x$576. This scenario is useful to simulate the performance of the algorithm for systems such as automated payment, and automatic permission systems. Also, this set enables performance evaluation for variable illumination conditions such as day and night conditions, and reflections from sun. In most of the images both imaging system and vehicle are stationary. Hence, there is no blur effect in the images due to motion or defocus. However, trucks with brand and model name or some other informal texts written on the front face make LPD a difficult issue and increases the false alarm rate of the algorithm. Besides, different type of plates located in different positions check the algorithm's robustness against different LP size and orientations. Sample images from this dataset can be seen in Figure 4.2.

Figure 4.2: Sample images from Dataset-2

### 4.1.3 Dataset-3

Although there are plenty of LPR algorithms existent in the literature, rarely the database used to test the performance of the system is shared publicly. Lacking number of common databases to test LPR algorithms' performance create an ambiguity about the applicability and the performance of the algorithms. Anagnostopoulos has collected and grouped many license plate images which are taken under different illumination conditions at different positions and shared the result at http://www.medialab.ntua.gr/research/LPRdatabase.html [3]. Dataset-3 used in this thesis work is the group of images that are shared in this dataset and labeled as "Day (color images-large sample)". Dataset-3 includes 135 images that are taken during daytime from different distances. Resolutions of the images are $640x480$ and $1792x1312$ and sample images taken from the Dataset-3 can be seen in Figure 4.3.

### 4.2 Ground Truth Construction

Performance of LPD and CS algorithms are mostly indicated using detection ratios [3]. But complicated algorithms used for LPR mostly requires many parameter inputs for scene adaptation. In this more than one dimensional parameter space, points which maximize the performance of the algorithm should be found. ROC curves which interpret the sensitivity or true

Figure 4.3: Sample images from Dataset-3

positive rate vs. false positive rate of a detection algorithm in a graphical way, are defined for this purpose [34]. Either for LP and character regions, true positive (TP), false positive (FP), false negative (FN) and *Recall* and *Precision* values are defines as follows:

$$
\begin{aligned}
TP &= \text{\# of true plate (characters) regions found,} \\
FP &= \text{\# of false plate (characters) regions found,} \\
FN &= \text{\# of plate (characters) regions not found,} \\
\text{Recall} &= \frac{TP}{TP + FN}, \\
\text{Precision} &= \frac{TP}{TP + FP}.
\end{aligned}
\tag{4.1}
$$

Although ROC curves are very useful, construction of them is not an easy task for multi-parameter optimization purpose, hence automating the performance calculation steps is required. For this reason, a software with graphical user interface is built and LP and character locations are signed over the image to obtain ground truth data of the image datasets used in this thesis work as seen on Figure 4.4.

Figure 4.4: User interface for construction of the ground truth data.

## 4.3 LPD Results

As an output of the LPD algorithm, firstly, we obtain the coordinates of the rectangular regions in the scene in terms of x and y offsets, and width and height of the regions. Then, actual positions of the LPs are inquired from the ground truth database using the image filename. Next, overlap ratios of the regions with respect to the ground truth area $OR_{gt}$ and detection result area $OR_{dr}$ are computed as seen in Equation 4.2. Regions are illustrated in Figure 4.5.

$$
\begin{aligned}
OR_{gt} &= 100 * \frac{\text{Area}(\text{DR} \cap \text{GT})}{\text{Area}(\text{GT})}, \\
OR_{dr} &= 100 * \frac{\text{Area}(\text{DR} \cap \text{GT})}{\text{Area}(\text{DR})}.
\end{aligned}
\tag{4.2}
$$

