

İSTANBUL TECHNICAL UNIVERSITY ★ INSTITUTE OF SCIENCE AND TECHNOLOGY

CAMERA BASED VEHICLE DETECTION AND TRACKING

**M.Sc. Thesis by
Burcu AYTEKİN**

Department : Mechatronics Engineering

Programme : Mechatronics Engineering

JANUARY 2009

CAMERA BASED VEHICLE DETECTION AND TRACKING

**M.Sc. Thesis by
Burcu AYTEKİN
(518051006)**

**Date of submission : 25 December 2008
Date of defence examination: 20 January 2009**

**Supervisor (Chairman) : Assis. Prof. Dr. Erdinç ALTUĞ (ITU)
Members of the Examining Committee : Prof. Dr. Levent GÜVENÇ (ITU)
Assis. Prof. Dr. Tankut ACARMAN
(GSU)**

JANUARY 2009

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYARLI GÖRÜ TEMELLİ ARAÇ BELİRLEME VE TAKİBİ

YÜKSEK LİSANS TEZİ
Burcu AYTEKİN
(518051006)

Tezin Enstitüye Verildiği Tarih : 25 Aralık 2008

Tezin Savunulduğu Tarih : 20 Ocak 2009

Tez Danışmanı : Yrd. Doç. Dr. Erdiñ ALTUĞ (İTÜ)
Diğer Jüri Üyeleri : Prof. Dr. Levent GÜVENÇ (İTÜ)
Yrd. Doç. Dr. Tankut ACARMAN (GSU)

OCAK 2009

FOREWORD

I would like to thank my advisor, Assis. Prof. Dr. Erdiñ ALTUĞ, for his guidance and support during my M.Sc. studies. This work has been supported by ITU Mekar Mechatronics Research Labs and the Automotive Control and Mechatronics Research Center directed by Prof. Dr. Levent GÜVENÇ. I also would like to thank Prof. Dr. Levent GÜVENÇ for giving me the opportunity to work with him and to help from his broad vision.

This work is the end of a period for me. With every end, feeling of excitement for a new beginning and a little fear or maybe a lot due to unknown future must be inevitable. However, there is only one thing I know quite well is that I have a family that is right behind me wheresoever I will step and is that the essence infused into me by them will always make me be a good person. I would like to thank my mother and father, the gift of life to me, Asiye and Mustafa AYTEKİN; my elder brother, Dr. Murat AYTEKİN and my one and only sister, Burçak AYTEKİN. They are the other side of my soul.

January 2009

Burcu AYTEKİN

Mechanical Engineer

TABLE OF CONTENTS

	<u>Page</u>
ABBREVIATIONS	vi
LIST OF FIGURES	viii
LIST OF SYMBOLS	x
SUMMARY	xii
ÖZET.....	xiv
1. INTRODUCTION.....	1
1.1 Purpose of the Thesis	2
1.2 Background of Vision-Based Intelligent Vehicle Research.....	4
1.3 Thesis Structure	6
2. VEHICLE DETECTION	7
2.1 Approaches Proposed in Literature	7
2.1.1 Knowledge-based methods	7
2.1.1.1 Symmetry	7
2.1.1.2 Color.....	8
2.1.1.3 Vertical/ horizontal edges	8
2.1.1.4 Texture	8
2.1.1.5 Vehicle lights	8
2.1.2 Stereo-based Methods	9
2.1.3 Motion-based Methods	9
2.2 Critique of Vehicle Detection Approaches	9
2.2.1 The first step: hypothesis generation.....	10
2.2.2 The second step: hypothesis verification	11
2.3 Objective	11
2.4 The Implemented Methods For Vehicle Detection Within the Thesis.....	12
2.4.1 Road area finding	13
2.4.1.1 Hough transform	13
2.4.1.2 Lane detection	15
2.4.2 Vehicle detection.....	20
2.4.2.1 Hypothesis generation – shadow detection	20
2.4.2.2 Hypothesis verification – vertical edges detection.....	25
3. VEHICLE TRACKING	27
3.1 Literature Overview of Object Tracking	27
3.2 Problem Conditions.....	29

	<u>Page</u>
3.3 Objective	31
3.4 The Theory of the Kalman Filter.....	32
3.4.1 The process to be estimated	32
3.4.2 The computational origins of the filter	34
3.4.3 The probabilistic origins of the filter	35
3.4.4 The summary of the discrete kalman filter algorithm.....	35
3.5 Dynamical System Formulation of the Implemented Vehicle Tracking...	36
3.5.1 The initialization of the kalman filter.....	40
3.6 The Implemented Algorithm	43
3.6.1 To update the filter: horizontal and vertical edges detection	44
4. CONCLUSION AND RECOMMENDATIONS	47
REFERENCES.....	53
APPENDICES	57
CURRICULUM VITA	65

ABBREVIATIONS

ACC	: Adaptive Cruise Control
DAS	: Driving Assistance Systems
ITS	: Intelligent Transportation Systems
ROI	: Region-of-interest
fps	: Frames per second

LIST OF FIGURES

	<u>Page</u>
Figure 1.1 : Schematic overview of the objective of the thesis.	6
Figure 2.1 : Basler A601FC color camera.	12
Figure 2.2 : The theory of the Hough transform.	14
Figure 2.3 : (a) Detected lines in the left half (320 x 240) part of the image..	16
(b) Detected lines in the right half part of the image.	16
Figure 2.4 : Two longitudinal edges that can be described as the transition from darker gray values to brighter ones or the transition from brighter gray values to darker ones.	17
Figure 2.5 : (a) The original half image;.....	17
(b) The filtered half image by the mask [-1 0 1]......	17
Figure 2.6 : Detected lines on the same lane line.....	18
Figure 2.7 : The output of the algorithm for the left half part of the image: Left-most line and Left line.....	18
Figure 2.8 : (a) Road area identification; (b) Besides scanning each lane independently, it is also possible to group the lane lines that can be detected in the current frame.	19
Figure 2.9 : Detected shadows.	21
Figure 2.10 : Successive shadow edges relating to the same vehicle.	22
Figure 2.11 : (a) The edges that could not be eliminated in the combining process.....	24
(b) An example of false hypotheses can also be seen at close range.....	24
Figure 2.12 : Defining the region-of-interest (ROI).	25
Figure 3.1 : (a) Tracking the object without position prediction might be successful; (b) Tracking without position prediction will fail....	30
Figure 3.2 : Signal flow representation of a linear, discrete-time dynamical system.	33
Figure 3.3 : A complete description of the operation of the Kalman filter.....	37
Figure 3.4 : The description of the bounding box and the control points.....	38
Figure 3.5 : Assumed probability distribution of the acceleration u	42
Figure 4.1 : The summary of the detection algorithm.....	48
Figure 4.2 : The flow chart of the implemented algorithms.	50
Figure 4.3 : Low sun from the side makes that vehicles cast long shadows...	51
Figure A.1 : Detection and tracking of mid-range and distant vehicles.	59
Figure A.1 (contd.) : Detection and tracking of mid-range and distant vehicles.....	60
Figure A.2 : Detection and tracking of the vehicles at close range.	61
Figure A.2 (contd.) : Detection and tracking of the vehicles at close range.	62
Figure A.3 : Detection and tracking of the vehicle in the situation where an overpass occurs shadow areas on the road.	63

Figure A.3 (contd.) : Detection and tracking of the vehicle in the situation
where an overpass occurs shadow areas on the road. 64

LIST OF SYMBOLS

\mathbf{x}_k	: The state vector.
\mathbf{R}_k	: The measurement noise covariance matrix.
\mathbf{Q}_{k-1}	: The process noise covariance matrix.
\mathbf{P}_k	: The error covariance matrix.
\mathbf{K}_k	: The Kalman filter gain.
\mathbf{H}_k	: The measurement matrix.
Φ_{k-1}	: The transition matrix.
\mathbf{w}_{k-1}	: The uncertainty in the process.
μ_k	: The uncertainty in the measurement.

CAMERA-BASED VEHICLE DETECTION AND TRACKING

SUMMARY

In recent years, developing on-board driver assistance systems (DAS) aiming to alert drivers about driving environments, and possible collision with other vehicles is becoming active research area among automotive industries, suppliers and universities. In these systems, robust and reliable vehicle detection and tracking are the basic steps. These basic steps could be accomplished by one or multiple sensors such as optical and radar sensors, etc.

Vision-based vehicle detection and tracking for intelligent driver assistance has received considerable attention over the last 15 years. There are at least three reasons for this attention:

1. The startling losses both in human lives and finance caused by accident severity,
2. The growth in technologies within the last 30 years of computer vision research,
3. The exponential growth in processor speeds that makes possible running computation-intensive video-processing algorithms.

With the ultimate goal of building autonomous vehicles for reducing accidents caused by the main threats of driver inattention, various projects have been launched worldwide. Monocular vision based vehicle detection and tracking systems are particularly interesting for their low cost and the high-fidelity information they provide about the driving environment.

The work presented within this master thesis purposed to study computer vision algorithms for automatic vehicle detection and tracking in monochrome images captured by mono camera. The work has mainly been focused on detecting and tracking vehicles viewed from behind in daylight conditions.

The method presented within the thesis includes road area finding which has been implemented by a lane detection algorithm to avoid false detection of vehicles caused by the distraction of background objects. Assuming that lanes are successfully detected, vehicle presence inside the road area is hypothesized by using “shadow” as a cue. Hypothesized vehicle locations are verified using “vertical edges” and “shadow” is also used for verification. After extracting vehicles, the algorithm effectively track them during successive image frames in a long image sequence using a Kalman filter based tracking algorithm.

The 2D-vehicle velocity provided by the algorithms implemented within the thesis will be used to estimate parameters of the (3D) real-world motion of vehicles relative to the host vehicle with the aim of forward collision warning as a future work.

BİLGİSAYARLI GÖRÜ TEMELLİ ARAÇ BELİRLEME VE TAKİBİ

ÖZET

Sürücüyü, sürüş koşulları ve çarpışma olasılığına karşı uyararak araç içi sürücü yardım sistemlerinin geliştirilmesi; otomotiv endüstrisi, yan sanayi ve üniversiteler arasında giderek yaygınlaşan bir uygulama alanı bulmaktadır. Bu sistemlerin temelini, dayanıklı ve güvenilir bir şekilde gerçekleştirilmesi amaçlanan araç algılama ve takibi çalışmaları oluşturmaktadır. Araç algılama ve takibi, optik ya da radar algılayıcılar gibi bir ya da çoklu algılayıcılar üzerine temellendirilmiş sistemler ile gerçekleştirilmektedir.

Sürücü yardım sistemlerinin geliştirilmesi sürecinde; görü-tabanlı araç algılama ve takibi üzerine, son 15 yıldır, ciddi bir eğilim söz konusudur. Görü-tabanlı araç algılama ve takibi çalışmalarına olan eğilimin başlıca üç sebebi;

1. Giderek artan trafik kazalarının sebep olduğu hayati kayıpların ve devlet ekonomisine getirdiği zararın endişe verici boyutlara ulaşması,
2. Bilgisayarla görü araştırmalarının son 30 yılı içerisinde teknolojiye meydana gelen büyüme,
3. İşlemci hızının giderek artması sonucu, işlem hızının öncelik taşıdığı video-işleme algoritmalarının çalışmasının mümkün kılınmasıdır.

Sürücünün dikkatsizliği, yorgunluğu gibi sürücü kaynaklı etmenlerin sebep olduğu kazaları azaltmak amacıyla nihai amacı sürücüdenden bağımsız – otonom araçlar gerçekleştirmek olan pek çok proje, tüm dünyada, uygulama alanı bulmuştur. Tekgözlü imgeleme olarak tabir edilen tek kamera ile gerçekleştirilen görü-tabanlı araç algılama ve takibi, düşük maliyeti ve yüksek kalitede veri sağlaması sebebiyle bilhassa ilgi görmektedir.

Bu dokümanda bahsi geçen yüksek lisans tezi kapsamında sunulan çalışmada, tek kamera aracılığıyla toplanan gri seviye görüntüler içerisinde araç algılama ve takibi amaçlanmıştır. Sunulan çalışmada, temel olarak, araçların arka görünüşleri algılanmaya ve sonrasında takip edilmeye çalışılmıştır. İşlenen görüntüler, gün içi saatlerine dairdir. Geliştirilen algoritmalar, gece görüntüleri için tasarlanmamıştır.

Tez kapsamında sunulan uygulamada; görüntünün arka planında yer alan araç dışı nesnelerin, algılama sürecinde hatalara sebep olmaması için doğrudan kameranın önünde gözlemlenen yol yüzeyi, bir şerit algılama algoritması aracılığıyla, belirlenmektedir. Şeritlerin güvenilir bir şekilde algılandıkları varsayılarak, araçların altında oluşan gölgelerin ayırt edici özellik olarak kullanımıyla, belirlenen yol yüzeyinde, muhtemel araçların konumları kestirilir.

Kestirilen araç konumlarının doğruluğu, dikey kenarlar ve yine araç altında oluşan gölgenin ayırt edici özellik olarak kullanımıyla tetkik edilir. Araç algılama sürecinin tamamlanması sonrasında, algılanan araçların takibi (ardışık görüntüler boyunca araçların konum değişikliklerinin tayini), Kalman filtresi temelli bir algoritma aracılığıyla, ardışık görüntüler boyunca gerçekleştirilir.

Tez kapsamında uygulanan algoritmalar, iki boyutlu görüntü düzleminde, araç hızının belirlenmesini sağlamaktadır. Nihai amaç; yoldaki diğer araçların, kameranın bulunduğu araca göre üç boyutlu bağıl mesafe ve hızlarının tayinidir. Üç boyutlu bağıl hız ve mesafe tayini, araçların yer koordinat sistemindeki gerçek hareketlerini belirlemektedir. Dolayısıyla, tehdit oluşturabilecek araçlara karşı sürücülerin uyarılmasını sağlayacak sistemlerin geliştirilmesi mümkün olabilecektir.

1. INTRODUCTION

Since the first vehicle which moved by its own power was build in Paris in the 18th century, technological and social developments led to today's dominant place of vehicles, trucks and busses in modern society. Since then, we have constantly been confronted with negative consequences of vehicles. By means of rules, infrastructure, road and car design these negative consequences were tried to be controlled. In attempt to reduce the numbers of vehicles on the road, vehicle-related taxes were introduced and increased and alternative means of transportation were promoted.

Nowadays every minute, on average, at least one person dies in a vehicle accident and at least 10 million people are injured each year, two or three million of them seriously. Losses in finance caused by vehicle accidents are also very challenging. This situation requires new solutions. Intelligent Transportation Systems (ITS) provides a modern, more drastic attempt to vehicle related problems we are facing today.

By means of (partially) automating driver tasks and by means of communication (vehicle-to-vehicle as well as roadside-to-vehicle) ITS aims to:

1. Increase the capacity of highways: higher speed, closer spacing, less human errors
2. Improve safety: warning systems, intelligent speed adaptation, less human errors
3. Reduce fuel consumption: optimal speed, optimal acceleration, reduced drag force (platooning), cost reduction
4. Reduce pollution: as a direct consequence of first and third item.

Researches within ITS can be classified as “road-side intelligence” and “in-car intelligence”. Road-side intelligence systems provide more global information about driving environment or destination such as systems that report about traffic flow, accidents and highway maintenance, dynamic navigation systems or systems that provide parking space information.

In-car intelligence systems consider the environment immediately around the vehicle. These systems can be ordered according to the level of autonomy of the vehicle. First the “advisory” and “warning” systems can be identified within this class of intelligence systems. Examples are systems for blind spot monitoring, collision warning, pedestrian warning, lane-departure warning, traffic sign recognition and driver monitoring. Next “driver-assistance systems” can also be identified within this class of intelligence systems. Typical example for this kind of systems is adaptive cruise control.

Today’s implementations mainly concern precrash sensing. Several national and international projects have been realized over the past several years to investigate new technologies for improving safety. Developing on-board driver assistance systems aiming to alert drivers about driving environment and possible collision has attracted a lot of attention and is becoming an active research area among automotive industries, suppliers and universities.

Vehicle detection and tracking is the first step of these systems and this thesis addresses a fundamental aspect for in-car intelligence systems.

