

Traffic and Road Sign Recognition

Hasan Fleyeh

This thesis is submitted in fulfilment of the requirements of
Napier University for the degree of
Doctor of Philosophy

July 2008

Abstract

This thesis presents a system to recognise and classify road and traffic signs for the purpose of developing an inventory of them which could assist the highway engineers' tasks of updating and maintaining them. It uses images taken by a camera from a moving vehicle. The system is based on three major stages: colour segmentation, recognition, and classification.

Four colour segmentation algorithms are developed and tested. They are a shadow and highlight invariant, a dynamic threshold, a modification of de la Escalera's algorithm and a Fuzzy colour segmentation algorithm. All algorithms are tested using hundreds of images and the shadow-highlight invariant algorithm is eventually chosen as the best performer. This is because it is immune to shadows and highlights. It is also robust as it was tested in different lighting conditions, weather conditions, and times of the day. Approximately 97% successful segmentation rate was achieved using this algorithm.

Recognition of traffic signs is carried out using a fuzzy shape recogniser. Based on four shape measures - the rectangularity, triangularity, ellipticity, and octagonality, fuzzy rules were developed to determine the shape of the sign. Among these shape measures octangonality has been introduced in this research. The final decision of the recogniser is based on the combination of both the colour and shape of the sign. The recogniser was tested in a variety of testing conditions giving an overall performance of approximately 88%.

Classification was undertaken using a Support Vector Machine (SVM) classifier. The classification is carried out in two stages: rim's shape classification followed by the classification of interior of the sign. The classifier was trained and tested using binary images in addition to five different types of moments which are Geometric moments, Zernike moments, Legendre moments, Orthogonal Fourier-Mellin Moments, and Binary Haar features. The performance of the SVM was tested using different features, kernels, SVM types, SVM parameters, and moment's orders. The average classification rate achieved is about 97%. Binary images show the best testing results followed by Legendre moments. Linear kernel gives the best testing results followed by RBF. C-SVM shows very good performance, but v-SVM gives better results in some case.

Acknowledgements

First and foremost, I would like to thank my supervisors Professor Howard Kirby and Dr. Chuen Wan. Their suggestions, guidance, encouragement, and support helped me greatly with this thesis.

I would like to gratefully acknowledge the support of Dalarna University, Sweden which funded and financially supported this work from the very beginning.

I must especially mention Professor Mark Dougherty, Dean of Research, Dalarna University, my friend and supporter who suggested the cooperation between Dalarna University and Napier University, and who supported me from my first trip to Edinburgh to the present day. For his valuable time which is given to me freely and for the discussions and criticisms which make this work what it is today, I would like to say thank you.

My thanks go to my friends Yella Siril and Pascal Rebreyend at Dalarna University for the useful discussions, help and support.

I would like to recognise the help and support of Dr. Ernst Nordström, Computer Engineering Department, Dalarna University and Prof. Björn Sohlberg, Electrical Engineering Department, Dalarna University for reviewing my papers, and for the valuable discussions and encouragement.

My acknowledgements go to Mrs. Sarah Berglind, English Department, Dalarna University and Mr. Jalal Alrazzaz who proof-read this thesis. Without their suggestions the thesis would not be what it is today.

Most of all, I would like to express my great thanks to the person who is closest to my heart, my wife Jinan Hussein who has shared the past twenty years of my life both in its sweet and sad days; who helped me during this thesis by collecting the traffic sign images; and, by taking on more domestic responsibilities to provide a quiet environment for my research. Without her love, support, and encouragement, this thesis would not have been possible.

My thanks go to my daughter Riam and my son Sadeem for their continual love and support.

To Jinan
My whole life...
and more.

TABLE OF CONTENTS

LIST OF FIGURES	X
LIST OF TABLES	XIV
1. Introduction.....	1
1.1 Background.....	1
1.2 Complexity of the Recognition Task	3
1.3 Aims and Objectives of the Research	4
1.4 Outline of the Thesis.....	4
2. Traffic Signs	7
2.1 Properties of Road and Traffic Signs.....	7
2.2 Swedish Road and Traffic Signs.....	8
2.3 Summary	11
3. Problem Formulation	14
3.1 What is Road Sign Recognition?	15
3.2 Road Sign Recognition Applications.....	16
3.3 Potential Difficulties	17
3.4 Summary	24
4. Technical Overview	25
4.1 Literature Review.....	25
4.1.1 Colour-Based Detection of Traffic Signs	26
4.1.2 Shape-Based Detection of Traffic Signs.....	27
4.1.3 Colour-Shape-Based Detection of Road Signs	30
4.1.4 Recognition and Classification	34
4.1.5 OCR and Pictograms Recognition	39
4.1.6 Analysis of the Literature Review	40
4.2 Colour	47
4.2.1 Variation of Colour in Outdoor Images	49
4.2.2 Hue Invariance	51
4.2.3 Colour Constancy.....	52