$OR_{gt}$ and $OR_{dr}$ are calculated from the results obtained for each image. Detection result is assumed to be true if $OR_{gt}$ is greater than 80% and $OR_{dr}$ is greater than 20% since these ratios allow true CS. Such ratios are chosen due to the fact that, characters can still be extracted correctly by the proposed CS algorithm in this thesis work even if they cover a small region in the detected LP region. If an LP region satisfies the above ratios, then $TP$ value is increased by one. If above overlap ratios are not satisfied, then false detection is assumed and $FP$ is increased by one. At the end of the process, number of correctly located plates are subtracted from the exact number of LPs in the scene, and the result is added to the $FN$.
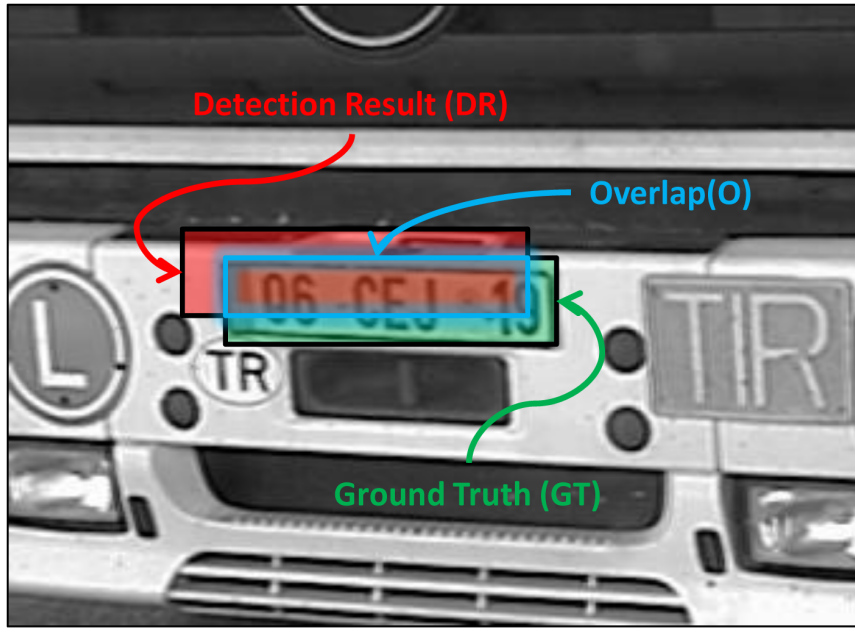
Figure 4.5: Overlapping regions.

One of the most significant parameters that affects the performance of the LPD algorithm is $k$ in Equation 3.8. $k$ determines the variance multiplication coefficient to segment image $R$. So, LPD algorithm performance is examined for varying values of $k$ over the datasets given in Section 4.1. Figure 4.6 gives the recall and precision values for varying values of $k$ between 1.5 to 5. Highest detection rate of 94% is achieved for $k = 3$ for Dataset-1, and 93% detection rates are achieved for Datasets-2 and 3 for $k = 1.5$. As seen on the detection results, increasing values of $k$ decreases the false alarm rate but there is not an exact correlation between recall and precision values, since, in some cases, increasing values of $k$ results in a more accurate LP region detection and results in true LP detection. Previously selected 20% limit ratio for $OR_{dr}$ causes this fact and shows that larger LP region does not indicate a true detection.

Another significant property of the proposed LPD technique is the image pyramid concept for variable size LP detection. Fusion of the multiscale information obtained from vertical edge density images enhances the performance of the detection algorithm and decreases the false detection ratio. Performance of the algorithm for varying values of $k$ on Dataset-1, Dataset-2 and Dataset-3 can be seen on Figure 4.7, Figure 4.8 and Figure 4.9, respectively.

Results show that, multiscale detection concept enhances both the recall and precision ratios