1.1 Purpose of the Thesis

Determining the position of other vehicles on the road and their motion relative to your own vehicle is an essential task to develop driver assistance systems like adaptive cruise control (ACC) and platooning. The most important vehicle a driver should pay attention to is the preceding one, to which a security distance should be kept. For this reason, an autonomous system capable of understanding what the position of the preceding vehicle is would be very useful to increase driver’s safety.

The problem can be addressed by using “direct range” sensors which include millimeter wave radars, laser radars (lidar) and stereo imaging as many researchers have done. Although radar and laser sensors measure distance to obstacles with a high degree of accuracy, obtaining their lateral positions required for estimating the possibilities of collision is difficult. Since vision is the most important sense used by humans for driving and optical sensors are passive and cheaper, another option is applying computer vision techniques. On the other hand, it is expected that optical sensors, such as normal cameras, should estimate both lateral positions of obstacles and their shape. As opposed to a stereo imaging design that is including the cost of the additional camera and processing power, a monocular visual processing system is easier to mass produce and costs less as an end product.

No 3D information about the position of other vehicles is directly available using a monocular camera. But studies to investigate the possibility of performing distance control, to an sufficient accuracy level, by a monocular imaging device (a single camera) using the laws of perspective and putting some constraints such as assuming a flat road have been realized.

To estimate parameters of the (3D) real-world motion of other vehicles on the road relative to your own vehicle using vision requires providing 2D-image velocity. The vehicle displacements in the image plane between successive image frames must be computed. In literature, this problem is generally addressed in two steps: vehicle detection and vehicle tracking. These steps are the basis of estimating positions of vehicles present in the scene and their relative motion.

This thesis focuses on vision-based on-road vehicle detection and tracking in monochrome (i.e., grayscale) images from a mono camera mounted on the rear-view mirror of the vehicle. All algorithms are implemented in MATLAB and tested on data supplied by the experimental vehicle used for multi-modal data collection and processing within the Drive Safe Project in which İstanbul Technical University Automotive Control and Mechatronics Research Center is a participant.

1.2 Background of Vision-Based Intelligent Vehicle Research

A large number of government institutions, automotive industries and suppliers, and R&D companies have launched various projects worldwide. These attempts have produced several prototypes and solutions, based on rather different approaches [1-4]. Looking at research on intelligent vehicles worldwide, Europe pioneers the research, followed by Japan and United States.

In Europe, The PROMETHEUS project (Program for European Traffic with Highest Efficiency and Unprecedented Safety) started this exploration in 1986. A large number of vehicle manufactures and research institutes from 19 European countries were involved. Several prototype vehicles and systems were designed as a result of the project. In 1987, the UBM (Universitaet der Bundeswehr Munich) experimental vehicle VaMoRs demonstrated fully autonomous longitudinal and lateral vehicle guidance by computer vision on a 20 km free section of highway at speed up to 96 km/h. Vision was utilized to provide input for both lateral and longitudinal control. That was the first milestone.

Within the PROMETHEUS project, the Institute of Measurement Science has developed real-time vision technology that may be used for a driver support system [5]. Freeways were chosen as the principal domain for testing and demonstrating the visual recognition of objects that are relevant for the understanding of traffic situations. The reason for choosing freeways is that the complexity of the traffic situations and the variety of objects are much lower on freeways than on other roads. Long range autonomous driving has been realized by the VaMP of UBM in 1995. The trip was more than 1,600 km [6]. Another experimental vehicle, mobile laboratory (MOB-LAB) was also part of the PROMETHEUS project [7]. It was equipped with four cameras, several computers, monitors and a control-panel to give a visual feedback and warnings to the driver. One of the most important subsystems in the MOB-LAB was the Generic Obstacle and Lane Detection (GOLD) system. The GOLD system addressed both lane and obstacle detection utilizing a stereo rig. The GOLD system has been ported on ARGO, a Lancia Thema passenger car with automatic steering capabilities [8].

In Japan, MITI, Nissan and Fujitsu pioneered the research by the project “Personal Vehicle System” [9]. In 1996, the Advanced Cruise-Assist Highway System Research Association (AHSRA) was established among automobile industries and many research centers [1]. The Japanese Smartway concept car will implement some driver assistance features, such as, lane keeping, intersection collision avoidance, and pedestrian detection. A model deployment project was planned to be operational by 2003 and national deployment in 2015 [2].

In the United States, many initiatives have been launched about this problem. The US government established the National Automated Highway System Consortium (NAHSC) in 1995. Several promising prototype vehicles and systems have been demonstrated within the last 15 years [10]. The Navlab group at Carnegie Mellon University has a long history of investigations of automated vehicles and intelligent driver assistance systems with a series of 11 vehicles, Navlab 1 through Navlab 11. The latest model in Navlab family is the Navlab 11, a robot Jeep Wrangler equipped with a wide variety of sensors for short range and midrange obstacle detection [10-12].

Major motor companies, such as Ford and GM, have already demonstrated several promising vehicles. Recently, the US Department of Transportation (USDOT) has launched a five year, 35 million dollar project with GM to develop rear-end collision avoidance system [2]. In March 2004 and November 2007, the world was stimulated by the competitions, “grand challenge” and “urban challenge”, organized by the US Defense Advanced Research Projects Agency (DARPA). In these competitions, fully autonomous vehicles attempted to independently navigate within a fixed time period, all with no human intervention whatsoever – no driver, no remote-control, just pure computer processing and navigation horsepower.

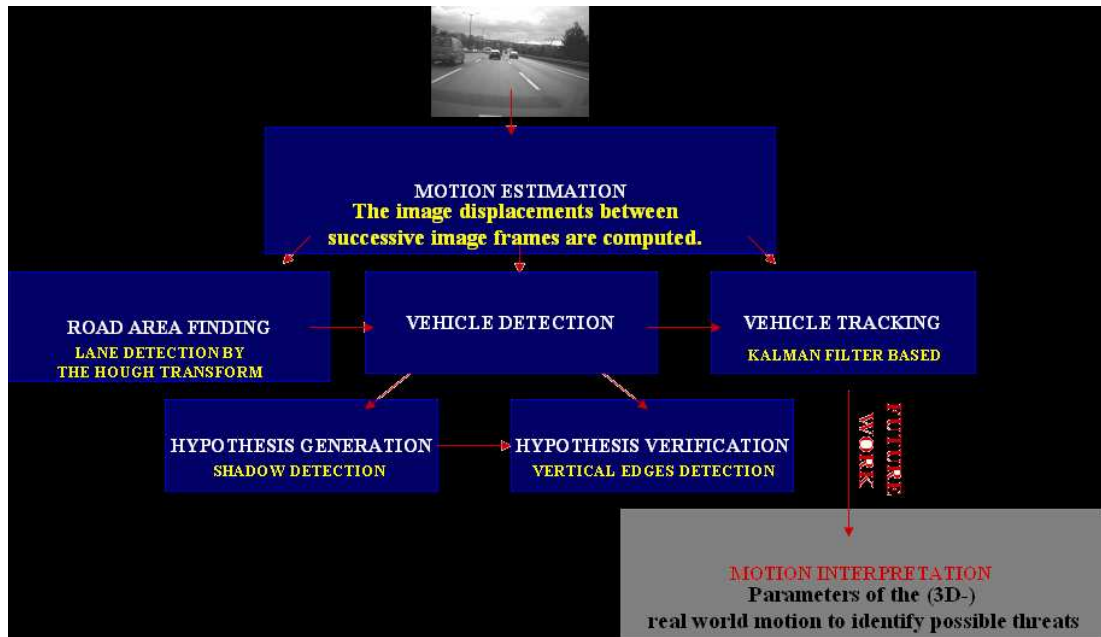


Figure 1.1 : Schematic overview of the objective of the thesis.

1.3 Thesis Structure

This thesis is organized as follows: Chapter 2 explains the approaches to the vehicle detection that have been proposed in the literature and the algorithms developed for the vehicle detection within the work of the thesis, which includes road area finding. In Chapter 3, the literature overview based on the object tracking is presented. In addition, the theory of the Kalman filter is mentioned and the implemented algorithm for the vehicle tracking based on the Kalman filter is explained in detail. Finally, Chapter 4 sums up the conclusions and presents the results of the evaluation of the developed algorithms.

2. VEHICLE DETECTION

From a general viewpoint vehicle detection is a problem of object detection, which is always an open issue in computer vision. Vision based vehicle detection requires a system that should be able to separate image data belonging to the background from the data belonging to the vehicles. Detection precedes the vehicle tracking.

2.1 Approaches Proposed in Literature

Various approaches have been proposed in the literature, which can be classified into one of the following three categories: 1) knowledge-based, 2) stereo-based, and 3) motion-based.

2.1.1 Knowledge-based methods

The Knowledge-based methods employ a priori information to extract vehicles. Different cues have been proposed in the literature and systems often include two or more of these cues to make detection more reliable.*

2.1.1.1 Symmetry

Images of vehicles observed from rear or frontal views are in general symmetrical in the horizontal and vertical directions. This observation has been used as a cue in several studies [13, 14]. When computing symmetry from intensity, the presence of uniform areas decreases the performance of the algorithm because these areas are sensitive to noise for symmetry estimations. Information about edges was included in the symmetry estimation to avoid from uniform areas [15]. Besides the fact that edges might not always be visible (object-background relation), this approach is still easily distracted by symmetrical background objects, such as houses.

* “Shadow” is also a cue within the “knowledge-based methods” used for vehicle detection. Using shadow as a cue for vehicle detection will be discussed in the following sections.

2.1.1.2 Color

Although color is a rare feature in literature, it is a very useful cue for obstacle detection, lane/road following [16- 18]. Color is liable for false detections and weak for non-colored vehicles. It can help in some situations anyway.

2.1.1.3 Vertical/ horizontal edges

Using constellations of vertical and horizontal line structures is one of the strongest cues used in literature for vehicle detection. This is because of the fact that different views of a vehicle contain many horizontal and vertical line structures, such as rear window, bumper, etc. In [19], the generalized Hough transform was used to identify rows and columns that might contain edges of the outer contour of a car. In [20], distant cars were identified by using projected edge information to extract pronounced horizontal and vertical edges, that might be part of a rectangular structure. Disadvantage of using these line structures is that they depend on the relation between object and background intensity and therefore the performance of the algorithm will decrease when e.g. a dark vehicle is observed against a dark background.

2.1.1.4 Texture

The presence of a vehicle in an image causes local intensity fluctuations. Due to general similarities among all vehicles, the intensity changes create a certain texture pattern [21]. Two approaches have been suggested in the literature: 1) using the entropy and 2) using the co-occurrence matrices [22]. Major difficulty of using texture as a cue for vehicle detection is that the background is also very likely to have texture.

2.1.1.5 Vehicle lights

Vehicle lights could be used as a salient visual feature for night time vehicle detection [23]. However, the vehicle light detection approach should only be seen as a complement to other approaches. Brighter illumination and the fact that vehicle lights are not compulsory to use during daytime in many countries makes it unsuitable for robust vehicle detection.

2.1.2 Stereo-based methods

Vehicle detection based on stereo vision uses two types of methods: the disparity map and Inverse Perspective Mapping. The difference in left and right images between corresponding pixels is called as disparity. The disparities of all the image points generates the disparity-map. A disparity histogram can be calculated from the disparity map. Since the rear-view of a vehicle is a vertical surface, and the points on the surface therefore are at the same distance from the camera, a peak in the histogram should occur [24].

The Inverse Perspective Mapping transforms an image point onto a horizontal plane in the 3D space. In [25], stereo vision was used to predict the image seen from the right camera, given the left image, using the Inverse Perspective Mapping.

Drawbacks of using stereo-vision are that traditional implementations are time consuming and robust solutions for the vehicle detection problem can only be obtained, if the camera parameters have been estimated accurately.

2.1.3 Motion-based methods

So far, clues based on spatial features to distinguish between vehicles and background were discussed. Another important cue for vehicle detection is relative motion. Pixels on the images appear to be moving due to the relative motion between the sensor and the scene. The vector field of this motion is referred to as optical flow. Examples of approaches based on the estimation of the optical flow field can be investigated in [26, 27]. In [26], the possibilities and drawbacks of using optical flow for vehicle detection were discussed. Optical flow can provide strong information for vehicle detection but it is sensitive to even small rotations of the camera and other mechanical disturbances and computing optical flow is time consuming because of the complexity.

2.2 Critique of Vehicle Detection Approaches

All the cues discussed within “the knowledge-based methods” use spatial features to distinguish between vehicles and background. Remember that the major difficulties of using the cues within this category are caused by the background since the background is also likely to have these features.

On the other hand, on-road vehicle detection requires faster processing than other applications related to optical sensors. Another key issue is that robustness to vehicle's movements and drifts must be considered. Remember that these two issues are the major difficulties of using the cues within "the stereo-based" and "the motion-based" approaches.

Consequently, different approaches to vehicle detection have been proposed in the literature as mentioned in the previous texts. Creating a robust system for vehicle detection using optical sensors is a very challenging problem. Special difficulties that make vehicle detection a challenge can be itemized as:

1. Since both camera and objects are in movement, the perceived size and pose of the objects change;
2. The objects exist in an environment that changes. Lighting and weather conditions vary substantially;
3. Vehicles might be occluded by other vehicles, buildings, etc;
4. The actual aspect of vehicles is quite wide;
5. For a precrash system to serve its purpose it is crucial to achieve real-time performance.

To cope with these difficulties, approaches in the literature are generally based on two-step vehicle detection: Hypothesis Generation and Hypothesis Verification.*

2.2.1 The first step: hypothesis generation

In the first step of vehicle detection, a vehicle's probable existence location is hypothesized. One or multiple cues are used within this step. Hypothesizing the locations of possible vehicles in the first step of vehicle detection decreases the whole image where vehicles are searched into the image regions where the vehicles probably exist. This decrease in the size of the image requires less processing time and therefore speeds up the process.

* Most of the information about the vehicle detection approaches in the literature was quoted from [28]. More detailed information about the vehicle detection approaches in the literature can be found in [28].

2.2.2 The second step: hypothesis verification

The existence of the located potential vehicles is verified in the second step of vehicle detection. The cues discussed within “the knowledge-based methods” can be used for the verification step. This kind of verification is generally called as “knowledge-based vehicle verification” or “template-based vehicle verification”.

Another category of the verification step can be called as “appearance-based vehicle verification”. Appearance-based methods learn the characteristics of the vehicle class from a set of training images, which should capture the variability in vehicle appearance. The verification using appearance models is treated as a two-class pattern classification problem: vehicle versus non vehicle. Usually, the variability of the non vehicle class is also modeled to improve the performance.

Appearance-based verification methods are more accurate than template-based methods; however, they are more costly due to classifier training. Nevertheless, due to the exponential growth in processor speed, appearance-based methods are getting popular.

2.3 Objective

Although the solutions to the vehicle detection problem are becoming more reliable and robust improving presented approaches and proposing new methods day by day, it is absolutely necessary to strictly define and delimit the problem due to the difficulties in conditions just mentioned in the previous texts. Detecting all vehicles in every possible situation is not realistic. The work in this thesis concerns with detecting trucks and busses as well as focusing largely on detecting personal vehicles. Detection under night illumination is not evaluated. The designed algorithms are tried to be improved to detect vehicles in various weather conditions and at any distance.

2.4 The Implemented Methods for Vehicle Detection within the Thesis

Template-based verification is used within the thesis in spite of all these advantages attached to the appearance-based verification. The reason is that appearance-based verification requires composing a training dataset and pattern classification background. Providing these requirements may have been a tough process. Implementing appearance-based vehicle verification is one of the future works planned to realize with the aim of improving the quality of the vehicle detection algorithm.

In practical applications within the literature, although it is possible to get rid of about two thirds of the image regions in which no vehicle exist using template-based verification, some backgrounds may still cause false detections. To avoid false detections of background, the method implemented within the thesis includes road area finding and searches possible vehicles inside this area.

The implemented algorithms for vehicle detection within the thesis can be classified as;

1. Road area finding: Lane detection,
2. Vehicle detection:
 - 2.1. Hypothesis generation: Shadow detection
 - 2.2. Hypothesis verification: Vertical edges detection.

The optical sensor used for image data acquisition is Basler A601FC color camera as shown in Figure 2.1. The resolution of the camera is 640 x 480 pixels and the frame rate is 30 frames per second (fps). The interface is IEEE 1394 high performance serial bus, also called as Firewire.