4.3 Moments and Invariants.....	53
4.3.1 Zernike Moments	53
4.3.2 Legendre Moments	57
4.3.3 Orthogonal Fourier-Mellin Moments	59
4.3.4 Binary Haar Features	60
4.4 Support Vector Machines	62
4.4.1 Linear Classification with Maximal Margin Classifier	63
4.4.2 Non-Linear Classification.....	69
4.4.3 Learning in Feature Space	70
4.4.4 Implicit Mapping to Feature Space.....	71
4.4.5 Kernels	72
4.4.6 Properties of Kernels.....	73
4.4.7 Examples of Kernels	73
4.4.8 Soft Margin Classifier.....	74
4.4.9 Multi-class Classifier	75
4.4.10 Types of SVM.....	76
4.5 Summary	77
5. Road Sign Recognition System Design.....	79
5.1 System Overview	79
5.2 The Camera.....	82
5.3 The Raw Images Database.....	83
5.4 Colour Segmentation Algorithms	86
5.4.1 The Dynamic Threshold Algorithm.....	87
5.4.2 A Modification of de la Escalera's Algorithm.....	89
5.4.3 The Fuzzy Colour Segmentation Algorithm.....	91
5.4.4 Shadow and Highlight Invariant Algorithm	96
5.4.5 Colour Segmentation in Poor Light Conditions	105
5.5 Recognition by Combining Colours and Shapes	107
5.5.1 Shape Measures	111
5.5.2 Fuzzy Shape Recogniser.....	112
5.6 The Algorithm.....	117

5.7 Training Database of the SVM Classifier.....	119
5.8 Classification with SVM.....	123
5.8.1 Classification using Binary Images	123
5.8.2 Classification with Moments and Features	126
5.9 Summary	130
6. Results Analysis.....	131
6.1 Colour Segmentation Algorithms	131
6.1.1 Performance Evaluation.....	131
6.1.2 Failure Analysis	136
6.2 The Recognition Stage	138
6.2.1 Performance Evaluation.....	138
6.2.2 Failure Analysis	139
6.3 The Classification Stage	142
6.3.1 Classification with Different Features	142
6.3.2 Classification with Different Kernels and SVM Types	144
6.3.3 Performance of SVM with Different Parameters.....	147
6.3.4 Classification with Different Moment Orders	151
6.3.5 Classification Time	154
6.3.6 Search for Optimal Parameters	155
6.4 Summary	162
7. Conclusions.....	163
7.1 Collection of Traffic Sign Images.....	163
7.2 Colour Segmentation Algorithms	163
7.3 Colour Segmentation in Poor Light Conditions	164
7.4 Octagonality as a New Shape Measure.....	165
7.5 The Fuzzy Shape Recogniser.....	165
7.6 SVM for Traffic Sign Recognition	166
7.7 Future Directions	166
7.7.1 Occluded Signs and Object Recognition	166
7.7.2 Detachment of Signs	167
7.7.3 Similarity Measures for Sign Detection.....	168

7.7.4 Real Time Applications	168
7.8 Final Remarks	169
References.....	170
Appendix A - The Swedish Road and Traffic Signs	181
Appendix B - Colour Physics and Colour Spaces	197
B.1 Introduction	197
B.2 Model of the Eye.....	198
B.3 Colour Spaces.....	199
B.3.1 Device dependent and device independent Colour Space.....	199
B.3.2 Colour Gamut.....	200
B.3.3 Terminology.....	200
B.3.4 CIE XYZ Colour Space.....	201
B.3.5 The RGB Colour Space.....	204
B.3.6 The CMY and CMYK Colour Space	206
B.3.7 The Nrgb Colour Space.....	207
B.3.8 HSV (HSB) Colour Space.....	207
B.3.8.1 Converting Colours from RGB to HSV	212
B.3.8.2 Converting Colours from HSV to RGB	212
B.3.9 The HSI (HLS) Colour Space	214
B.3.9.1 Converting Colours from RGB to HSI.....	215
B.3.9.2 Converting colours from HSI to RGB.....	216
B.3.10 Comparison of HSI (HSL) and HSV.....	218
B.3.11 The Improved HLS colour space	219
B.3.11.1 Converting Colours from RGB to IHLS	219
B.3.11.2 Converting Colours from IHLS to RGB	219
B.3.12 The YIQ Colour Space.....	220
B.3.13 The YUV Colour Space	222
B.3.14 The YCbCr Colour Space	223
B.3.15 The L*a*b* Colour Space.....	224
B.3.16 The L*u*v* Colour Space.....	226
Appendix C – Explicit Form of Zernike Polynomials	227