41
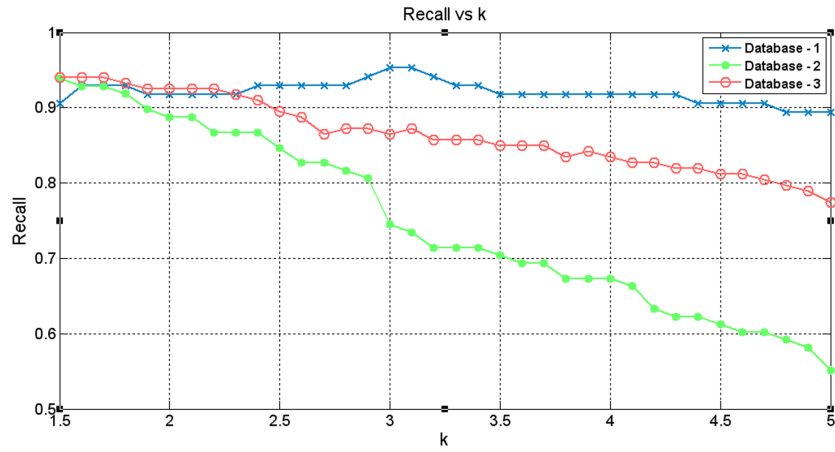
(a)
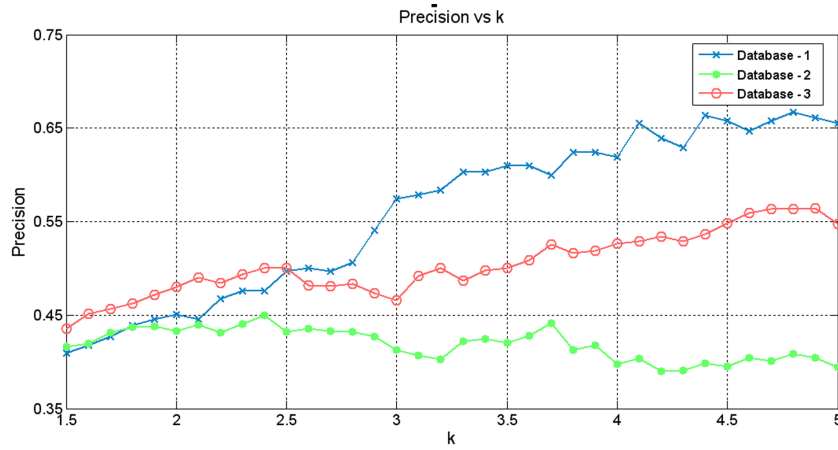


(b)

Figure 4.6: (a) Recall vs k curves, (b) Precision vs k curves.

(a)
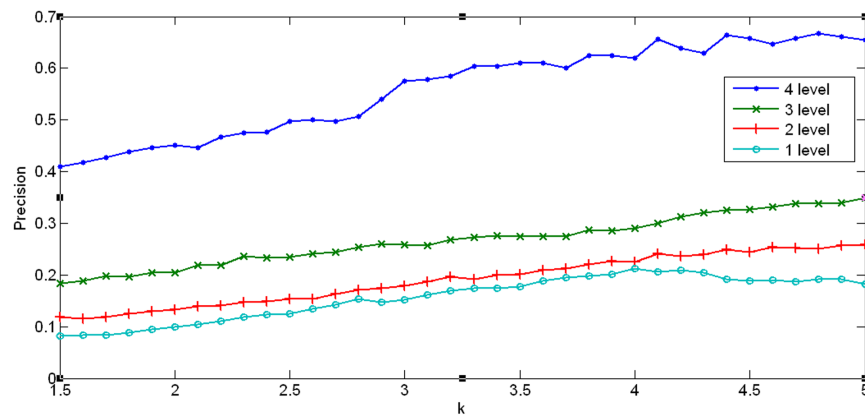


(b)

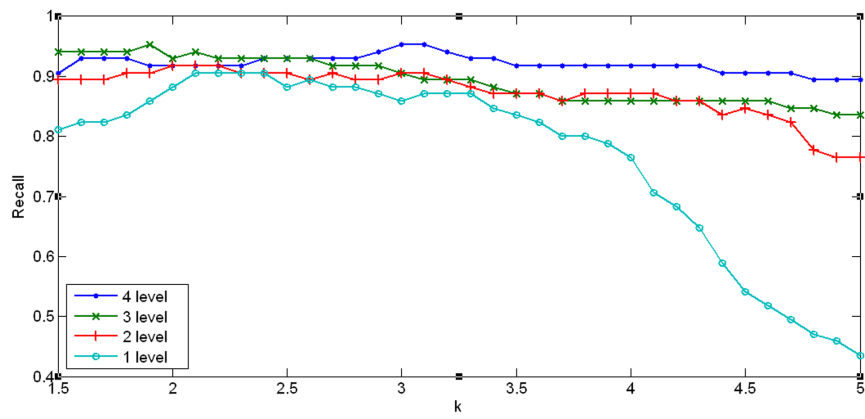Figure 4.7: (a) Recall vs k curves, (b) Precision vs k curves for Dataset-1 for different number of image pyramid levels.