Figure 2.1 : Basler A601FC color camera.

All algorithms are implemented in MATLAB and monochrome images acquired from just one camera are processed within the thesis. The vision data is supplied by the experimental vehicle used for multi-modal data collection and processing within the Drive Safe Project in which İstanbul Technical University Automotive Control and Mechatronics Research Center is a participant. More detailed information on the Drive Safe Project can be found in [29, 30].

2.4.1 Road area finding

Finding road area is realized by means of a simple algorithm for detecting the free-driving-space of our vehicle – the host vehicle. The free-driving-space is defined as the road observed directly in front of the camera. Estimation of the free-driving-space is based on the lane detection algorithm implemented by Hough transform.

2.4.1.1 Hough transform

Edge detection methods yield pixels lying only on edges. In practice, the resulting pixels seldom characterize an edge completely because of noise, breaks in the edge from nonuniform illumination, and other effects that introduce spurious intensity discontinuities. Thus, edge detection algorithms typically are followed by linking procedures to assemble edge pixels into meaningful edges. One approach that can be used to find and link segments in an image is the Hough transform. In particular, it is used to extract lines, circles and ellipses in the images.

The Hough transform, illustrated in Figure 2.2, maps every point (x, y) in the image plane to a sinusoidal curve in the Hough space ($\rho\theta$ - space) according to:

$$y \cos \theta + x \sin \theta = \rho \quad (2.1)$$

where ρ can be interpreted as the perpendicular distance between the origin and a line passing through the point (x, y) and θ the angle between the x-axis and the normal of the same line.

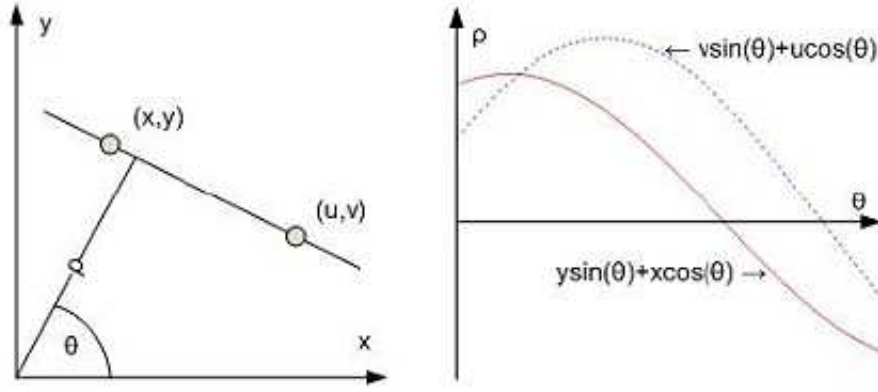


Figure 2.2 : The Hough transform transforms a point in the image plane to a sinusoidal curve in the Hough space. All image points on the same line will intersect in a common point in the Hough space [31].

The sinusoidal curves from different points along the same line in the image plane will intersect in the same point in the Hough space, superimposing the value at that point. In the second graphic, the intersection point corresponds to the line that passes through both (x, y) and (u, v) .

The computational attractiveness of the Hough transform arises from subdividing the $\rho\theta$ parameter space into so-called accumulator cell. Usually the expected maximum range of the parameters is $-90^\circ \leq \theta \leq 90^\circ$ and $-D \leq \rho \leq D$, where D is the distance between corners in the image (the diagonal of the image).

Initially the accumulator cell is set to zero. Then for each of the desired feature points (x_k, y_k) detected in the image plane, we let θ equal each of the predefined values within the θ range and solve for the corresponding ρ using the equation 2.1. The resulting ρ values are then rounded off to the nearest value within the predefined ρ range.

The corresponding element $A(i, j)$ of the accumulator cell defined with parameter space coordinates (ρ_i, θ_j) is then incremented. At the end of this procedure, a value of Q in $A(i, j)$, means that Q points in the xy -plane lie on the line $x \cos \theta_j + y \sin \theta_j = \rho_i$. By thresholding, dominant line segments can be detected.*

* Most of the information about the Hough transform was quoted from [32] and pages 393-395. More detailed information about the Hough transform can be found in [32].

2.4.1.2 Lane detection

Processing the whole image is unnecessary and thus time consuming while realizing lane detection. To focus on the lines that mark the lanes, the image is divided into two half images: Left half and right half as shown in Figure 2.3. The Hough transform is applied for each half part to detect lines.

Each lane line has two longitudinal edges that can be described as the transition from darker gray values to brighter ones or the transition from brighter gray values to darker ones in monochrome images as seen in Figure 2.4. Because of that one of these edges is enough to define the lane line, both half parts of the image are filtered by a simple mask such as $[1 \ 0 \ -1]$ or $[-1 \ 0 \ 1]$ before applying the Hough transform (See Figure 2.5).

There are, of course, many detected lines on the same lane line as seen in Figure 2.6. These lines must be reduced to one line as being one line on the lane line.

The algorithm is capable of giving two lines with a particular angle difference between them as an output for each half image. These lines are defined as Left-most, Left for the left half part of the image and Right-most, Right for the right half part of the image (as described in Figure 2.7).

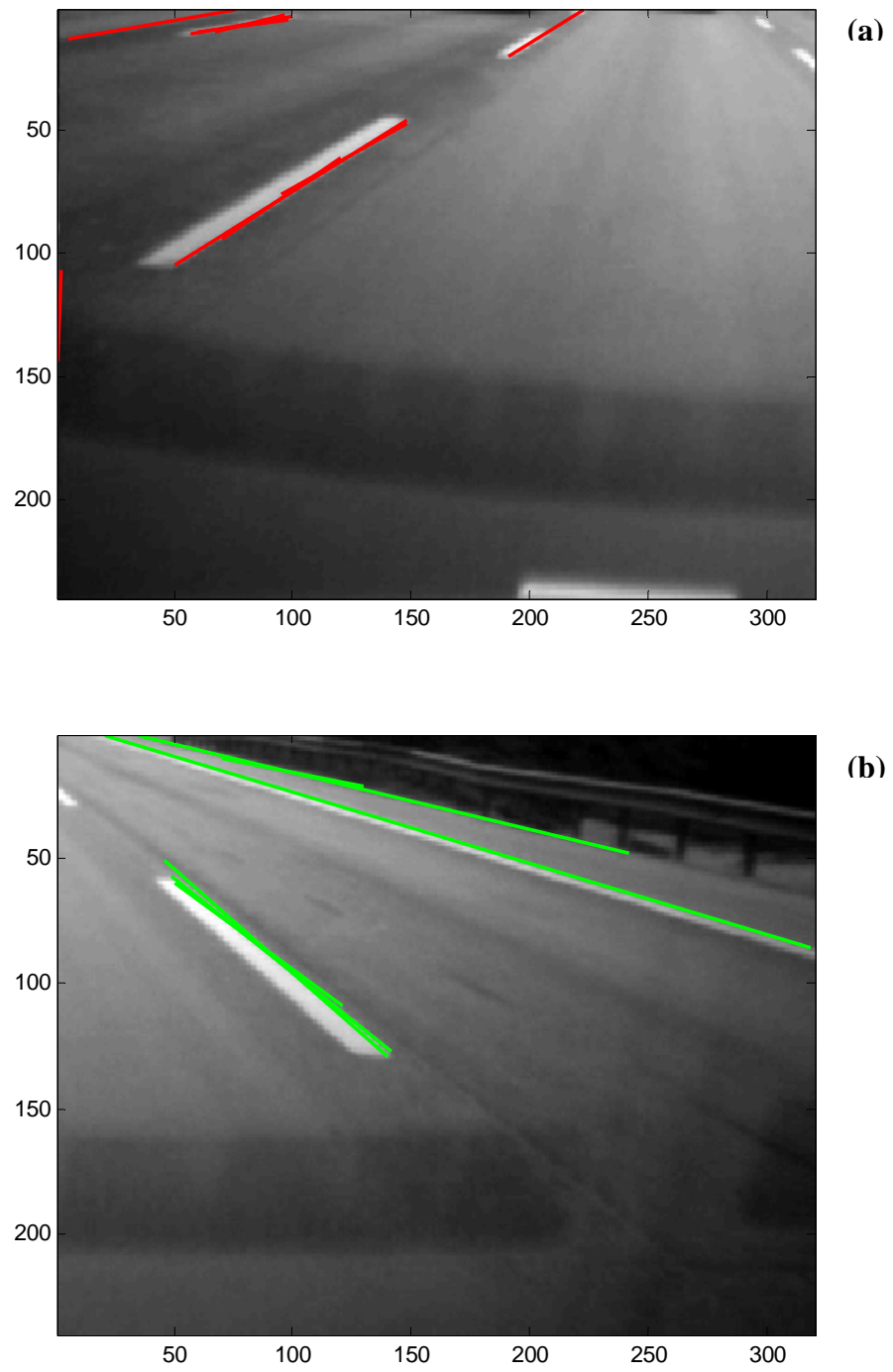


Figure 2.3 : (a) Detected lines in the left half (320 x 240) part of the image.
(b) Detected lines in the right half part of the image.



Figure 2.4 : Two longitudinal edges that can be described as the transition from darker gray values to brighter ones or the transition from brighter gray values to darker ones.

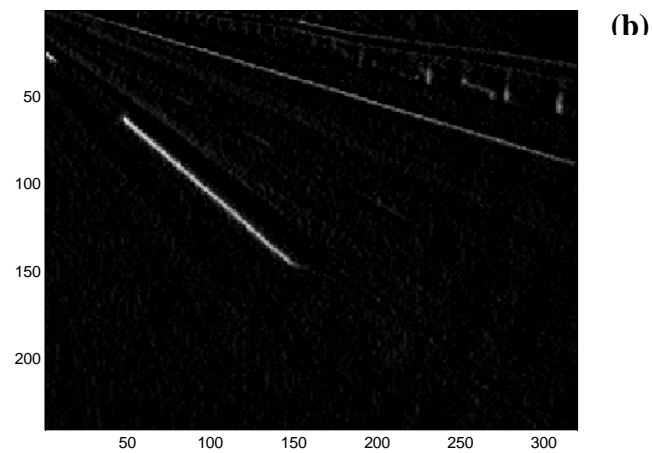


Figure 2.5 : (a) The original half image;
(b) The filtered half image by the mask $[-1 \ 0 \ 1]$.

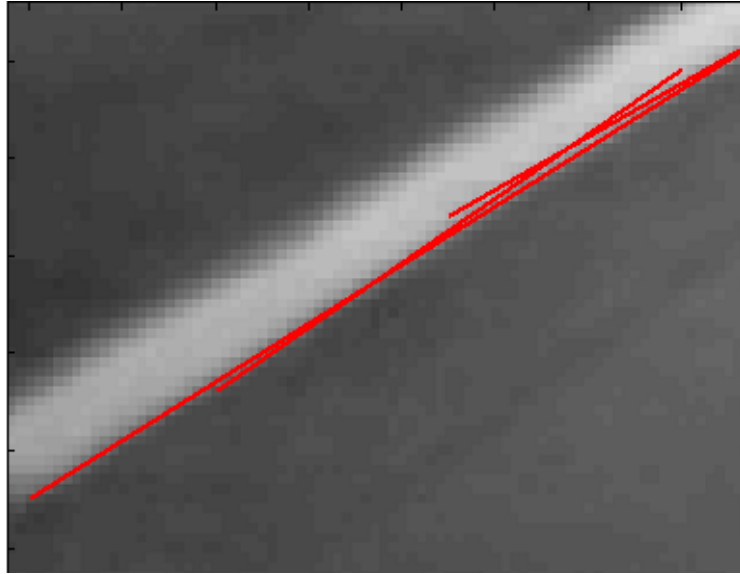


Figure 2.6 : Many lines are detected on the same lane line.

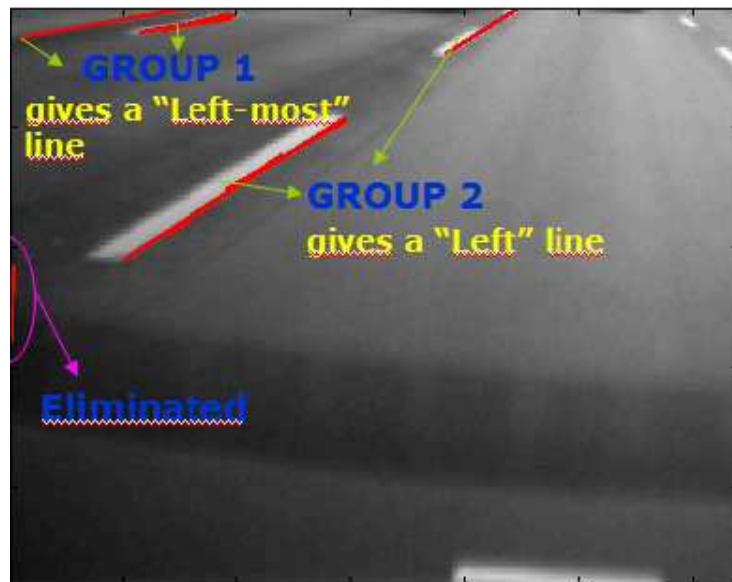


Figure 2.7 : The output of the algorithm for the left half part of the image:
Left-most line and Left line.

In figure 2.7, the output of the algorithm for the left half part of the image is illustrated. The same approach is also realized for the right half part of the image. The lines GROUP 1 are reduced to one line as giving a “Left-most” line and the lines within GROUP 2 are reduced to one line as giving a “Left” line.

It is possible to obtain lines that are irrelevant with lanes. These lines are easily eliminated utilizing the angle value given as an output for each line by the Hough transform.

Assuming that lanes have been successfully detected, vehicle presence is hypothesized by scanning each lane starting from the bottom to a certain vehicle position, corresponding to a predefined maximum distance in the real world.

In fact, it is difficult to acquire lane information in every frame of a sequence of images. The lane lines may not be easily eligible or may be interrupted by the vehicles. Developing a lane tracking algorithm may be a solution to this problem in some circumstances. Besides scanning each lane independently, it is also possible to group the lane lines that can be detected in the current frame to avoid from undetectable lane lines as seen in Figure 2.8.

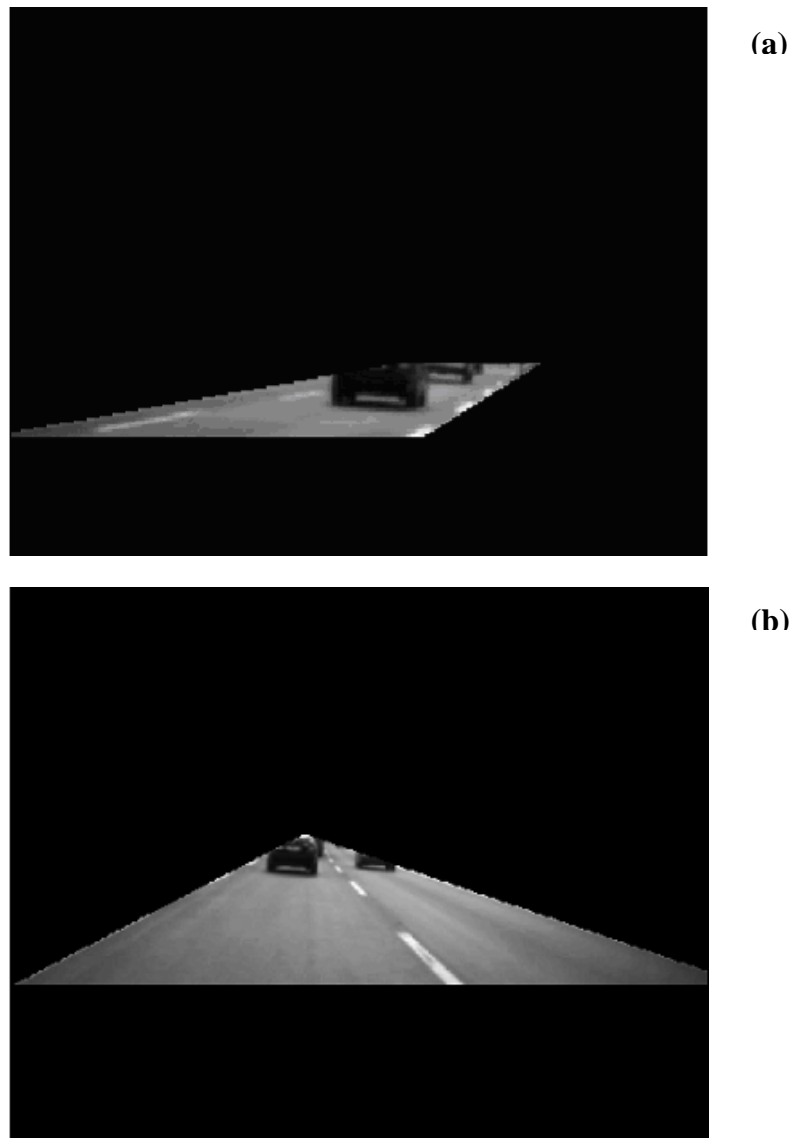


Figure 2.8 : (a) Road area identification. (b) Besides scanning each lane independently, it is also possible to group the lane lines that can be detected in the current frame.