Appendix D – Results of Training and Testing of SVM.....	229
Appendix E – The Access Database	232
Appendix F – List of Publications	238

LIST OF FIGURES

Figure 1.1: The relationship among Road Sign Inventory, Road Sign Recognition and ITS.....	2
Figure 1.2: A traffic scene in the middle of Stockholm.....	3
Figure 2.1: Warning signs.....	9
Figure 2.2: Prohibitory signs.....	9
Figure 2.3: Mandatory signs	10
Figure 2.4: Indicator and Supplementary signs	10
Figure 3.1: A block diagram of the road sign recognition and classification.....	16
Figure 3.2: Faded signs.....	18
Figure 3.3: Bad weather conditions (Rain and Snow).	18
Figure 3.4: Bad lighting geometry.....	19
Figure 3.5: The presence of obstacles in the scene.....	19
Figure 3.6: Similar objects in the scene or similar background colour.	20
Figure 3.7: Damaged signs.	20
Figure 3.8: Size of signs depends on the distance between the camera and the sign.	21
Figure 3.9: Motion blur problem.	21
Figure 3.10: Reflection from sign board.....	22
Figure 3.11: Colours in different countries: left, Netherlands; right, Sweden.....	22
Figure 3.12: Different pictograms are used in different countries.....	23
Figure 3.13: Stickers which damaged the pictogram of a sign.	24
Figure 4.1: Traffic scene model.	48
Figure 4.2: The CIE diagram.	50
Figure 4.3: Main steps to perform translation and scaling invariance.....	57
Figure 4.4: Binary Haar features applied to different road signs.....	61
Figure 4.5: (a) Pattern recognition problem with two classes. (b)(c) Separating hyperplane separates the two classes without training errors. (d) Optimal separating hyperplane.....	64
Figure 4.6: Optimal solution can be obtained by maximizing the margin.....	64
Figure 4.7: Linear classification for two-dimensional input vectors.	65

Figure 4.8: A non-linear problem. It is impossible to linearly separate the patterns in the input space.	70
Figure 4.9: A mapping from a two-dimensional input space to a two-dimensional feature space.....	71
Figure 4.10: An example of soft margin classifier.	75
Figure 5.1: A block diagram of the RSRS.	80
Figure 5.2: Minolta DiMAGE 7Hi.....	84
Figure 5.3: The vector model of the Hue and Saturation.....	87
Figure 5.4: Results of colour segmentation using dynamic threshold algorithm applied for Red, Yellow and Green signs.....	89
Figure 5.5: Saturation transfer function.	90
Figure 5.6: Hue transfer function of Red.	90
Figure 5.7: Hue transfer function of Green.....	90
Figure 5.8: Hue transfer function of Blue.	90
Figure 5.9: Results of applying the modified version of de la Escalera algorithm.....	91
Figure 5.10: Hue membership functions.....	92
Figure 5.11: Saturation membership functions.....	93
Figure 5.12: The Output functions.....	93
Figure 5.13: The fuzzy system surface.	94
Figure 5.14: Segmentation results from fuzzy colour segmentation algorithm.	95
Figure 5.15: Children signs from different European countries.	95
Figure 5.16: Flowchart of shadow and highlight invariant algorithm.	103
Figure 5.17: Results from shadow and highlight invariant algorithm.	104
Figure 5.18: Block diagram of the colour segmentation algorithm for poor light conditions.....	106
Figure 5.19: Results of applying colour segmentation in poor light.....	107
Figure 5.20: Colour combinations of traffic signs.	108
Figure 5.21: Traffic sign tree.	109
Figure 5.22: Occlusions and Convex hull. First row: Occluded objects. Second row: Results of Segmentation and Extraction. Third row: Convex hull.	113
Figure 5.23: The R1 Membership Functions.	114