(a)



(b)

Figure 4.8: (a) Recall vs k curves, (b) Precision vs k curves for Dataset-2 for different number of image pyramid levels.
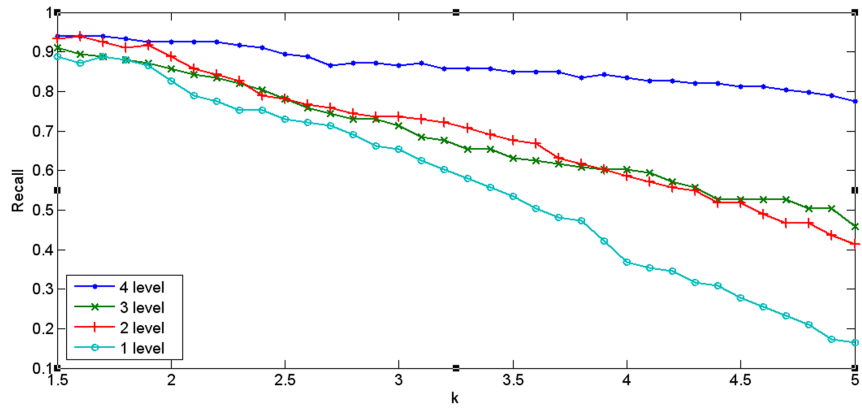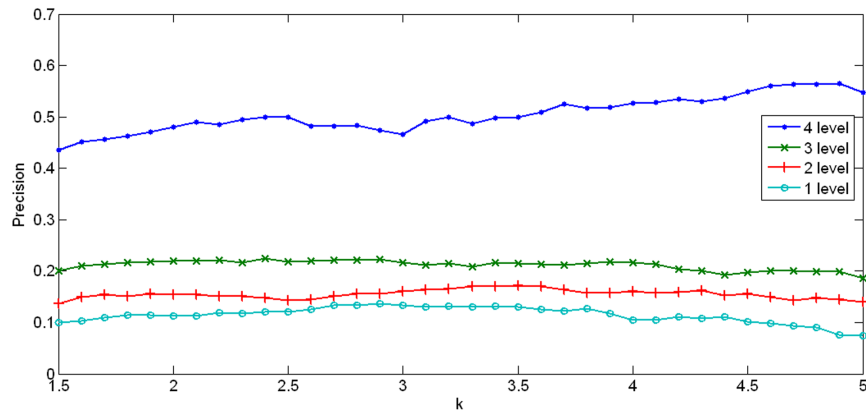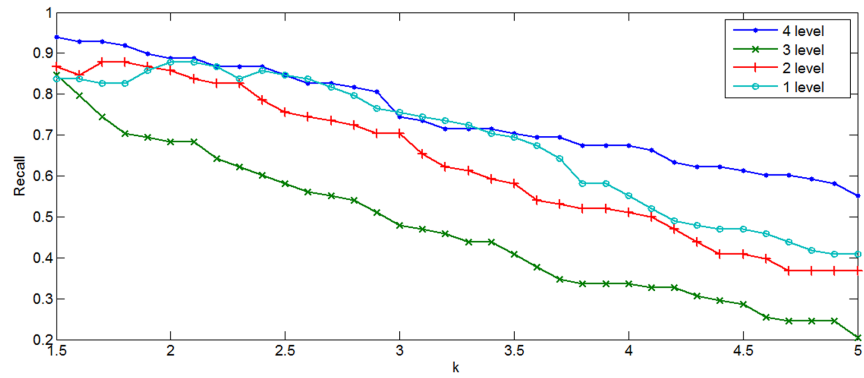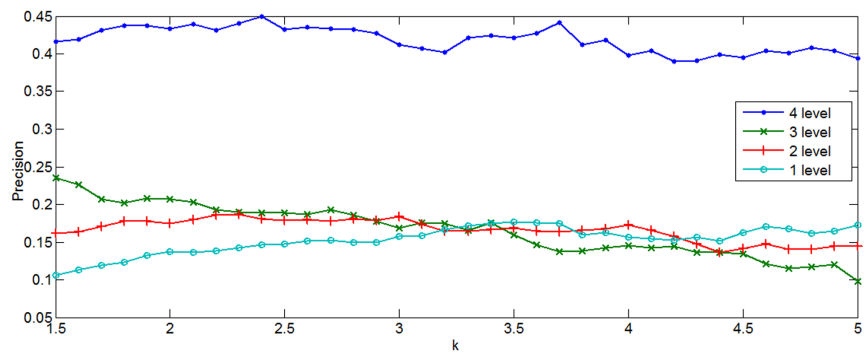
(a)



(b)

Figure 4.9: (a) Recall vs k curves, (b) Precision vs k curves for Dataset-3 for different number of image pyramid levels.

significantly and gives better results over each image datasets that include licence plates with different sizes under different illumination, orientation and background clutter conditions.