2.4.2 Vehicle detection

As mentioned in the previous texts, vehicle detection process is realized in two steps:

1) Hypothesis generation, and 2) Hypothesis verification.

In the following parts of chapter 2, feature extraction techniques used as a basic of the vehicle detection process are not explained in detail. Detailed information about basic image processing operations and feature extraction techniques can be found in [32, 33].

2.4.2.1 Hypothesis generation – shadow detection

Vehicles may appear in many shapes and color. Nevertheless, one feature they all have in common is that they cause shadow on the road. Potential vehicle candidates can be extracted by detecting the shadows underneath vehicles.

In the literature, potential shaded areas are defined as intensities with a significant darker color than the road. In [34], a normal distribution is assumed for the intensity of the road surface and the threshold value of the shadow is defined based on the mean and variance of this distribution. The mean and deviation of different regions in a road may be different. Hence, this approach might not always hold true.

Another approach is based on looking in the image for vertical transitions from brighter gray values to darker ones. Instead of computing the mean of road pixels, pixels with negative vertical gradient values are considered as local darker regions [35].

To detect the shadows underneath vehicles, vertical transitions from brighter gray values to darker ones are searched in the image as scanning the image bottom-up. Considering the problem within this thesis, this approach can be realized implementing an edge detection algorithm as scanning predefined road area bottom-up.

The edges with vertical transitions – horizontal edges are obtained by a vertical edge detector. Sobel edge detector is implemented within the thesis and negative vertical gradient values less than a predefined threshold value are considered as local darker regions, as seen in Figure 2.9. A systematic way to choose appropriate threshold values was not developed within the thesis. Beside the fact that the intensity of the shadow depends on illumination of the image, which in turn depends on weather conditions, it is a weakness of the implemented algorithm. The threshold value was determined as an appropriate fixed value for a series of different training samples after testing on them.



Figure 2.9 : Detected shadows are plotted as red dots.

Shadow is used as an initial cue for vehicle detection within the thesis. Hence, false detections caused by applying a predefined, fixed threshold can be prevented in the following steps of the hypothesis generation as well as in the hypothesis verification. Nevertheless, in the weather conditions that the shadows underneath vehicles can not be distinctly eligible, the predefined threshold value might not be appropriate to detect the shadows underneath vehicles. Therefore, developing a systematic way to choose appropriate threshold values must be consider as a future work within this study.

Before implementing the following steps of the hypothesis generation algorithm, a simple preselection is performed. The shadow edges with the length smaller than a predefined pixel value are eliminated. This predefined pixel value can be selected as a value in the range of 10 – 15 pixel. The pixel values in this range are appropriate as potential bottom edges of ROIs (Region-of-Interest) for both mid-range and distant vehicles within further analysis – hypothesis verification step.

As seen in Figure 2.10, there are, of course, many shadow edges in successive rows, relating to the same vehicle. These edges must be reduced to one as representing the bottom edge of the potential vehicle.



Figure 2.10 : Successive shadow edges relating to the same vehicle.

The edges whose the distance between their “y” coordinates is less than or equal to “2” pixels are combined as giving the bottom edge of the potential vehicle. The value like “2” pixel is appropriate for both mid-range and distant vehicles while realizing this combining process.

The detected shadow edges underneath a vehicle do not always have the lengths same as or near to the length of the bottom edge of the vehicle. Shadow length changes during different weather conditions and times of the day. In this case, combining the shadow edges whose the lengths are more than a reasonable value requires facing with a critical situation during defining ROIs of the potential vehicles.

Defining ROI whose the size is considerably more than the size of the potential vehicle can cause false detections and thus verification errors for further analysis. In such a case, the background or eligible features of the other vehicles might be in the ROI defined for the hypothesized vehicle.

Evaluating each lane independently during the hypothesis generation step as solution to the problem described above might provide to obtain more reliable ROIs. However the fact that every lane is not detectable in each frame of an image sequence, grouping the detected lanes given a reasonable road area might be necessary, as mentioned in the previous texts. Besides this, if each lane is evaluated independently, detecting a vehicle while it is changing the lane might not be easy.

Consequently, grouping the detected lanes as defining a reasonable road area and evaluating them, in this manner, for the presence of vehicles is realized within the thesis. The problems within the hypothesis generation step are eliminated under these circumstances.

The width of a vehicle in an image is related to the width of the lane where the vehicle is currently located. Therefore a reasonable value for the width of the potential vehicle can be determined according to the width of the lane where the vehicle is currently present.

Since which lane the potential vehicle is present and the width of the lane where the potential vehicle is currently present are known, it is possible to calculate a value for the width of a potential vehicle according to the lane where it is present. To define ROIs that represent the potential vehicle for further analysis in the best possible way, the calculated value, just mentioned above, is utilized as a reference length for the bottom edge of the vehicle and, consequently, for the width of the potential vehicle.

The calculated value for the width of the potential vehicle and the proposed approach to calculate this value is more appropriate for passenger cars. ROIs defined for large size vehicles utilizing the mentioned approach does not sufficiently cover the area of the vehicle. However this is not a critical problem as much as defining ROIs whose the sizes are considerably more than the size of the vehicle, since the defined ROIs for large size vehicles are still have distinctive features for hypothesis verification step even if they do not sufficiently present the related vehicles.

In spite of the combining process, there might be still more than one edge over the same vehicle that could not be eliminated, as seen in Figure 2.11. The final step of the hypothesis generation is implemented to reduce these edges to one bottom edge for each hypothesized vehicle.

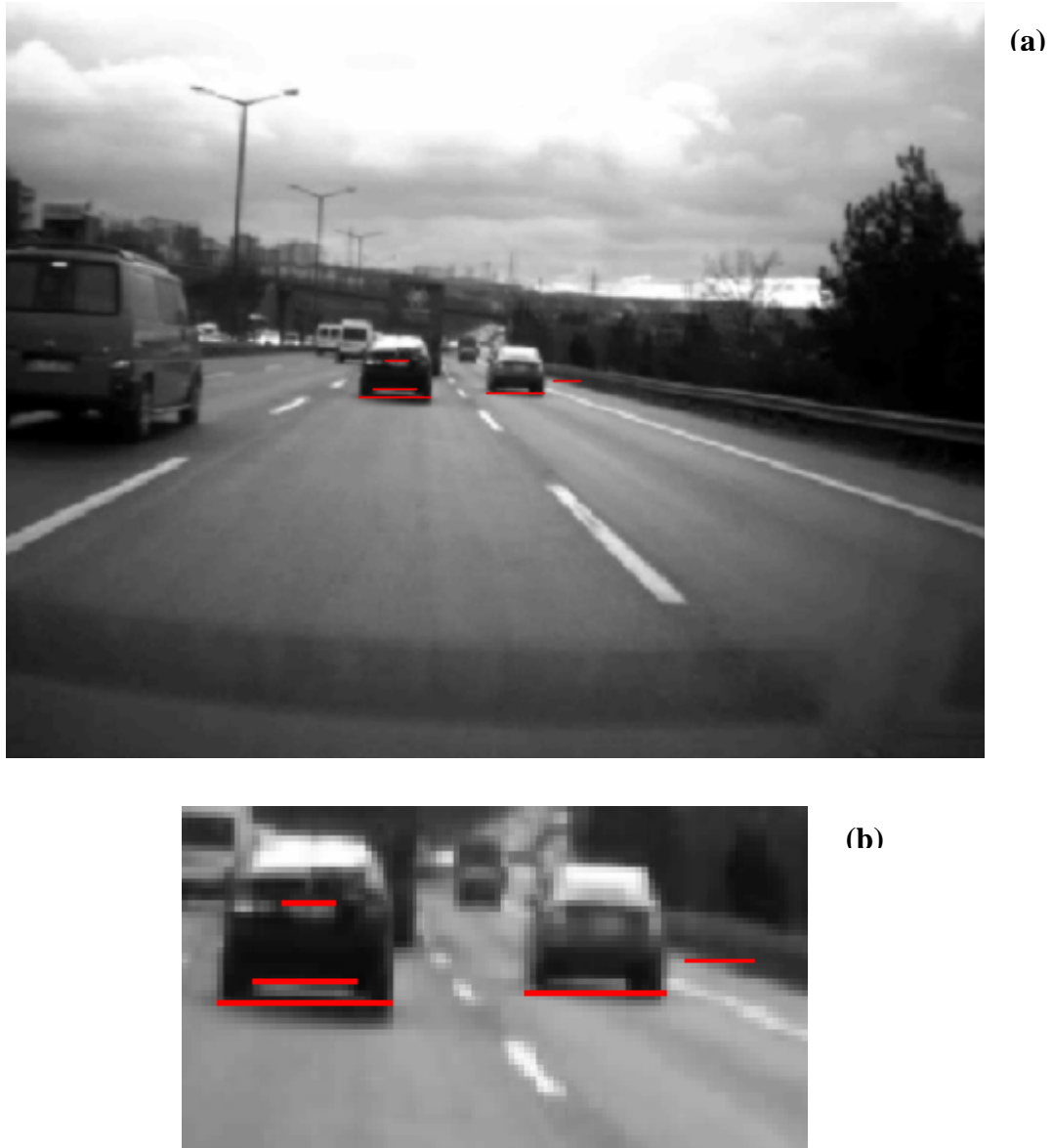


Figure 2.11 : (a) The edges that could not be eliminated in the combining process.
(b) An example of false hypotheses can also be seen at close range.

Consequently, the final bottom edges that represent each potential vehicle are utilized to determine the width of ROIs for the hypothesis verification step.

In the hypothesis verification step, the hypothesized presence of vehicles is verified and false hypotheses (one of the false hypotheses can be seen in Figure 2.11) are eliminated.

2.4.2.2 Hypothesis verification – vertical edges detection

Potential vehicles can be detected and located using shadow as discussed in the hypothesis generation step. Meanwhile, shadow can also be used for vehicle verification, since the located potential vehicle should have a shadow proper to its expected width corresponding to its location in the image. If the shadow is too wide or narrow, then it is rejected.

For each remaining potential vehicle, a region-of-interest is defined as described in Figure 2.12. The final bottom edge that represents a potential vehicle designate the width of a rectangular box hypothesized as forming the area of the vehicle. The potential bottom edge of ROI corresponding to the potential vehicle is defined enlarging the width of this hypothesized rectangular box. The bottom edge of the ROI is set as adding 6 pixels to the x coordinate of the end point of the shadow edge and subtracting 6 pixels from the x coordinate of the start point of the shadow edge. The value like “6” pixel is appropriate for different ranges the vehicles locate in the image. The side edge length of the ROI is determined as the half of the shadow edge length.

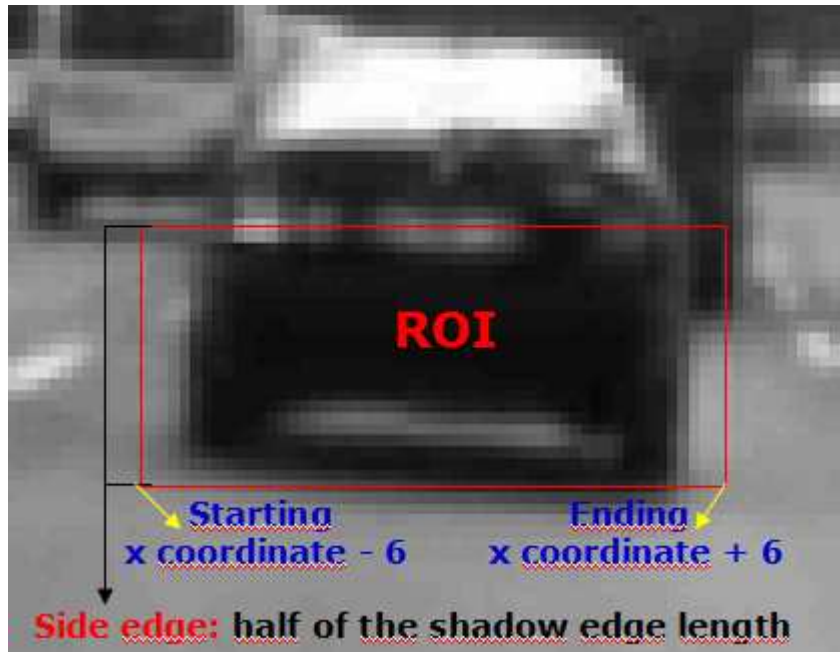


Figure 2.12 : Defining the region-of-interest (ROI).

Once the ROI is determined, refined search for the target vehicle is started in ROI. In the refined search, the horizontal projection vector w of the vertical edges V (Remember that the horizontal edge detector detects the vertical edges, [32, 33]) in the region defined as an $n \times m$ matrix is computed as follows:

$$w = (w_1, w_2, \dots, w_n, t) = \left(\sum_{j=1}^n V(x_1, y_j, t), \dots, \sum_{j=1}^n V(x_m, y_j, t), t \right) \quad (2.2)$$

The projection vector of the vertical edges is searched starting from the left and also from the right. The largest projection values found in both directions during the search determine the positions of the left and right sides of the potential vehicle.

To verify that the potential object is a vehicle;

If one horizontal edge and two vertical edges can be found in the same ROI, then it is considered that a vehicle exists in the image.

Since there are no consistent cues associated with the top of a vehicle, it can be detected by assuming that the aspect ratio of any vehicle is a predefined, specific value.

3. VEHICLE TRACKING

One of the essential qualities of intelligent driver assistance systems is the ability of tracking other vehicles on the road. There are three key steps in video analysis: 1) detection of interesting moving objects, 2) tracking of such objects from frame to frame, and 3) evaluation of object tracks to recognize their behavior. Chapter 2 described how vehicles could be detected and recognized from a single image. However, as we assume to analyze long image sequences, if the objects have been identified in the current frame or previous frames, this information could be used and will be helpful in the detection of objects in the next frame. In its simplest form, tracking can be defined as the problem of estimating the trajectory of an object in the image plane as it moves around a scene. In other words, a tracker assigns consistent labels to the tracked objects in different frames of a video [36].*

Vehicle tracking forms the basis for estimating parameters of the (3D) real world motion of the vehicles on the road. In this chapter, the algorithm used to track vehicles and extract 2D motion parameters is presented.

3.1 Literature Overview of Object Tracking

In machine vision, visual tracking is the process of extracting geometric information of the motion of an object from image data. The goal of visual tracking is to analyze specific attributes of a target via measurements obtained from a sequence of image data. For example, determining the image position (2D) of a target as the target object moves through the camera's field of view, or obtaining the pose of an object (3D position and orientation) may be intended to determine. Visual tracking is the problem known as the temporal correspondence problem: the problem of matching a target region through successive frames of a sequence of images typically taken at closely-spaced intervals.

* Most of the information about the object tracking and problem conditions was quoted from [36]. More detailed information can be found in [36].