Figure 5.24: The R2 Membership Functions	115
Figure 5.25: The T Membership Functions	115
Figure 5.26: The E Membership Functions	115
Figure 5.27: The O Membership Functions	115
Figure 5.28: Output Membership Functions.....	116
Figure 5.29: Examples of sign recognition.....	119
Figure 5.30: A traffic sign is a rim and a pictogram.....	119
Figure 5.31: Classification is carried out by rim classifier and pictogram classifier.....	119
Figure 5.32: Convex hull is used to preserve objects details.....	121
Figure 5.33: Part of the training database and their corresponding categories of traffic sign rims.....	122
Figure 5.34: Part of the training database and their corresponding categories of Speed-Limit signs.	123
Figure 6.1: Comparison of quality of segmentation of two different algorithms.	135
Figure 6.2: The effect of fog and how it can be treated.	136
Figure 6.3: Instability of segmentation with yellow.	137
Figure 6.4: Yellow Instability because of low illumination.	137
Figure 6.5: False Positives.	140
Figure 6.6: Sensitivity to connected signs.	142
Figure 6.7: The effect of parameter C on the classification accuracy of the SVM classifier when C-SVM and Linear kernel are used.....	148
Figure 6.8: Number of support vectors versus parameter ν of SVM model using ν -SVM model and linear kernel.....	148
Figure 6.9: Performance of the SVM model using Linear kernel and different values of parameter ν	149
Figure 6.10: Performance of the SVM model using C-SVM, RBF kernel, $C = 1$ and different values of parameter γ	149
Figure 6.11: Performance of the SVM model using C-SVM, Sigmoid kernel, $C = 1$, $\gamma = 0.1$ and different values of parameter r	150
Figure 6.12: Performance of the SVM model using C-SVM, polynomial kernel, $C = 1$, $\gamma = 1$, $r = 0$ and different values of parameter d	151

Figure 6.13: Effect of the order of Geometric moments order on classification rate when C is constant.....	152
Figure 6.14: Effect of the order of Zernike moments on classification rate when C is constant	152
Figure 6.15: Effect of the order of Legendre moments on classification rate when C is constant	153
Figure 6.16: Effect of the order of OFMM moments on classification rate when C is constant	153
Figure 6.17: Effect of $(\Delta x + \Delta y)$ of Haar Features on classification rate when C is constant	154
Figure 7.1: Occlusions produced by the existence of obstacles.....	167
Figure 7.2: Traffic signs may be seen as one object after segmentation.	168

LIST OF TABLES

Table 2.1: Swedish standard colours.	12
Table 2.2: The meaning of colours used for traffic signs.	12
Table 2.3: Shapes of traffic signs and their meanings.	13
Table 5.1: Specifications of Minolta DiMAGE 7Hi.....	83
Table 5.2: Raw images database comprises different categories.....	85
Table 5.3: Normalised Hue and Saturation.....	86
Table 5.4: The effect of imaging conditions on invariance of colours. ‘Y’ denotes invariance and ‘N’ denotes sensitivity of colour models to imaging conditions.....	101
Table 5.5: Shape Measures of Stop Sign.	116
Table 5.6: Shape Measures of Yield Sign.....	116
Table 5.7: Shape Measures of No-Entry Sign	116
Table 5.8: Shape Measures of Rectangular Signs.....	116
Table 5.9: Confusion matrix of training set of Rims with binary images.....	124
Table 5.10: Confusion matrix of test set of Rims with binary images.	124
Table 5.11: Confusion matrix of training set of Speed-Limit with binary images.	125
Table 5.12: Confusion matrix of testing set of Speed-Limit with binary images.	125
Table 5.13: Speed-Limit images which are incorrectly classified.	125
Table 5.14: Traffic signs rims and their categories.....	126
Table 5.15: Confusion matrix of test set of Rims with Geometric moments.	127
Table 5.16: Confusion matrix of test set of Rims with Zernike moments.	127
Table 5.17: Confusion matrix of test set of Rims with Legendre moments.	128
Table 5.18: Confusion matrix of test set of Rims with OFMM moments.	128
Table 5.19: Confusion matrix of test set of Rims with Haar features.....	128
Table 5.20: Confusion matrix of test set of Speed-Limit with Geometric moments.....	128
Table 5.21: Confusion matrix of test set of Speed-Limit with Zernike moments.	129
Table 5.22: Confusion matrix of test set of Speed-Limit with Legendre moments.....	129
Table 5.23: Confusion matrix of test set of Speed-Limit with OFMM moments.....	129
Table 5.24: Confusion matrix of test set of Speed-Limit with Haar Features.	129
Table 6.1: Segmentation success rate (%) of different colour segmentation algorithms.	132