## 4.4   CS Results

CS algorithm takes the original image and the BB parameters of the LPs located in the image as input and gives the BB parameters of the detected characters as an output. CS algorithm performance is measured similar to the LPD performance measuring operations. $OR_{gt}$ and $OR_{dr}$ are calculated as seen in Equation 4.2 for each character output of the CS algorithms. If $OR_{gt}$ and $OR_{dr}$ are both greater than 60%, then character is assumed to be located correctly. Selection of characters in the image such as 1 and $I$ can be difficult and user can select a region twice larger than the original character. In these situations, to have a tolerance against erroneous selections, such ratios are defined for correct detection decision. If above overlap ratios are not satisfied, then false detection is assumed and $FP$ is increased by one. At the end of the process, number of correctly segmented are subtracted from the exact number of characters in the plate, and result is added to the $FN$.

The most crucial parameter that affect the result of CS algorithms is $K$ which was previously defined in Equation 3.12 and determines the difference level from the mean local region intensity. For this reason, CS algorithm performance for varying values of $K$ is examined over the datasets and Figure 4.10 gives the recall and precision values for varying values of $K$ between 2 to 20. Highest CS rate of 95% is achieved for $k = 10$ for Dataset-1, 94% CS rate id achieved for Datasets-2 for $K = 8$, and 88% CS rate is achieved for Datasets-3 for $K = 10$. As seen on the CS results, increasing values of $K$ does not necessarily decrease the false CS rate because, 60% limiting factor eliminates characters that are both smaller or greater than the ground truth character region.

## 4.5   Overall Performance

In the previous two sections, performance results of LPD and CS algorithms for varying values of the parameters $k$ and $K$ are given. As seen in Figure 4.6, $k = 2.5$ gives best result for LPD and Figure 4.10 shows that $K = 8$ should be selected for local thresholding. Using these parameter values, successful results obtained on sample images can be seen in Figure 4.11.
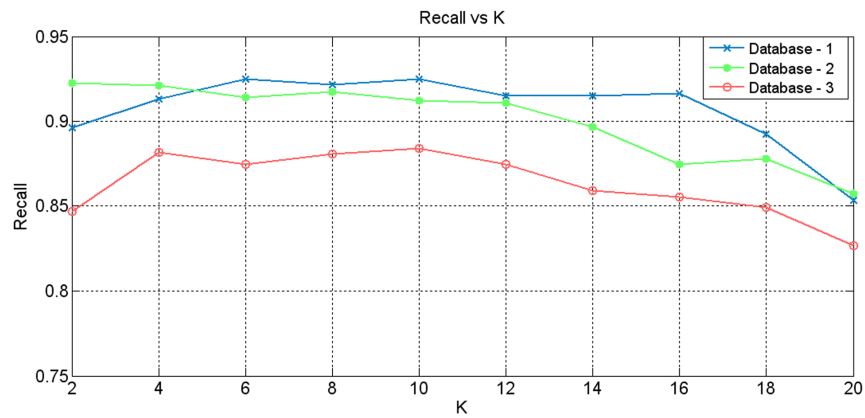
(a)
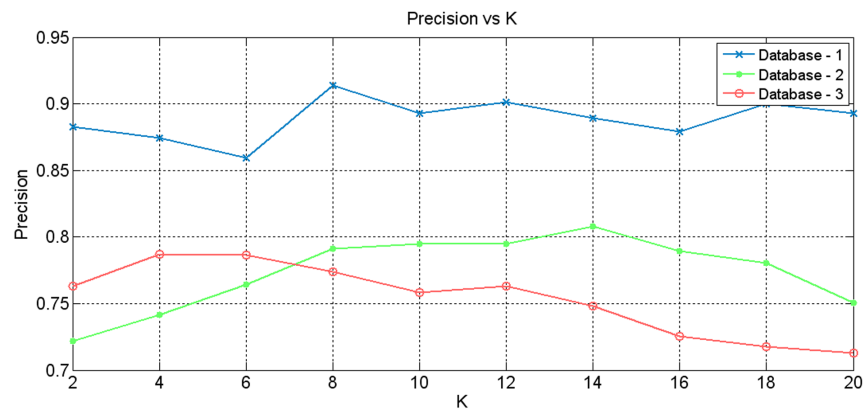


(b)

Figure 4.10: (a) Recall vs K curves, (b) Precision vs K curves.