The motion of an object in space causes changes in the image. The motion detected on the image, visual motion, is related to the motion in space. The motion field is defined as the 2D vector field of velocities of the image points, caused by the motion relative to the viewing camera. The motion field can be thought of as the projection of the 3D velocity field on the image plane. Determining the motion field provides the basis information so that one can obtain the 3D motion of objects. Detecting 2D motion in the image is generally classified into two categories: 1) optical flow, and 2) tracking. Optical flow, as mentioned in the motion-based detection methods, is based on estimating the apparent motion of the image brightness pattern. Optical flow differs from the true motion field except where the image gradients are strong. Much work in tracking is realized by utilizing the other category – the feature-based approach. The basis of the feature-based approach is the processing of the images to extract “features” (edges, regions of homogeneous color and/or texture, etc.). The feature-based approach has advantages. First, feature extraction reduces the vast amount of data present in the image, without necessarily eliminating salient information. Second, optical flow can only analyze the motion field along edges hence computing dense flow field can be counter-productive and computational expensive. In feature-based method, feature extraction reduces the whole image into subimage regions. Thus a comparatively computational efficiency can be provided. Feature-based tracking generally works in such a way: an object template is prestored as the basis of recognition and position, then in every next frame, template is matched. The matching is based on the output of a cost function. If the cost function is less than a predefined threshold value then it is assumed that target is present in the current frame. There are various cost functions, of which the most popular is sum-of-squared-difference (SSD). In [37], the detected vehicles are tracked using a combination of distance based matching, SSD and edge density of detected vehicle regions. In [38], recognition and localization of the preceding vehicle in the image is realized utilizing a correlation-based approach.

Due to the constraint of real-time performance, the challenge in visual tracking is to match the amount of data to be processed to the available computational resources. This can be done in a number of ways: simplifying the problem, utilizing specialized image processing hardware, designing clever algorithms, or all of them. Target of interest is not searched in whole image frame to increase efficiency of the algorithm. Template is matched in the Region of Interest (ROI) where target was likely to be found. ROI is determined based on the assumption that target can not move too much in consecutive two frames. Therefore, ROI will be somewhere in surrounding of the region where the last time the object was presented. However, it is possible that there may be significant change in target shape or orientation in the next frame. The image changes due to motion, illumination, and occlusions may causes errors in the measurements. If this is the case, then tracker starts losing the target.

In order to tackle the above-mentioned problem, the use of a sufficiently rich and accurate predictive model is required. In [20], The position and size of the target of interest is determined by a simple recursive filter with the aim of real-time multiple vehicle tracking from a moving vehicle. The Kalman filter is exactly useful as a solution to the problem mentioned above, handling noisy measurements (and also a noisy process). In [39], a real-time vision-based approach for detecting and tracking vehicles from a moving platform is developed. Tracking is realized by combining a simple image processing technique with a 3D extended Kalman filter and a measurement equation that projects from 3D model to image space. In [40,41], Kalman filter is used to produce optimal estimates of the state of a dynamic system with the aim of motion estimation of vehicles for in-car systems.

3.2 Problem Conditions

Ideally, a tracking algorithm would be able to locate the object anywhere within the image at any point in time. However typically only a limited region of the image is searched. Reasons for this are efficiency (especially necessary for real-time applications) and the possibility that there might be many other similar looking objects in the image.

The intuitive approach is to search within a region around the last position of the object. But as seen in Figure 3.1, this approach will fail if the object moves outside the target range. There are many possible reasons that occur this case:

1. The object is moving too fast.
2. The frame rate is too slow.
3. The searched region is too small.

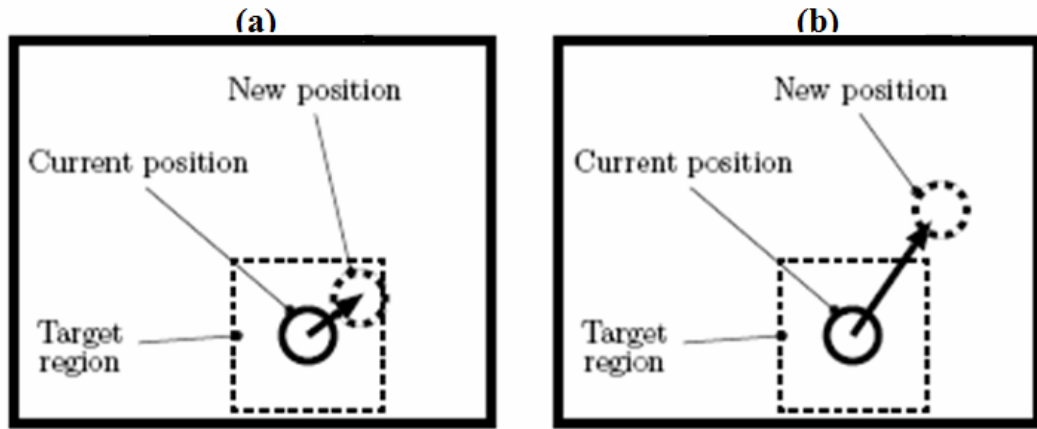


Figure 3.1 : (a) Tracking the object without position prediction might be successful. (b) Tracking without position prediction will fail.

These problems are related to each other and can be avoided by ensuring a high enough frame rate for example. But given other constraints, these problems are often inevitable.

In addition, even when the target can be accurately located. It seldomly appears the same in all images. Changes in orientation, lighting, occlusions, and imperfections in the camera continuously affect the appearance of the same target. So essentially, to observe the true location of the target in a certain manner is very difficult under the usual circumstances.

One can simplify tracking problem by imposing some constraints on the motion and/or appearance of objects. For example, almost all tracking algorithms assume that the object motion is smooth with no abrupt changes. One can further constrain the object motion to be of constant velocity or constant acceleration based on a priori information. Prior knowledge about the number and size of objects, or the object appearance and shape, can also be used to simplify the problem.

3.3 Objective

If a summary of the above-mentioned discussions is made, two major problems can be identified:

1. The object can only be tracked if it does not move beyond the searched region.
2. Various factors such as lighting and occlusions can affect the appearance of the target, thus making accurate tracking complex.

To solve the first problem, making predictions about the locations of the detected vehicles in successive frames of a long image sequence is attempted. But in making predictions, it is necessary to consider the second problem as well. Thus the prediction method needs to be robust enough to handle this source of error. A Kalman filter which estimates the positions and uncertainties of moving vehicles in the next frame is used within this master thesis. How large a region should be searched in the next frame for each target, that is, where to look for the target objects, around the predicted positions is determined by the Kalman filter to be sure to find the locations of the target objects within a certain confidence.

The region that covers the detected vehicle is called as “the bounding box”. Two control points for each bounding box are considered. The image coordinates of these control points are predicted for each next frame through an image sequence using the Kalman filter. The width of the bounding box in the image plane is computed using the image coordinates of the predicted control points. The ROI where the new target is searched, is defined expanding the predicted width of the bounding box in the image plane to a predefined pixel value.

3.4 The Theory of the Kalman Filter

The Kalman filter, rooted in the state-space formulation or linear dynamical systems, provides a recursive solution to the linear optimal filtering problem. The solution is recursive in that each updated estimate of the state is computed from the previous estimate and the new input data, so only the previous estimate requires storage. The Kalman filter is essentially a set of mathematical equations that implement a predictor-corrector type estimator that is optimal in the sense that it minimizes the estimated error covariance. In addition to eliminating the need for storing the entire past observed data, the Kalman filter is computationally more efficient than computing the estimate directly the entire past observed data at each step of the filtering process. The Kalman filter has been the subject of extensive research and application, particularly in the area of autonomous or assisted navigation. The Kalman filter has also been used extensively for tracking in interactive computer graphics [42].

Consider a linear, discrete-time dynamical system described by the block diagram shown in Figure 3.2. The Kalman filter addresses the general problem of trying to estimate the state of the discrete-time dynamical system that is governed by the linear stochastic difference equation. The state vector or simply state, denoted by x_k , is defined as the minimal set of data that is sufficient to uniquely describe the unforced dynamical behavior of the system; the subscript k denotes discrete time. In other words, the state is the least amount of data on the past behavior of the system that is needed to predict its future behavior. Typically, the state x_k is unknown. To estimate it, a set of observed data, denoted by the vector y_k , is used.

In mathematical terms, the block diagram of Figure 3.2 embodies the following pair of equations:

3.4.1 The process to be estimated

A discrete time process that is governed by the linear stochastic difference equation is defined as,

$$x_{k+1} = F_{k+1,k} x_k + w_k \tag{3.1}$$

with a measurement equation that is

$$y_k = H_k x_k + v_k \quad (3.2)$$

where $F_{k+1,k}$ is the transition matrix taking the state x_k from time k to time $k + 1$, y_k is the observable at time k and H_k is the measurement matrix.

The random variables w_k and v_k represent two additive noise terms: the process and measurement noise (respectively). They are assumed to be independent (of each other), white, with normal probability distributions and with covariance matrices defined by

$$E[w_n w_k^T] = \begin{cases} Q_k & \text{for } n = k \\ 0 & \text{for } n \neq k \end{cases} \quad (3.3)$$

where Q is the process noise covariance matrix and

$$E[v_n v_k^T] = \begin{cases} R_k & \text{for } n = k \\ 0 & \text{for } n \neq k \end{cases} \quad (3.4)$$

where R is the measurement noise covariance matrix.

If noises are uncorrelated, as is usually assumed to be the case, the off-diagonal terms are zero as described in the equation 3.3 and the equation 3.4. Most commonly the noise processes are assumed to be stationary; i.e., their statistics do not vary with time. The covariance matrices related to the noises are assumed to be constant.

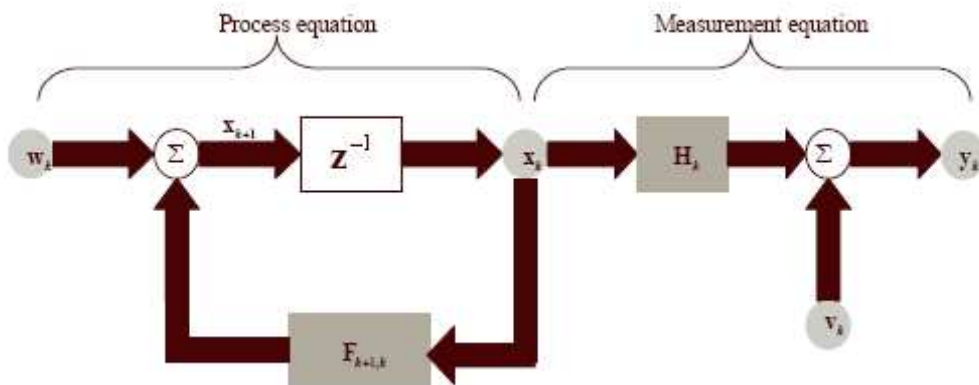


Figure 3.2 : Signal flow representation of a linear, discrete-time dynamical system [43] .

The Kalman filtering problem, namely, the problem of jointly solving the process and measurement equations for the unknown state in an optimum manner may now be formally stated as follows:

Use the entire observed data, consisting of the vectors y_1, y_2, \dots, y_k , to find for each $k \geq 1$ the minimum mean-square error estimate of the state x_k .

3.4.2 The computational origins of the filter

A priori state estimate at step k is defined as $\hat{x}_k^- \in \Re^n$ (note the “super minus”) given knowledge of the process prior to step k , and a posteriori state estimate at step k is defined as $\hat{x}_k \in \Re^n$ given measurement y_k . Then, a priori and a posteriori estimate errors can be depicted as

$$\begin{aligned} e_k^- &\equiv x_k - \hat{x}_k^-, \text{ and} \\ e_k &\equiv x_k - \hat{x}_k. \end{aligned} \tag{3.5}$$

The a priori estimate error covariance is then

$$P_k^- = E[e_k^- e_k^{-T}], \tag{3.6}$$

and the a posteriori estimate error covariance is

$$P_k = E[e_k e_k^T]. \tag{3.7}$$

In deriving the equations for the Kalman filter, finding an equation that computes an a posteriori state estimate \hat{x}_k as a linear combination of an a priori state estimate \hat{x}_k^- and a weighted difference between an actual measurement y_k and a measurement prediction $H \hat{x}_k^-$ is the initial goal, as shown below in equation (3.8).

$$\hat{x}_k = \hat{x}_k^- + K(y_k - H \hat{x}_k^-) \tag{3.8}$$

The difference $(y_k - H \hat{x}_k^-)$ in equation (3.8) is called the measurement innovation, or the residual. The residual reflects the discrepancy between the predicted measurement $H \hat{x}_k^-$ and the actual measurement y_k . A residual of zero means that the two are in complete agreement.

The matrix K in equation (3.8) is chosen to be the gain or the blending factor that minimizes the a posteriori estimate error covariance equation (3.7). The implementation of this minimization can be found in [30,31]. One form of the resulting K that minimizes equation (3.7) is given by

$$\begin{aligned} K_k &= P_k^- H^T (H P_k^- H^T + R)^{-1} \\ &= \frac{P_k^- H^T}{H P_k^- H^T + R} \end{aligned} \quad (3.9)$$

Looking at equation (3.9), as the measurement noise covariance R approaches zero, the gain K weights the residual more heavily. On the other hand, as the a priori estimate error covariance P_k^- approaches zero, the gain K weights the residual less heavily.

3.4.3 The probabilistic origins of the filter

The Kalman filter maintains the first two moments of the state distribution,

$$\begin{aligned} E[x_k] &= \hat{x}_k \\ E[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T] &= P_k \end{aligned} \quad (3.10)$$

The a posteriori state estimate equation (3.8) reflects the mean (the first moment) of the state distribution – it is normally distributed if the conditions of equation (3.3) and (3.4) are met. The a posteriori estimate error covariance equation (3.6) reflects the variance of the state distribution (the second non-central moment).

More details on the probabilistic origins of the Kalman filter can be found in [42].*

3.4.4 The summary of the discrete Kalman filter algorithm

The equations for the Kalman filter fall into two groups: time update equations and measurement update equations. The time update equations are responsible for projecting forward (in time) the current state and error covariance estimates to obtain the a priori estimates for the next time step. The measurement update equations are responsible for incorporating a new measurement into the a priori estimate to obtain an improved a posteriori estimate.

* Most of the information about the theory of the Kalman filter was adapted from [42]. More detailed information about the Kalman filter can be found in [42].

The time update equations can also be thought of as predictor equations, while the measurement update equations can be thought of as corrector equations. The specific equations for the time and measurement updates are presented in the following equations. A complete description of the operation of the filter can also be found in Figure 3.3:

$$\hat{x}_k^- = F_{k,k-1} \hat{x}_{k-1} \quad (3.11)$$

$$P_k^- = F_{k,k-1} P_{k-1} F_{k,k-1}^T + Q_k \quad (3.12)$$

The equations described above (3.11 and 3.12) are the discrete Kalman filter time update equations. How the time update equations project the state and covariance estimates forward from time step $k - 1$ to step k can be seen clearly.

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1} \quad (3.13)$$

$$\hat{x}_k = \hat{x}_k^- + K_k (y_k - H \hat{x}_k^-) \quad (3.14)$$

$$P_k = (I - K_k H) P_k^- \quad (3.15)$$

The equations described above (3.13 – 3.15) are the discrete Kalman filter measurement update equations.

3.5 Dynamical System Formulation of the Implemented Vehicle Tracking

The attribute sought at any point in time described by the state vector x_k . Often this state vector contains the coordinates of the target with respect to a chosen reference frame. The two control points of the bounding box of the vehicle in the image are considered. The bounding box refers to a rectangular that covers the area of the vehicle. These two control points are chosen as the bottom left and right points of the bounding box $p_k = [x_{k,1}, y_{k,1}, x_{k,2}, y_{k,2}]^T$ where the subscript k denotes the frame of the sequence under consideration (See Figure 3.4). Within the image, the control points related to the bounding box of the vehicle move with velocity $v_k = [v_{x,k,1}, v_{y,k,1}, v_{x,k,2}, v_{y,k,2}]^T$.