Table 6.2: Segmentation success rate (%) of different algorithms tested under different effects.....	133
Table 6.3: A comparison of the processing Time of different colour segmentation algorithms.....	134
Table 6.4: Average number of objects generated by different segmentation algorithms.	135
Table 6.5: Recognition rates of traffic signs in different test conditions.....	138
Table 6.6: Time of recognition.	139
Table 6.7: Speed of recognition is affected by size of the sign.	141
Table 6.8: Average classification rate of traffic sign rims with different features.	142
Table 6.9: Average classification rate of speed-limit signs with different features.....	143
Table 6.10: Default values of γ computed for different features.....	143
Table 6.11: Classification rates of sign rims and Speed-Limit signs using different kernels and SVM types when binary images are used.	144
Table 6.12: Classification rates of sign rims and Speed-Limit signs using different kernels and SVM types when Geometric moments are used.	145
Table 6.13: Classification rates of rim shapes and Speed-Limit signs using different kernels and SVM types when Zernike moments are used.	145
Table 6.14: Classification rates of rim shapes and Speed-Limit signs using different kernels and SVM types when Legendre moments are used.	146
Table 6.15: Classification rates of rim shapes and Speed-Limit signs using different kernels and SVM types when OFMM are used.	146
Table 6.16: Classification rates of rim shapes and Speed-Limit signs using different kernels and SVM types when Binary Haar Features are used.	147
Table 6.17: A comparison of training and testing times using different features and normalised images.	154
Table 6.18: Classification rates of grid search for sign rims when C-SVM and ν - SVM are used.....	156
Table 6.19: SVM parameters of grid search for traffic signs rims when C-SVM and ν - SVM are used.....	156

Table 6.20: Classification rates of grid search for Speed-Limit signs when C-SVM and ν -SVM are used.....	157
Table 6.21: SVM parameters of grid search for Speed-Limit signs when C-SVM and ν -SVM are used.....	157
Table 6.22: Classification rates of SA search for sign rims when C-SVM and ν -SVM are used.....	159
Table 6.23: SVM parameters of SA search for traffic signs rims when C-SVM and ν -SVM are used.....	159
Table 6.24: Classification rates of SA search for Speed-Limit signs when C-SVM and ν -SVM are used.....	160
Table 6.25: SVM parameters of SA search for Speed-Limit signs when C-SVM and ν -SVM are used.....	160
Table 6.26: A comparison of times of search of grid and SA for traffic sign rims.	161
Table 6.27: A comparison of times of search of grid and SA search for Speed-Limit...	161

1. Introduction

1.1 Background

Road and traffic signs considered in this thesis are those that use a visual/symbolic language about the road(s) ahead that can be interpreted by drivers. The terms are used interchangeably in this thesis, and elsewhere might also appear in combination, as “road traffic signs”. They provide the driver with pieces of information that make driving safe and convenient. A type of sign that is NOT considered in this thesis is the direction sign, in which the upcoming directions for getting to named towns or on numbered routes are shown not symbolically but essentially by text.

Road and traffic signs must be properly installed in the necessary locations and an inventory of them is ideally needed to help ensure adequate updating and maintenance. Meetings with the highway authorities in both Scotland and Sweden revealed the absence of but a need for an inventory of traffic signs. An automatic means of detecting and recognising traffic signs can make a significant contribution to this goal by providing a fast method of detecting, classifying and logging signs. This method helps to develop the inventory accurately and consistently. Once this is done, the detection of disfigured or obscured signs becomes easier for human operator.

Road and traffic sign recognition is the field of study that can be used to aid the development of an inventory system (for which real-time recognition is not required) or aid the development of an in-car advisory system (when real-time recognition is necessary). Both road sign inventory and road sign recognition are concerned with traffic signs, face similar challenges and use automatic detection and recognition.

A road and traffic sign recognition system could in principle be developed as part of an Intelligent Transport Systems (ITS) that continuously monitors the driver, the vehicle, and the road in order, for example, to inform the driver in time about upcoming decision points regarding navigation and potentially risky traffic situations. Figure 1.1 depicts these relationships among the three fields.

ITS focuses on integrating information technology into transport infrastructure and vehicles. These systems can include road sensors, in-vehicle navigation services,

electronic message signs, and traffic management and monitoring. The aim of intelligent transport systems is to increase transportation efficiency, road safety and to reduce the environmental impact with the use of advanced communication technologies [1, 2].

This thesis aims to develop a system to recognise and classify road and traffic signs for the purpose of developing an inventory which could assist the highway authorities to update and maintain the traffic signs. It is based on taking images by a camera from a moving vehicle and invoking colour segmentation, shape recognition, and classification to detect the signs in these images.