Figure 4.11: Successful license plate detection and character segmentation results for $k = 2.5$, and $K = 8$.

Figure 4.12: False license plate detection and character segmentation results for $k = 2.5$, and $K = 8$.

Sample images are chosen to cover different distance, orientation and illumination conditions.

Figure 4.12 shows the erroneous results due to false LPD and CS. In low illumination conditions with noise and highly cluttered background, LPD algorithm can not locate the correct LP region in the scene. Also, CS algorithm can not divide merged characters since their total width lies outside the range defined for character division.

´´$1 - precision$, $recall$´´ curves are not given in this work since we can not obtain an exact correlation between the increasing values of recall and decreasing values of precision due to the correct detection constraints.

Overall performance results obtained for LPD and CS steps are compared with the techniques proposed in the work [10] and [13], since common datasets are used in performance evaluation step. Performance evaluation results are written as indicated in the papers and although not exactly mentioned, datasets are assumed to be same. More detailed comparison of the algorithm performances are given et al [3]

Table 4.1: Comparison of the techniques [10] and [13] with the proposed algorithm in terms of LPD and CS performances.

|  | LPD | CS |
|---|---|---|
| Proposed Algorithm | 93%(Dataset-2 and 3) | 94%(Dataset-2) 88%(Dataset-3) |
| Anagnostopoulos et al. [10] | 98%(Dataset-2) | 94.4%(Dataset-2) |
| Kahraman et al. [13] | 96.5%(Dataset-3) | 89.1%(Dataset-3) |

Algorithm is applied using MATLAB 7.9, average processing time for a $1280x960$ image is measured to be 1.3 seconds for an Intel i3-350 core PC with 2.0GB of RAM and Windows 7 operating system. Considering the code is still not optimized, algorithm can be implemented in real-time after some optimization operations.

# CHAPTER 5

# CONCLUSIONS AND FUTURE WORK

## 5.1 Conclusions

In this thesis work, a multi-scale license plate detection algorithm and a rule-based character extraction algorithm were developed and tested. License plate recognition algorithms consist of three steps known as license plate detection, character extraction and character recognition and first two steps are crucial for robustness against distance, orientation and illumination changes. The performance of the algorithms implemented in each step were tested on three different datasets and results are presented individually using recall and precision rates for different algorithm parameter values. An average success rate of 88% for license plate detection and 91% for character segmentation is achieved using optimized parameters for the datasets. Considering variable distance license plates under different orientation and illumination conditions, this success rate is significant for mobile LPR systems.

In the license plate detection step, a multi-scale color edge density based approach was adopted. Firstly, a color image pyramid was constructed using fixed scale factors and color edges for each pyramid image were found. Edge based license plate detection is invariant against illumination changes and better performance was achieved under different day and time conditions. Then edge density over each pyramid image was found by using averaging filter with a similar structure to a license plate structure. Edge density images over the pyramid were fused based on the fact that vertical edges can be extracted from the license plate regions in more than one scales. Regions that have high edge density response at the fused edge density image were segmented by using a threshold value that depends on statistical values of the fused density image. Segmented image is labeled, and then, final elimination

process was applied using connected component analysis. License plate detection algorithm proposed in this thesis achieved a success rate between 85% to 95% for the license plate datasets introduced in the Chapter 4.

In the character segmentation step, license plate regions obtained in the previous step were converted to grayscale images, and were binarized using a local thresholding technique. Therefore, false segmentation results due to shadow, dirt or insufficient illumination can be avoided. Segmented license plate region was labeled and inter and intra connected component features are used to eliminate improper segmentation results. Firstly, individual connected components were removed using minimum character area, minimum character height, minimum character area to character region area ratio and maximum character region area to plate region area ratio criteria. Then, consistency of remaining CCs were checked with respect to character height and character alignment and inconsistent CCs were removed. Finally, missing characters were searched and broken characters were corrected using updated character alignment information. As a result, a success rate between 87% to 93% was achieved for the test datasets.