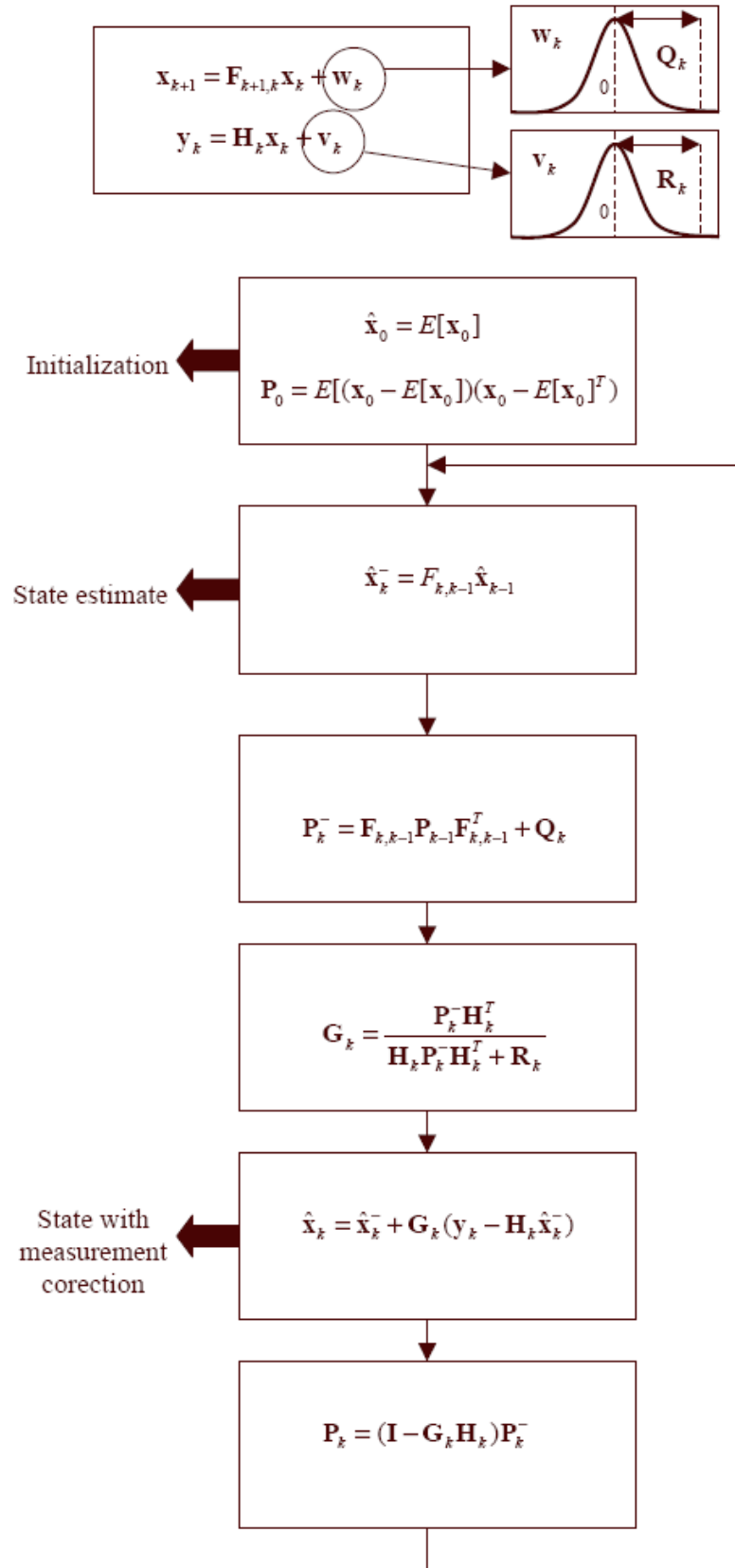


Figure 3.3 : A complete description of the operation of the Kalman filter [43] .

A state vector

$$x_k = [x_{k,1}, y_{k,1}, x_{k,2}, y_{k,2}, v_{x,k,1}, v_{y,k,1}, v_{x,k,2}, v_{y,k,2}]^T \quad (3.16)$$

can be chosen to describe the motion of the bounding box on the image plane. Nevertheless, since the two control points of the bounding box that is chosen to track are the points on the same horizontal edge (the bottom edge of the bounding box), the state vector is reduced to

$$x_k = [x_{k,1}, y_{k,1}, x_{k,2}, v_{x,k,1}, v_{y,k,1}, v_{x,k,2}]^T. \quad (3.17)$$

The position and size of the region-of-interest, in other words the tracking window, in subsequent frames is determined by predicting this state vector in terms of the theory behind the Kalman filter. Therefore the chosen state vector in Equation (3.17) is appropriate to predict the position and size of the region-of-interest in subsequent frames.

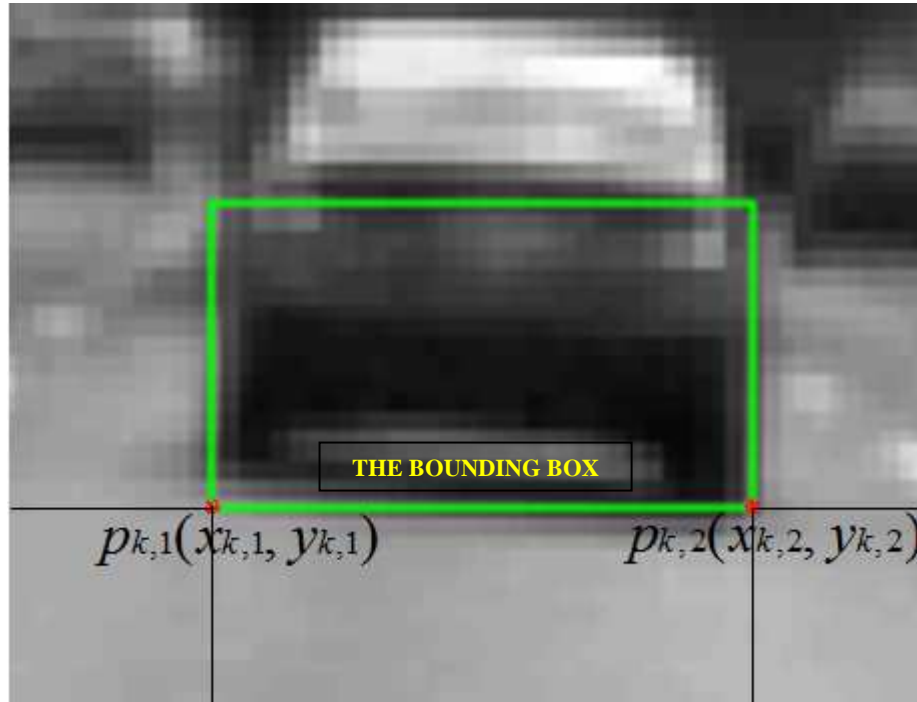


Figure 3.4 : The description of the bounding box and the control points.

If a sufficiently small sampling interval, δt , is assumed, a constant velocity between frames can also be assumed. The motion can be expressed as:

$$\begin{aligned}
p_k &= p_{k-1} + v_{k-1} * \delta_t + \xi_{k-1} \\
v_k &= v_{k-1} + \eta_{k-1}
\end{aligned} \tag{3.18}$$

where ξ_{k-1}, η_{k-1} are the uncertainty in the model, usually taken to be zero-mean, white, Gaussian random processes. Re-writing this in terms of the state vector, a dynamical model of the target motion is obtained as

$$x_k = \Phi_{k-1} x_{k-1} + w_{k-1} \tag{3.19}$$

where

$$\Phi_{k-1} = \begin{bmatrix} 1 & 0 & 0 & \delta_t & 0 & 0 \\ 0 & 1 & 0 & 0 & \delta_t & 0 \\ 0 & 0 & 1 & 0 & 0 & \delta_t \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.19a}$$

and

$$w_{k-1} = \begin{bmatrix} \xi_{k-1} \\ \eta_{k-1} \end{bmatrix} \tag{3.19b}$$

w_{k-1} is the uncertainty in the process; i.e., process noise. ξ_{k-1} can be assumed to be zero and the uncertainty in the process can thus be defined as:

$$\begin{aligned}
w_{k-1} &= \begin{bmatrix} 0 & 0 & 0 & u_{x,k-1,1} & u_{y,k-1,1} & u_{x,k-1,2} \end{bmatrix}^T \\
\eta_{k-1} &= \begin{bmatrix} u_{x,k-1,1} & u_{y,k-1,1} & u_{x,k-1,2} \end{bmatrix}^T
\end{aligned} \tag{3.20}$$

As to measurements, the positions of the bounding box control points, p_k , at every frame of a sequence are evaluated. Therefore, the measurement model of the Kalman filter becomes

$$z_k = H_k x_k + \mu_k \tag{3.21}$$

where

$$H_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad (3.21a)$$

and μ_k is the uncertainty in the measurement; i.e., measurement noise (again, often assumed to be a zero-mean, white, Gaussian random process).

3.5.1 The initialization of the Kalman filter

The main problem with Kalman filtering is that statistical models are required for the system and the measurement instruments. Unfortunately, they are typically not available, or difficult to obtain. In the actual implementation of the filter, the measurement noise covariance R is usually measured prior to operation of the filter. Measuring the measurement error covariance R is generally practical (possible). An off-line analysis of the measurement instruments prior to running the process (system identification) can be made to determine the variance of the measurement noise. The determination of the process noise covariance Q is generally more difficult because the process can not be observed directly. In other words, if the measurements in the off-line analysis also contain errors, the process can not be accurately profiled. Sometimes a relatively simple (poor) process model can produce acceptable results if enough uncertainty is injected into the process via the selection of Q . Certainly in this case, the process measurements must be reliable. Whether or not a rational basis for choosing the parameters is provided, often times superior filter performance (statistically speaking) can be obtained by tuning the filter parameters. These parameters can be pre-computed, for example, by determining the steady-state value under conditions where Q and R are in fact constant.

Since H_k is a 3 x 6 matrix, the three additive noises are assumed zero-mean white, uncorrelated with each other, and with variances $\sigma_{x_{k,1}}^2(k)$, $\sigma_{y_{k,1}}^2(k)$, $\sigma_{x_{k,2}}^2(k)$, respectively. The measurement noise covariance matrix, needed for the Kalman filter implementation, is thus given by

$$R_k = E[\mu_k \mu_k^T] = \begin{bmatrix} \sigma_{x_{k,1}}^2 & 0 & 0 \\ 0 & \sigma_{y_{k,1}}^2 & 0 \\ 0 & 0 & \sigma_{x_{k,2}}^2 \end{bmatrix} \quad (3.22)$$

The process noise covariance matrix is formally defined as $Q_{k-1} \equiv E[w_{k-1}w_{k-1}^T]$. Using the definition of the noise vector w_{k-1} and the assumption that the process-noise terms $u_{x,k-1,1}$, $u_{y,k-1,1}$, $u_{x,k-1,2}$ are uncorrelated;

$$Q_{k-1} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_1^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_2^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_3^2 \end{bmatrix} \quad (3.23)$$

where $\sigma_1^2 = E[u_{x,k-1,1}^2]$, $\sigma_2^2 = E[u_{y,k-1,1}^2]$ and $\sigma_3^2 = E[u_{x,k-1,2}^2]$ represent the variance of the noise terms. Remember that these terms ($u_{x,k-1,1}$, $u_{y,k-1,1}$, $u_{x,k-1,2}$) represent the change in the velocity (Equation 3.18).

Specific numbers must of course be put in for those variances in order to define the Kalman filter numerically. To do this, a model for the vehicle acceleration that is simple and appears reasonable on physical grounds [44] is used to model the 2D image motion within the thesis. The vehicle acceleration u in either of the two directions (image coordinates; x and y) is assumed to be random and equally likely to be positive or negative with some maximum value A . The acceleration is assumed to be uniformly distributed between $\pm A$. The probability density function of the acceleration in either direction is thus assumed to have the form of Figure 3.5. Three impulse functions representing discrete probabilities at $\pm A$ and 0 acceleration have been superimposed to make the model a little flexible. These then simply say that there is a probability P_2 that the vehicle will proceed at constant image velocities, while there is probability P_1 that its acceleration (deceleration) in either direction is at the maximum value A . The height of the uniform distribution is just $a = (1 - 2P_1 - P_2) / 2A$ and that the variance of the random variable u is given by

$$\sigma_u^2 = \frac{A^2}{3}(1 + 4P_1 + P_2) \quad (3.24)$$

To find σ_1^2 , σ_2^2 and σ_3^2 , $\sigma_n = T\sigma_u$ must be considered (T is time interval and n = 1,2,3). Thus,

$$\sigma_n^2 = T^2 \sigma_u^2 = \frac{A^2 T^2}{3} (1 + 4P_1 + P_2), \quad \text{where } n = 1, 2, 3. \quad (3.25)$$

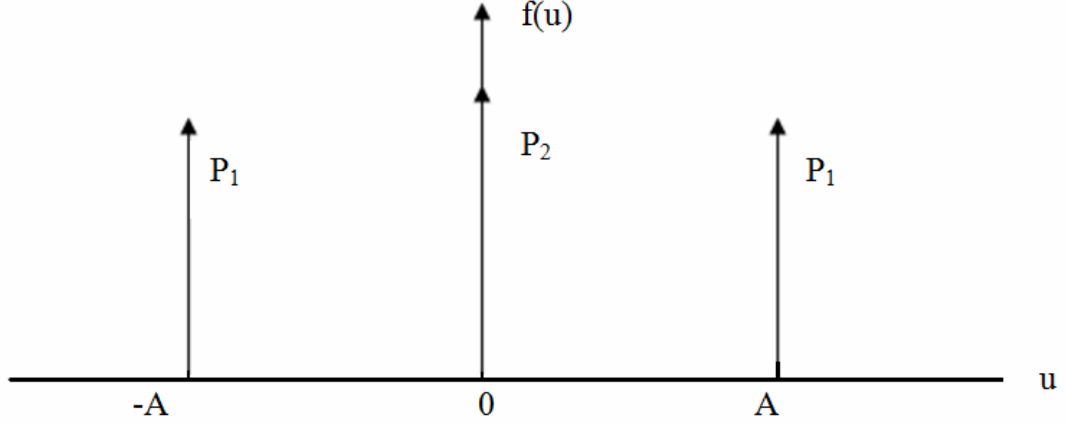


Figure 3.5 : Assumed probability distribution of the acceleration u .

Although the noise is assumed to be stationary, so that the variances do not vary with time, it may be possible to employ an algorithm which adjusts these process noise variances after each time step based on the observed measurements and evaluated change in the velocities considering these measurements.

Filter initialization requires a first error covariance matrix as well as the noise covariance matrices. From its definition, the error covariance matrix is given as $E[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T] = P_k$. The diagonal terms are just the mean-squared errors in the signal vector estimates. To initialize the filter, a first estimate is required as well as a first covariance matrix corresponding to the use of that first estimate. A first estimate can be found in several ways [44]. In some estimation problems an optimal (least mean squared error) can be found using the orthogonality principle, or, equivalently, by starting with a previous $\hat{x}_0 = 0$, which is indeed the optimal estimate of the zero-mean signal components when no observations are available. In such cases, the corresponding error covariance matrix P_0 would be simply the steady-state covariance matrix C of the signal vector since

$$P_0 = E[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T] = E[x_0 x_0^T] = E[x_k x_k^T] = C \quad (3.26)$$

3.6 The Implemented Algorithm

The tracking algorithm implemented within the thesis uses the following steps:

1. After the recognition of the vehicles is realized in the vehicle detection process and the current state vector is determined, tracking starts from the next image.
2. Repeat for each frame in the image sequence:
 - Use the dynamical model to predict the position of the detected vehicle in the image.
 - Calculate the region-of-interest for the predicted vehicle position. The ROI is determined as described in the hypothesis verification step of the vehicle detection process explained in Chapter 2 (See Section 2.4.2.2).
 - In determined ROI, search for the corresponding vehicle by using pronounced horizontal (shadow edge) and vertical edges.
 - Once the tracked vehicle is found, get the optimal estimation of the tracked vehicle in the current frame.
 - Update the position of the tracked vehicle based on the measurements corresponding to the position of the vehicle in the current frame.

The same tracking process is realized for each vehicle recognized in the vehicle detection process.

The vehicle detection algorithm is called every 10th frame due to the possibility of the presence of the new vehicles. It is possible for an object in the image not to be detected from one or two images. Hence, any of the detected and tracked vehicles may not be detected in the next call of the vehicle detection algorithm after 10 frames, even if it is still there.

Employing the capabilities of the tracking algorithm might be reasonable to avoid this problem. Before the elimination of the vehicle that is no longer detectable in the current frame, the sub-region that is the boundary box of the vehicle in the previous frame is correlated with the sub-region in the current frame whose the size and the position is the same with the boundary box in the previous frame. If the normalized correlation of the image regions is high, it is inferred that the vehicle might be still there.