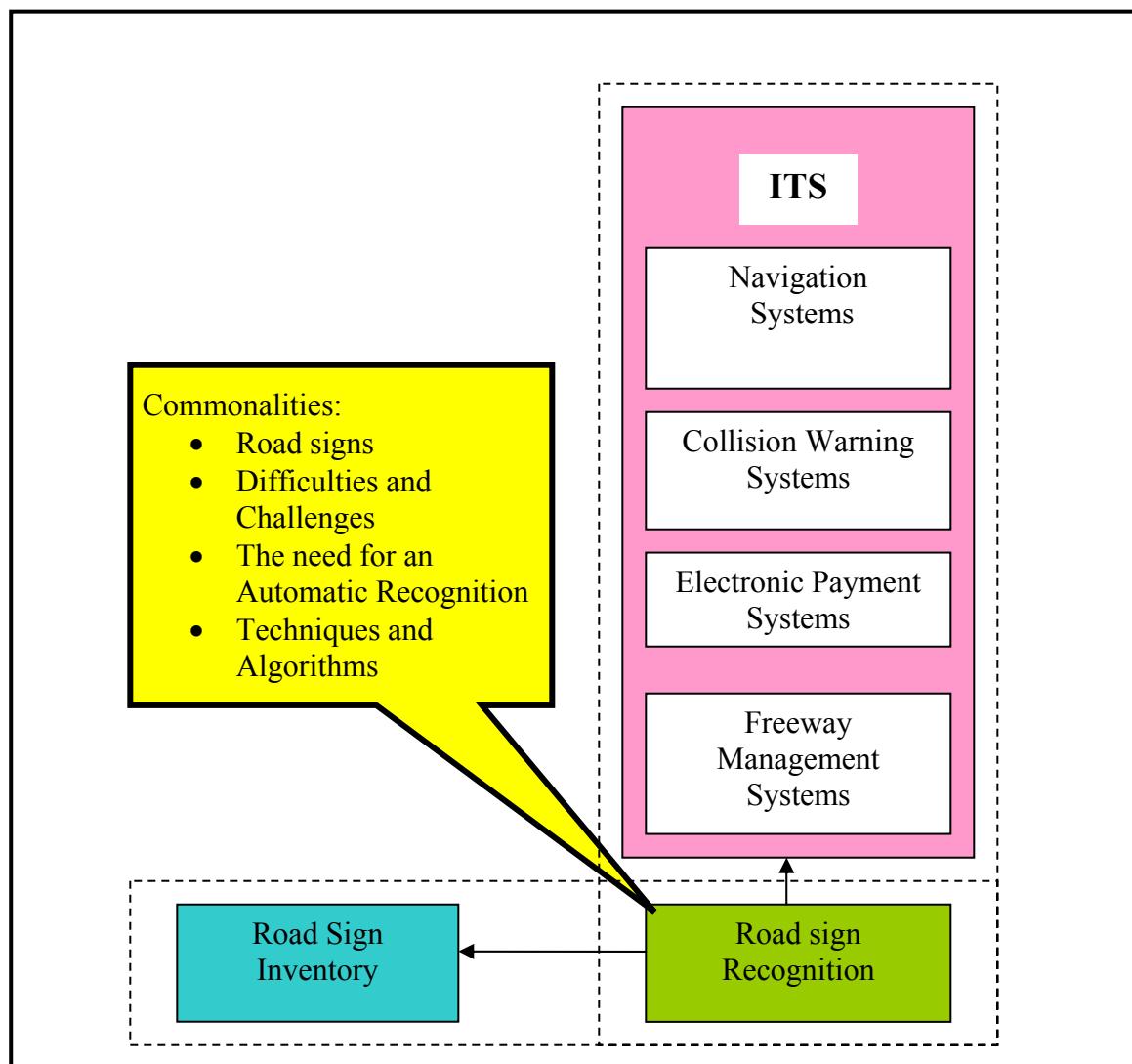


Figure 1.1: The relationship among Road Sign Inventory, Road Sign Recognition and ITS.

1.2 Complexity of the Recognition Task

A normal road in the middle of most cities in the world like the one shown for Stockholm in Figure 1.2, presents a complex scene. It may include people, vehicles with different colours, a number of shops and their signs, and a number of traffic signs to control the traffic on this road. Fundamentally, if a person is asked to point out the traffic sign in the image, they can do this easily.



Figure 1.2: A traffic scene in the middle of Stockholm.

However, from the point of view of computer vision, this image contains some difficulties which are addressed here:

- The existence of a number of similar objects (either in colour or in shape) in the scene.
- The presence of obstacles in the scene which can partially or totally occlude the sign.
- The amount of information in the scene is vast and time is needed to analyse the scene and extract the desired information.