In Chapter 4, results obtained for each step on the sample datasets which were chosen to cover different scenario conditions were evaluated. For this purpose, ground truths for each dataset were constructed by using a GUI based software and then performance calculation for different parameter values was automated using these ground truths. Recall and precision curves for different parameter values for license plate detection and character segmentation algorithms were given.

Consequently, license plate detection and character segmentation algorithms that are invariant to scale and illumination were described and result of the algorithms were presented in this thesis work. Our main contribution is the multiscale processing of the scene to enable detection of different size LPs, selection of scene dependent threshold values for segmentation, and fusion of inter and intra character properties for CS. Therefore, it is considered that algorithm proposed in this work can be used in a mobile LPR system.

## 5.2 Future Work

Calculation of edge densities depends on a fixed size averaging filter and selection of the width and height of the filter depends on the nominal resolution of the image. Future work of this study should include the usage of a filter bank for better scale and rotation invariance. Also, license plate detection algorithm outputs rough location of the plate in the scene and some characteristic properties of a license plate region should be merged to give the exact coordinates of the license plate boundary as an output.

Character segmentation operation depends on a local thresholding technique and size of the local regions are simply adapted to the height of the license plate region. Multiple segmentation results can be obtained using variable size local regions and best result can be selected for the further operations according to a score assignment process.

The software was implemented on MATLAB 7.9 and average time it takes to locate the license plate and character positions in the scene for an image of resolution $1280x960$ is approximately equal to 1.3 seconds. Future works include the optimization of the algorithms to satisfy real-time implementation requirements.

# REFERENCES

[1] A. D. Joseph, A. R. Beresford, and J. Bacon, "Intelligent Transportation Systems," *IEEE Pervasive Computing*, vol.5, pp. 63-67, 2006.

[2] C. Lum, L. Merola, J. Willis, and B. Cave, "License Plate Recognition Technology (LPR)," *Final Report*, 2010.

[3] C. N. E. Anagnostopoulos, I. E. Anagnostopoulos, I. D.Psoroulas, V. Loumos, and E. Kayafas, "License Plate Recognition From Still Images and Video Sequences: A Survey," *IEEE Transactions on Intelligent Transportation Systems,* vol. 9, no. 3, pp. 377-391, 2008.

[4] "Vehicle registration plates of Turkey," *http://en.wikipedia.org*, last accessed: 15/07/2011.

[5] K. Yamada, T. Tsukada, K. Yamada, K. Kozuka,and S. Yamamoto, "Robust Recognition Methods for Inclined Licence Plates under Various Illumination Conditions Outdoors," *Proc. of IEEE/IEEJ/JSAI International Conference on Intelligent Transportation Systems*, 1999.

[6] P. Comelli, P. Ferregina, M. Granieri, and F. Stabile, "Optical Recognition of Motor Vehicle License Plates," *IEEE Transactions on Vehicular Technology,* vol. 44, no. 4, pp. 790-7991995.

[7] J. Barroso, A. Rafael, E. L. Dagless, and J. B. Cruz, "Number Plate Reading using Computer Vision," *Proc. of the IEEE Internetional Symposium on Industrial Electronics,* pp. 761, 1997.

[8] Y. Lu, "Machine Printed Character Segmentation," *Pattern Recognition,* vol. 3, pp. 213-221, 1995.

[9] R. Zunino, and S. Rovetta, "Vector Quantization for License-Plate Location and Image Coding," *IEEE Transactions on Industrial Electronics,* vol. 47, no. 1, pp. 319-330, 2000.

[10] C. N. E. Anagnostopoulos, I. E. Anagnostopoulos, V. Loumos, and E. Kayafas, "A License Plate-Recognition Algorithm for Intelligent Transportation System Applications," *IEEE Transactions on Intelligent Transportation Systems,* Vol. 7, No. 3, 2006.

[11] J. Sauvola, and M. Pietikainen, "Adaptive document image binarization," *Pattern Recognition,* Vol. 33, No. 2, pp. 225-236, 2000.

[12] R. Al-Hmouz, and S. Challa, "License plate localization based on a probabilistic model," *Machine Vision and Applications,* Vol. 21, No. 3, 2010.

[13] F. Kahraman, B. Kurt and M. Gökmen, "License Plate Character Segmentation Based on the Gabor Transform and Vector Quantization," *ISCIS 2003,* pp. 381-388, 2003.