3.6.1 To update the filter: horizontal and vertical edges detection

In each determined ROI, a refined search is realized to detect horizontal edge (the shadow edge) and vertical edges that is the vertical sides of the vehicle. The ROI is determined as explained in the hypothesis verification step of the vehicle detection algorithm and the vertical sides of the vehicle are also extracted by the same way defined in the hypothesis verification step. Nevertheless, in contrast with the vehicle detection algorithm, the horizontal edge (the shadow edge) detection must be realized in each determined ROI to locate the corresponding vehicle in the image. The horizontal edges are extracted implementing a Sobel edge detector. The projection vector of the horizontal edges in the ROI (defined as an $n \times m$ matrix) is computed as follows:

$$v = (v_1, v_2, \dots, v_m, t) = \left(\sum_{i=1}^m H(x_i, y_1, t), \dots, \sum_{i=1}^m H(x_i, y_n, t), t \right) \quad (3.27)$$

Because of that the top horizontal edge of the vehicle is not tried to detect, the projection vector of the horizontal edges is searched from the bottom of the vector to the middle of it. The largest projection value determines the position of the bottom edge of the vehicle.

Another difference from the vertical edges detection in the hypothesis verification step is the selection of the threshold value. Even if the ROIs are accurately representing the vehicles in the image, there might be still distinctive features of the other vehicles in the same ROI, that can cause false detections in the situations such as vehicle occlusions.

To handle with this problem (especially the false detections problem caused by the occlusions), two threshold values are determined based on the literature survey and the observations obtained within the thesis: 1) The half of the largest projection value and 2) the largest projection value. Firstly, the projection vector is searched from the left and also from the right until a vector entry that is more than or equal to the half of the largest projection value is found. The maximum change in the image coordinates of the vertical sides of each vehicle is stored during the execution of the tracking algorithm. If the positions of the vector entries found based on the first threshold value (the half of the largest projection value) cause a change in the coordinates more than the maximum one, a new search is started using the second threshold value (the largest projection value). Otherwise, the positions of the vector entries selected using the first threshold determine the positions of the left and right sides of the vehicle.

4. CONCLUSION AND RECOMMENDATIONS

This thesis proposed a multiple vehicle detection and tracking system, which includes road area finding. Video captured by just one camera is used to detect and track multiple vehicles. The system gives information of the ongoing traffic via the camera mounted on the rear-view mirror of the host vehicle.

In the first step of the detection algorithm, the locations of the potential vehicles in the image are hypothesized using the shadows underneath a vehicle (as a distinctive feature) scanning the defined road area bottom-up to avoid false detections of delineators. The road area is defined using the lane information obtained by the Hough transform.

In the second step of the detection algorithm, the hypothesized locations of the potential vehicles in the image are verified using the vertical edges as well as the shadows underneath a vehicle. During the verification, the presence of a vehicle is considered if one horizontal edge and two vertical edges can be found. The summary of the vehicle detection algorithm is illustrated in Figure 4.1.

After extracting vehicles, the developed tracking algorithm effectively track them during successive image frames in a long image sequence using a Kalman filter based approach. Finally, the 2D image velocity relative to the host vehicle for each detected vehicle is provided. The flow chart of the implemented algorithms can be seen in Figure 4.2.

The shadow detection step of the vehicle detection algorithm can be considered as a coarse search, while the detection algorithm is employed only for small regions represented each potential vehicle after the region-of-interests defined for each potential vehicle. The coarse search is implemented over the defined road-area, while ROIs make possible to employ a refined search over located small regions. Hence, the coarse search takes a substantial amount of time. It can take about 1–1.5 seconds depending on the number of the detected shadows.

However, if the dynamics of the moving objects are known, predictions can be made about the positions of the objects in the current image and the relevant positions of the moving objects can be estimated in successive frames of an image sequence. The Kalman filter based tracking algorithm implemented within the thesis can reduce the processing time needed to execute the vehicle detection algorithm to approximately 0.02 seconds.



Figure 4.1 : The summary of the detection algorithm.

The developed algorithms within the thesis were implemented by MATLAB. Besides the fact that MATLAB does not provide a sufficient performance for this kind of vision applications related to the time constraint in comparison with the environments C/C++ based programs can be executed and mainly real-time computer vision is aimed, the developed algorithms should be translated from the MATLAB implementation to C/C++ using this kind of environments just mentioned about.

The most serious drawback of using the shadow cue for vehicle detection is scenes with low sun, making vehicles cast long shadows (See Figure 4.3). Hence, the detected shadows become wider in the case of a sun from the side or ill positioned in the case of the camera facing the sun.

As mentioned in the hypothesis generation step (See section 2.4.2.1), the shadow lengths change due to the different weather conditions and even times of the day. This situation can cause defining ROI whose the size is considerably more than the size of the potential vehicle.

Defining such a ROI can cause false detections thus verification errors for further analysis due to the background or eligible features of the other vehicles where might be in the same ROI. Such false detections might especially occur in the situation the vehicles in adjacent lanes. Surprisingly, this problem has not been mentioned enough in the literature.

As a solution to the mentioned problem, the information of the detected lanes is utilized within the thesis. Beside the fact that the width of a vehicle in an image is related to the width of the lane where the vehicle is currently present, it is possible to calculate a reference value for the width of a potential vehicle according to the lane where it is present. If the width of the potential shadow edge is too wide compared to the reference value, then this shadow edge is eliminated (See section 2.4.2.1).

Most of the previous vehicle detection and tracking methods used lane or determined free driving space as in the work implemented within this thesis. However, if the lane does not exist or due to an intersection, etc., it is difficult to acquire such information.

Although searching the road area defined via the lane information to extract distinctive cues reduces the computational cost in comparison with searching the whole image, using appearance-based verification methods to verify the presence of a potential vehicle hypothesized by searching the whole image can provide a more robust algorithm to handle the problem associated with the lane detection.

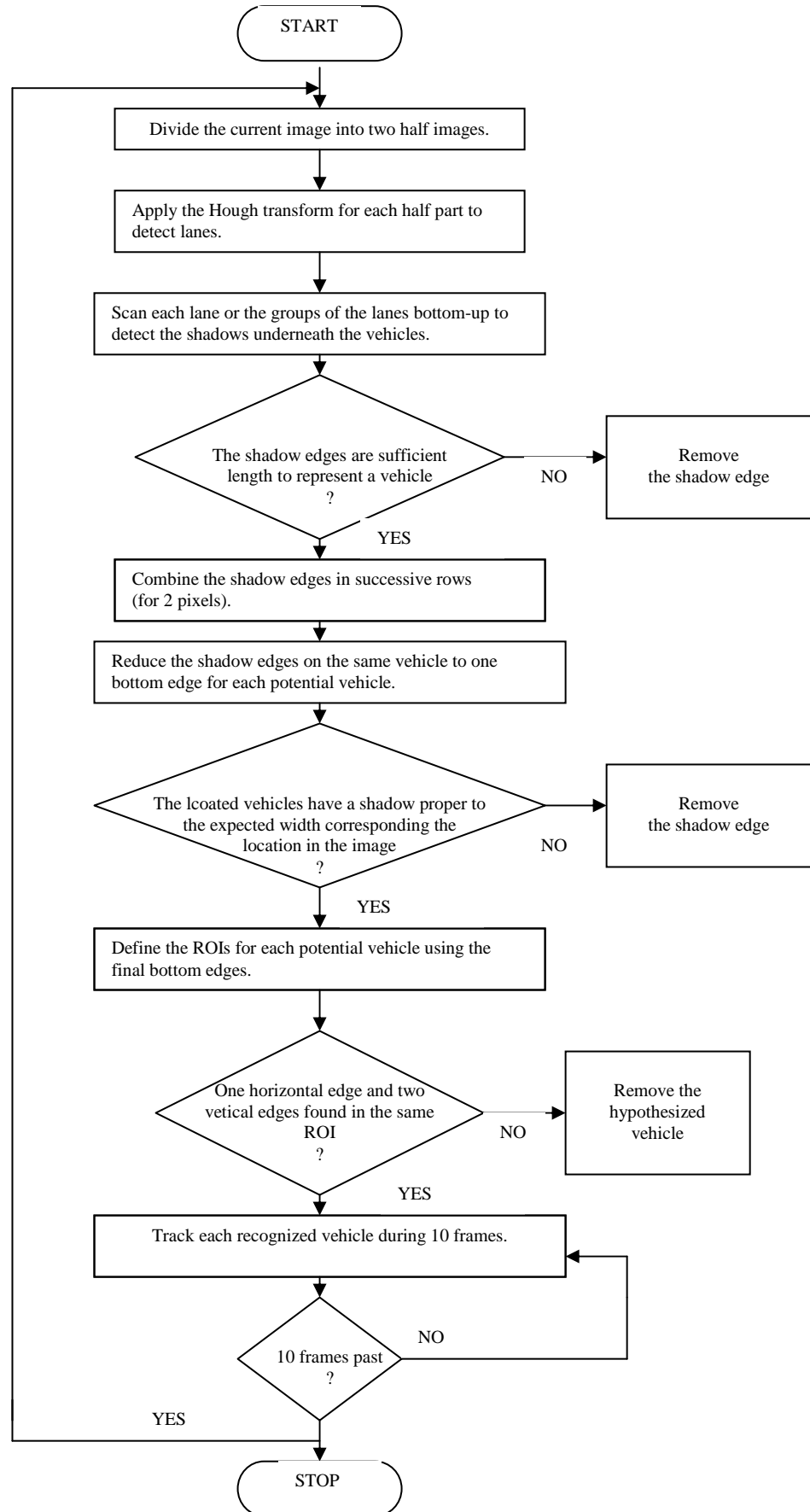


Figure 4.2 : The flow chart of the implemented algorithms.

Despite the problems related to the lane information, it should be taken into account that the lane information in which the observed vehicle is moving is an important parameter. In the case that the shadows underneath vehicles are used as a cue for the detection, the lane information is especially important due to the change in the shadow length during different weather conditions and times of the day, as mentioned in the previous paragraphs. Hence extracting the lane information might require in the vehicle detection and tracking applications. If the dynamics of the lanes due to the moving camera is known, developing a lane tracking algorithm as a solution to the problems related to the lane information might be reasonable in some circumstances within the applications the lane information is needed.



Figure 4.3 : Low sun from the side makes that vehicles cast long shadows.

In Appendix A, the experimental results of the implemented algorithms can be seen. The developed algorithms are executed for the images of the daylight. The detected vehicles are tracked during the frames of an image sequence. Each vehicle is represented by a different color. The aspect ratio of any vehicle is assumed to be 1 and the bounding boxes of the recognized vehicles are plotted based on this aspect ratio. It is also possible to modify the algorithms to detect and track the vehicles in the night, as a future work.

In Figure A.1, the mid-range and the distant vehicles are detected and tracked during the frames of an image sequence. In these frames, it is also possible to observe the performance of the developed algorithms in detecting and tracking the vehicles which make a lane change maneuver.

In Figure A.2, the vehicles at close range are detected and tracked during an image sequence. In these frames, the host vehicle is approaching to another vehicle from the rear. This image sequence is quite usable to illustrate a dangerous situation. In such a case, estimating Time-to-Collision will make the driver to be warned about the distance of the vehicle from the leading vehicle and can make the driver take an action for avoiding a possible collision.

In Figure A.3, a drawback of the shadow-based algorithm is illustrated. The shadow an overpass occurs on the road causes false detections. The area underneath the vehicle is still distinctly darker than any other areas underneath the overpass. Thus, the shadow underneath the vehicle can be detected when the vehicle passes underneath the overpass.

However, the ROIs in which no vehicles exist can not be eliminated using the vertical edges as a clue for verification, as seen in the following frames. Using a combination of different clues in the hypothesis verification step might prevent such false detections. In the illustrated frames, using a combination of vertical edges and texture pattern might be considered as a solution to the mentioned problem.

As a future objective, the 2D-vehicle velocities provided by the algorithms implemented within the thesis are intended to be used for estimating parameters of the (3D) real-world motion of the vehicles relative to the host vehicle with the aim of preventing possible dangerous situations.

Providing the information about the driving environment for drivers makes possible to warn about the time it takes for other vehicles to contact with them and thus the situations rear-end collisions might occur or the collisions caused by sudden lane change might be avoided by estimating Time-to-Collision.

REFERENCES

- [1] **Bertozzi, M., Broggi, A., Cellario, M. and Fascioli, A.**, 2002. Artificial Vision in Road Vehicles, *Proceedings of IEEE* , vol. **90**, pp. 1258-1271.
- [2] **Bishop, R.**, 2000. Intelligent Vehicle Applications Worldwide, *IEEE Intelligent Systems*, vol. **15**, pp. 78-81
- [3] **Heimes, F. and Nagel, H.**, 2002. Towards Active Machine-Vision-Based Driver Assistance for Urban Areas, *International Journal of Computer Vision*, vol. **50**, pp. 5-34.
- [4] **Franke, U. et al.**, 2001. From Door to Door – Principles and Applications of Computer Vision for Driver Assistant Systems, chapter 6 in *Intelligent Vehicle Technologies*, eds. L. Vlacic and F. Harashima and M. Parent, Butterworth Heinemann, Oxford, UK, pp. 131-188.
- [5] **Graefe, V.**, 1993. Vision for Intelligent Road Vehicles, *Proceedings of IEEE Symposium on Intelligent Vehicles*, Tokyo, pp. 135-140.
- [6] **Dickmanns, E.**, 2002. The Development of Machine Vision for Road Vehicles in the Last Decade, *Proceedings of IEEE Intelligent Vehicle Symposium*, vol. **1**, pp. 268-281.
- [7] **Bertozzi, M. and Broggi, A.**, 1998. Gold: A Parallel Real-Time Stereo Vision System for Generic Obstacle and Lane Detection, *IEEE Trans. Image Processing*, vol. **7**, pp. 62-81.
- [8] **Bertozzi, M., Broggi, A. and Fascioli, A.**, 1997. Obstacle and Lane Detection on Argo Autonomous Vehicle, *IEEE Intelligent Transportation Systems*, pp. 1010-1015.
- [9] **Tsugawa, S. and Sadayuki**, 1994. Vision-Based Vehicle on Japan: Machine Vision Systems and Driving Control Systems, *IEEE Trans. Industrial Electronics*, vol. **41**, pp. 398-405.
- [10] **Thorpe, C., Carlson, J.D., Duggins, D., Gowdy, J., MacLachlan, R., Mertz, C., Suppe, A. and Wan, C.**, 2003. Safe Robot Driving in Cluttered Environments, *Proceedings of 11th International Symposium of Robotics Research*, Siena, Italy.
- [11] **Thorpe, C. and Kanade, T.**, 1985. Vision and Navigation for Carnegie-Mellon Navlab, *Proceedings of DARPA Image Understanding Workshop*.
- [12] **Thorpe, C., Hebert, M., Kanade, T. and Shafer, S.**, 1988. Vision and Navigation for Carnegie-Mellon Navlab, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. **10**, pp. 362-373.
- [13] **Kuehnle, A.**, 1991. Symmetry-Based Recognition of Vehicle Rears, *Pattern Recognition Letters*, vol. **12**, pp. 249-258.