1.3 Aims and Objectives of the Research

The overall aim is to develop a system that can be used for traffic sign inventory. This system can assist local or national authorities in the task of maintaining and updating their road and traffic signs by automatically detecting and classifying one or more traffic signs from a complex scene (like the one shown in Figure 1.2) when captured by a camera from a vehicle.

The main strategy is to find the right combination of colours in the scene so that one colour is located inside the convex hull of another colour and combine this with the right shape. If a candidate is found, the system tries to classify the object according to the rim-pictogram combination and give the result of this classification.

The objectives are thus:

1. To understand the properties of road and traffic signs and their implications for image processing for the recognition task.
2. To understand colour, colour spaces and colour space conversion.
3. To develop robust colour segmentation algorithms that can be used in a wide range of environmental conditions.
4. To develop a recogniser that is invariant to in-plane transformations such as translation, rotation, and scaling based on invariant shape measures.
5. To identify the most appropriate approach for feature extraction from road signs.
6. To develop an appropriate road sign classification algorithm.
7. To evaluate the performance of the aforementioned methods for robustness under different conditions of weather, lighting geometry, and sign.

1.4 Outline of the Thesis

This thesis is divided into seven chapters and a number of appendices. In addition to the current chapter, there are six other chapters that cover different theoretical and practical topics. Each chapter is self-contained, but there are some dependencies among the different chapters. These chapters are as follows:

Chapter 1:

This chapter describes the aims and objectives of this thesis and outlines its relationship to the field of intelligent transport systems

Chapter 2:

This chapter demonstrates the traffic and road signs, their properties, categories and proper colours and shapes. It concentrates on the Swedish traffic and road signs and the differences between the Swedish standard and that of the other countries.

Chapter 3:

This chapter explains why the automation of the categorisation task in road sign recognition is important for highway authorities and why it is difficult to accomplish for academics. It presents the challenges which need to be dealt with when images are taken in outdoor environments, and concludes with a statement of the requirement for a scientifically satisfactory solution

Chapter 4:

In this chapter a review of the literature and previous work is presented. It covers papers, technical reports and internet resources which were collected for review. The review includes the study and analysis of colour segmentation algorithms, shape identification and recognition, road sign classification, and pictogram classification. It also reviews the different techniques undertaken to achieve road sign recognition as a computer vision task.

The chapter explores colour and its properties and stability when dealing with different light conditions and geometry. It concentrates on hue as the main source of information for colour segmentation.

Zernike moments, Legendre moments, Orthogonal Fourier-Mellin Descriptors, Binary Haar invariants are discussed in this chapter. This is followed by a description of the Support Vector Machines (SVM) classifier which is invoked in the last stage.

Chapter 5:

This chapter covers the methodology invoked to build the traffic sign recognition system. Image collection and the image database are presented at the beginning, followed by the colour segmentation algorithms which were developed in this research. The chapter covers details of the recognition algorithm which was designed to identify traffic signs in the scene. Finally, the chapter describes the classification method used for traffic sign recognition

Chapter 6:

This chapter presents details of the experimental work carried out to evaluate the various algorithms developed in the previous chapter. Colour segmentation algorithms were tested in different weather conditions, light conditions, and sign conditions. The same set of experiments was applied to the fuzzy shape recogniser. In the last part of this chapter another set of experiments was applied on the SVM classifier. The test includes training with different features, SVM types, kernels, and different moment orders. Finally, search for optimum values is presented.

Chapter 7:

This chapter summarises the main contributions and conclusions gained from this research. It presents the future plans which can be implemented to improve the work and issues for further investigation.

Appendix A:

In this appendix, details of the Swedish road and traffic signs together with their meanings and their groups are presented.

Appendix B:

This appendix brings together in one place techniques of describing colour as a source of information. This covers the famous colour spaces, their representation, advantages and disadvantages, and how to convert from one colour space to another.

Appendix C:

In this appendix, the explicit forms of Zernike moments are given.

Appendix D:

This appendix depicts the results of the experiments applied on the SVM and used for the evaluation of this classifier and its performance.

Appendix E:

The Access database of the traffic sign images is presented in this appendix. It shows the table of images, the table of signs, the phrases tables and the relationship among them.

Appendix F:

This appendix presents the list of publications achieved during this research.

2. Traffic Signs

Road and traffic signs, traffic lights and other traffic devices are used to regulate, warn, guide or inform road users. They help achieve an acceptable level of road traffic quality and increase safety with orderly and predictable movement of all traffic, both vehicular and pedestrians [3, 4].

Road and traffic signs are designed to be easily recognised by drivers mainly because their shapes and colours are readily distinguishable from their surroundings [5].