[14] A. Conci, J. E. R. de Carvalho and T. W. Rauber, "A Complete System for Vehicle Plate Localization, Segmentation and Recognition in Real Life Scene," *IEEE Latin America Transactions,* vol. 7, no. 5, pp. 497-506, 2009.

[15] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. on Sys., Man.,* pp. 62-66, 1979.

[16] S. L. Chang, L. S. Chen, Y. C. Chung, and S. W. Chen, "Automatic License Plate Recognition," *IEEE Transactions on Intelligent Transportation Systems,* vol. 5,no. 1, pp. 42-53, 2004.

[17] J. M. Guo, and Y. F. Liu, "License Plate Localization and Character Segmentation With Feedback Self-Learning and Hybrid Binarization Techniques," *IEEE Transactions on Vehicular Technology,* vol. 57, no. 3, pp. 42-53, 2008.

[18] K. M. Hung, H. L. Chuang, and C. T. Hsieh, "License Plate detection Based on Expanded Haar Wavelet Transform," *IEEE Fourth International Conference on Fuzzy Systems and Knowledge Discovery,* pp. 415-419, 2007.

[19] H. Caner, H. S. Gecim, and A. Z. Alkar, "Efficient Embedded Neural-Network-Based License Plate Recognition System," *IEEE Transactions on Vehicular Technology,* vol. 57, no. 5, pp. 2675-2683, 2008.

[20] X. Fan, and G. Fan, "Graphical Models for Joint Segmentation and Recognition of License Plate Characters," *IEEE Signal Processing Letters,* vol. 16, vo. 1, pp. 10-13, 2009.

[21] H. Mahini, S. Kasaei, and F. Dorri, "An Efficient Features-Based License Plate Localization Method," *18th International Conference on Pattern Recognition (ICPR),* pp. 841-844, 2006.

[22] Y. Li, and M. Wang, "Novel and fast algorithms of licence plate locations and extractions," *IEEE International Conference on Information and Automation,* pp. 2462-2465, 2010.

[23] C. Yu, M. Xie and J. Qi, "A Novel System Design of License Plate Recognition," *IEEE Int. Symposium on Computational Inteligence and Deisgn,* pp. 114-117, 2008.

[24] C. A. Rahman, W. Badawy and A. Radmanesh, "A Real Time Vehicle's License Plate Recognition System," *Proc. of IEEE Conference on Advanced Video and Signal Based Surveillance,* 2003.

[25] Z. X. Chen, C. Y. Liu, F. L. Chang and G. Y. Wang, "Automatic License-Plate Location and Recognition Based on Feature Salience," *IEEE Transactions on Vehicular Technology,* vol. 58, no. 7, pp. 3781-3785, 2009.

[26] J. K. Tyan, C. Neubauer, and L. Goganovic, "A character segmentation algorithm for recognition of vehicle license plate," *SPIE Conference on Mobile Robots XIV,* 1999.

[27] X. Shi, W. Zhao, and Y. Shen, "Automatic License Plate Recognition System Based on Color Image Processing," *Computational Science and Its Applications,* 2005.

[28] L. G. C. Hamey, and C. Priest, "Automatic Number Plate Recognition for Australian Conditions," *IEEE Proc. of the Digital Imaging Computing: Techniques and Applications (DICTA),* 2005.

[29] Y. Nakagawa, and A. Rosenfeld, "Some experiments on variable thresholding," *Pattern Recognition,* vol. 11, no. 3, pp. 191-204, 1979.

[30] I. Giannoukos, C. N. Anagnostopoulos, V. Loumos, and E. Kayafas, "Operator context scanning to support high segmentation rates for real time license plate recognition," *Pattern Recognition,* vol. 43, pp. 3866-3878, 2010.

[31] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision,* vol. 60, no. 2, pp. 91-110, 2004.

[32] J. Y. Bouguet, "Pyramidal Implementation of the Lucas Kanade Feature Tracker: Description of the algorithm," *Intel Corp. Microprocessor Research Labs,* 2000.

[33] J. F. Kenney, and E. S. Keeping, "Linear Regression, Simple Correlation, and Contingency," *Mathematics of Statistics,* 2nd ed. Princeton, pp. 199-237, 1951.

[34] Tom Fawcett, "ROC Graphs, Notes and Practical Considerations for Researchers," *HP Laboratories,* vol. 31, 2004.