- [14] **Zielke, T., Brauckmann, M. and von Seelen, W.,** 1993. Intensity and Edge-Based Symmetry Detection with an Application to Car-Following, *Computer Vision, Graphics, and Image Processing: Image Understanding*, vol. **58**, pp. 177-190.
- [15] **Bertozzi, M., Broggi, A. and Fascioli, A.,** 2000. Vision-Based Intelligent Vehicles: State of the Art and Perspectives, *Robotics and Autonomous Systems*, vol. **32**, pp. 1-16.
- [16] **Crisman, J. and Thorpe, C.,** 1988. Color Vision for Road Following, *Proceedings of SPIE Conf. Mobile Robots*, Cambridge, Massachusetts, pp. 246-249.
- [17] **Buluswar, S.D. and Draper, B.A.,** 1998. Color Machine Vision for Autonomous Vehicles, *International Journal of Engineering Applications of Artificial Intelligence*, vol. **1**, no. 2, pp. 245-256.
- [18] **Guo, D., Fraichard, T., Xie, M. and Laugier, C.,** 2000. Color Modelling by Spherical Influence Field in Sensing Driving Environments, *Proceedings of IEEE Intelligent Vehicles Symposium*, Dearborn, Mi. Usa, pp. 249-254.
- [19] **Dellaert, F. and Thorpe, C.,** 1997. Robust Car Tracking using Kalman Filtering and Bayesian Templates, *Proceedings of SPIE Conf. Intelligent Transportation Systems*, vol. **3207**, pp. 17-83.
- [20] **Betke, M., Haritaoglu, E. and Davis, L.S.,** 2000. Real-time multiple vehicle detection and tracking from a moving vehicle, *Machine Vision and Applications*, vol. **12**, no. 2, pp. 69-83.
- [21] **Kalinke, T., Tzomakas, C. and von Seelen, W.,** 1998. A Texture-Based Object Detection and Adaptive Model-Based Classification, *Proceedings of IEEE Intelligent Vehicles Symposium*, Stuttgart, Germany, pp. 143-148.
- [22] **Haralick, R., Shanmugam, B. and Dinstein, I.,** 1973. Texture Features for Image Classification, *IEEE Trans. System, Man, Cybernetics*, vol. **3**, pp. 610-621.
- [23] **Kim, S. and Kim, K et al,** 2005. Front and Rear Vehicle Detection and Tracking in the Day and Night Times using Vision and Sonar Sensor Fusion, *Intelligent Robots and Systems, IEEE/RSJ International Conference*, Alberta, Canada, pp. 2173-2178.
- [24] **Franke, U. and Kutzbach, I.,** 1996. Fast Stereo based Object Detection for Stop&Go Traffic, *Proceedings of IEEE Intelligent Vehicles Symposium*, Tokyo, Japan, pp. 339-344.
- [25] **Zhao, G. and Yuta, S.,** 1993. Obstacle Detection by Vision System for An Autonomous Vehicle, *Intelligent Vehicles*, pp. 31-36.
- [26] **Giachetti, A., Campani, M. and Torre, V.,** 1998. The use of optical flow for road navigation, *IEEE Trans. On Robotics and Automation*, vol. **14**, no. 1, pp. 34-48.

- [27] **Morimoto, C., DeMenthon, D., Davis, L.S., Chellappa, R. and Nelson, R.C.,** 1995. Detection of independently moving objects in passive video, *Proceedings of IEEE Intelligent Vehicles Symposium*, Detroit, Michigan, pp. 270-275.
- [28] **Sun, Z., Bebis, G. and Miller, R.,** 2006. On-Road Vehicle Detection: A Review, *IEEE Transactions on pattern analysis and machine intelligence*, vol. **28**, no. 5, pp. 694-711.
- [29] **Erçil, A., Abut, H., Erzin, E., Göçmençelebi, A., Göktan, A., Güvenç, L., Özatay, E. and Tandoğdu, H.,** 2005. The drivesafe project, *Proceedings of the 1st AUTOCOM Workshop on Preventive and Active Safety for Road Vehicles*, İstanbul.
- [30] **Daniş, S., Aytekin, B., Dinçmen, E., Sezer, V., Ararat, Ö., Öncü, S., Güvenç, B.A., Acarman, T., Altuğ, E. and Güvenç, L.,** 2008. Framework for Development of Driver Adaptive Warning and Assistance Systems That Will Be Triggered by A Driver Inattention Monitor, *Otekon'08 4th Automotive Technologies Congress*, Bursa.
- [31] **Lundagards, M.,** 2008. *Vehicle Detection in Monochrome Images*, M.Sc. Thesis. Linköping University.
- [32] **Gonzalez, R.C., Woods, R.E. and Eddins, S.L.,** 2004. *Digital Image Processing Using Matlab*, Pearson Prentice Hall Press.
- [33] **Nixon, M. and Aguado, A.,** 2002. *Feature Extraction and Image Processing*, Butterworth Heinemann, Oxford.
- [34] **Tzomakas, C. and Seelen, W.,** 1998. Vehicle Detection in Traffic Scenes Using Shadows, *Technical Report 98-06*, Institut für Neuroinformatik, Ruht-Universität, Bochum, Germany.
- [35] **Liu, W., Wen, X., Duan, B., Yuan, H. and Wang, N.,** 2007. Rear Vehicle Detection and Tracking for Lane Change Assist, *Proceedings of IEEE Intelligent Vehicles Symposium*, İstanbul, pp. 252-257.
- [36] **Yılmaz, A., Javed, O. and Shah, M.,** 2006. Object Tracking: A Survey, *ACM Computing Surveys*, vol. **38**, no. 4.
- [37] **Srinivasa, N.,** 2002. Vision-based Vehicle Detection and Tracking Method for Forward Collision Warning in Automobiles, *Proceedings of IEEE Intelligent Vehicle Symposium*, vol. **2**, pp. 626-631.
- [38] **Broggi, A., Cerri, P. and Ghidoni, S.,** 2005. A Correlation-Based Approach to Recognition and Localization of the Preceding Vehicle in Highway Environments, *International Conference on Image Analysis and Processing*, vol. **3617**, pp. 1166-1173.
- [39] **Dellaert, F. and Thrope, C.,** 1997. Robust Car Tracking Using Kalman Filtering and Bayesian Templates, *Proceedings of SPIE, Intelligent Transportation Systems*, vol. **3207**, pp. 72-83.
- [40] **Liu, X.,** 2000. *Development of A Vision-Based Object Detection and Recognition System for Intelligent Vehicle*, Ph.D. Thesis. University of Wisconsin – Madison.

- [41] **Leeuwen, van MB.**, 2002. *Motion Estimation and Interpretation for In-Car Systems*, Ph.D. Thesis. University of Amsterdam.
- [42] **Welch, G. and Bishop, G.**, 2001. An Introduction to the Kalman Filter, Lecture Notes. University of North Carolina, Department of Computer Science.
- [43] **Cuevas, E., Zaldivar, D. and Rojas, R.**, 2005. Kalman filter for vision tracking, *Technical Report B 05-12*, Freie Universität Berlin, Fachbereich Mathematik und Informatik.
- [44] **Schwartz, M. and Shaw, L.**, 1975. *Signal Processing: Discrete Spectral Analysis, Detection, and Estimation*, McGRAW-HILL International Book Company.

APPENDICES

APPENDIX A : Experimental results of the implemented algorithms.

APPENDIX A



Figure A.1 : Detection and tracking of mid-range and distant vehicles.



Figure A.1 (contd.) : Detection and tracking of mid-range and distant vehicles.

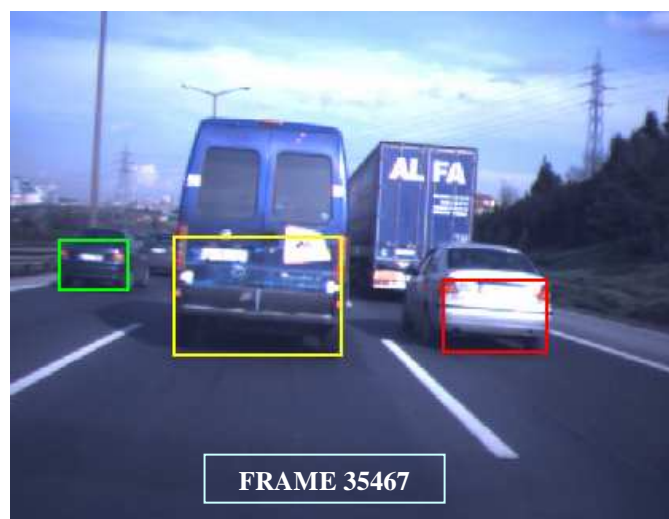
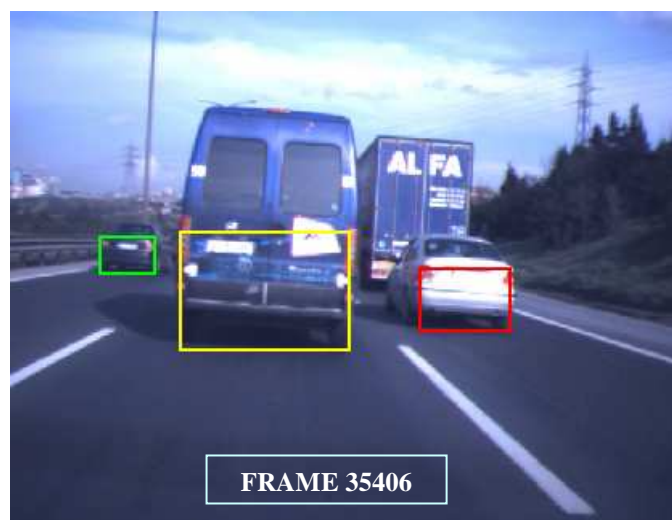
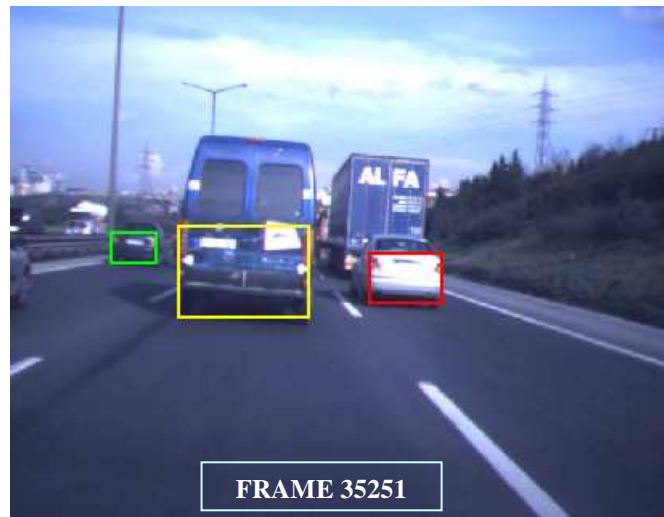


Figure A.2 : Detection and tracking of the vehicles at close range.

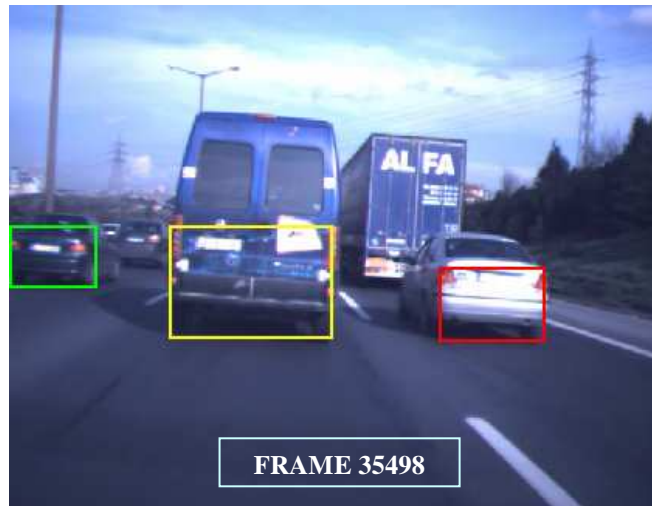


Figure A.2 (contd.) : Detection and tracking of the vehicles at close range.

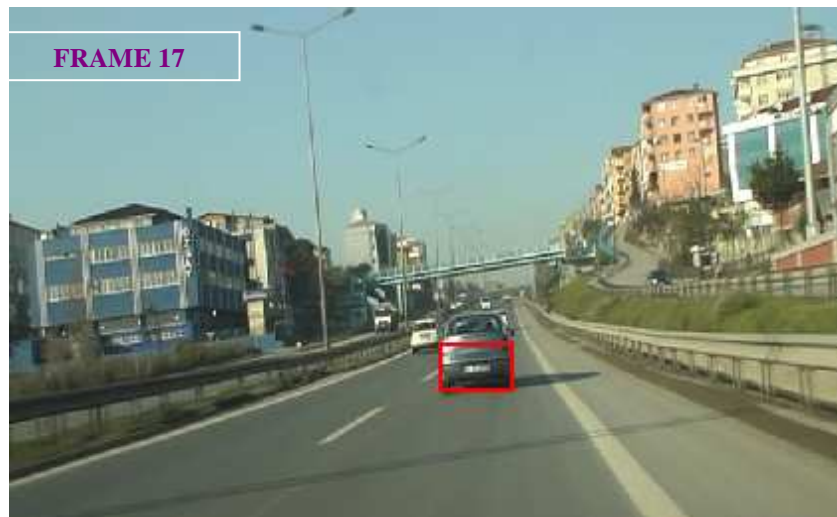


Figure A.3 : Detection and tracking of the vehicle in the situation where an overpass occurs shadow areas on the road.

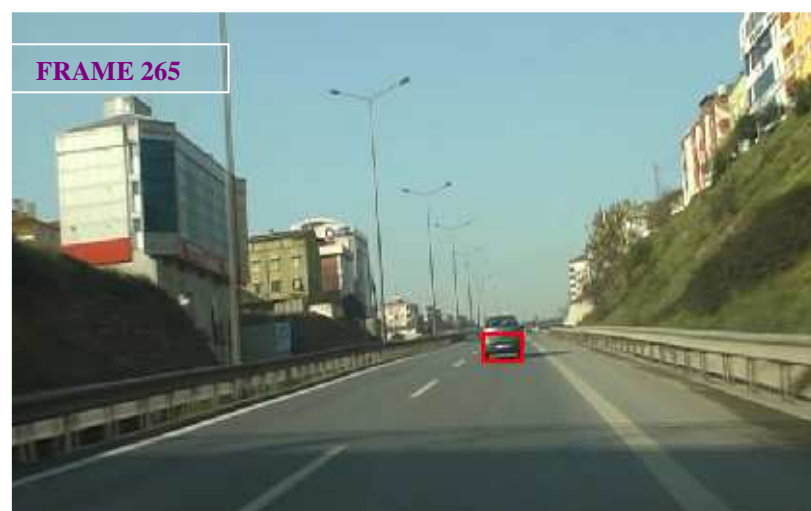
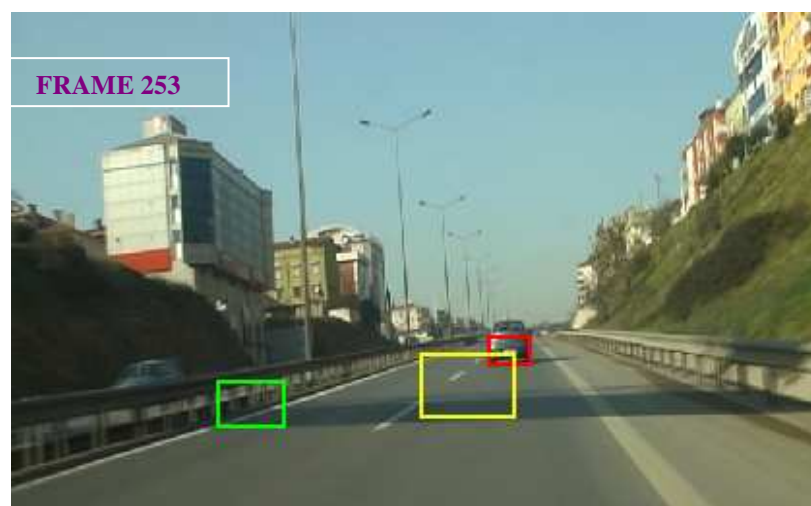


Figure A.3 (contd.) : Detection and tracking of the vehicle in the situation where an overpass occurs shadow areas on the road.

CURRICULUM VITA

Candidate's full name: Burcu AYTEKİN

Place and date of birth: İstanbul, 23.07.1982

Permanent Address: Şehitler Caddesi, Güldeniz Sitesi, No: 73/ 2, Tuzla/ İstanbul

Universities and

Colleges attended: Kadir Has High School and Kocaeli University

Publications:

- **Daniş, S., Aytekin, B., Dinçmen, E., Sezer, V., Ararat, Ö., Öncü, S., Güvenç, B.A., Acarman, T., Altuğ, E. and Güvenç, L.,** 2008. Framework for Development of Driver Adaptive Warning and Assistance Systems That Will Be Triggered by A Driver Inattention Monitor, *Otekon'08 4th Automotive Technologies Congress*, Bursa.
- **Aytekin, B., Altug, E.,** 2009. Bilgisayarlı Görü Yöntemi ile Araç Belirleme ve Takibi, *Submitted to IEEE 17. Sinyal İşleme ve İletişim Uygulamaları Kurultayı.*