The Swedish Road Administration is in charge of defining the appearance of all signs and road markings in Sweden. Traffic signs in Sweden are fully regulated by this administration. They are placed two meters from the road and the base-sign is at a height of 1.6 meters for roads used by vehicles with motors. According to the Road Administration, the maximum number of signs on a single pole is three with the most important sign at the bottom. In accordance with European signs, all signs are designed to have a reflective layer added on selective parts of the sign.

Most Swedish road signs use pictograms to indicate the message of the sign. However, there are some exceptions in which text replaces pictograms. The STOP sign is one example of this kind of sign. All signs use Swedish text except the STOP sign where the English “STOP” word replaces the Swedish “STOPP” word.

The usual background colour on warning and prohibition signs on most European signs is white, whereas this colour is yellow in Sweden. The reason is to enhance the visibility of the signs during winter time. White signs would be very hard to see in snowfall conditions. A thicker rim is used for warning and prohibition signs in Sweden compared with their European counterparts.

2.1 Properties of Road and Traffic Signs

Road and traffic signs are characterised by a number of features which make them recognisable with respect to the environment:

- ❖ Road signs are designed, manufactured and installed according to strict regulations [6].
- ❖ They are designed in fixed 2-D shapes such as triangles, circles, octagons, or rectangles [7, 8].
- ❖ The colours of the signs are chosen to contrast with the surroundings, which make them easily recognisable by drivers [9].
- ❖ The colours are regulated by the sign category [10].
- ❖ The information on the sign has one colour and the rest of the sign has another colour.
- ❖ The tint of the paint which covers the sign should correspond to a specific wavelength in the visible spectrum [6, 11].
- ❖ The signs are located in well-defined locations with respect to the road, so that the driver can, more or less, anticipate the location of these signs [10].
- ❖ They may contain a pictogram, a string of characters or both [11].
- ❖ In every country the road signs are characterised by using fixed text fonts, and character heights.
- ❖ They can appear in different conditions, including partly occluded, distorted, damaged and clustered in a group of more than one sign [7, 11].

2.2 Swedish Road and Traffic Signs

Signs in Sweden may be either ideogram-based which contain simple ideographs to express the sign meaning, or text-based where the contents of the sign may either be text or arrows or other symbols [12]. Swedish traffic signs can be categorized into four groups:

1. Warning signs: A traffic warning sign, Figure 2.1, is a type of traffic sign which indicates a hazard ahead on the road. It is characterised by an equilateral triangle with a thick red rim and a yellow interior. A pictogram is used to specify different warnings. The red-yellow combination is easily seen in snowy weather conditions. Other signs such as the YIELD sign and the distance to level crossing signs and track level crossing also belong to this class.

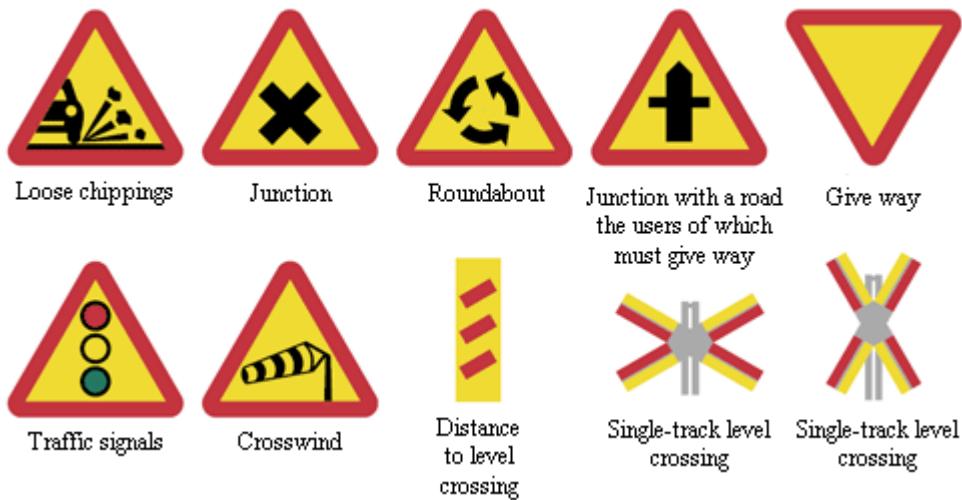


Figure 2.1: Warning signs.

2. Prohibitory signs: They are used to prohibit certain types of manoeuvres or some types of traffic. The no entry, no parking, and speed limit signs belong to this category, Figure 2.2. Normally, they are designed in a circular shape with a thick red rim and a yellow interior. There are few exceptions; the STOP sign is an octagon with a red background and white rim, the NO PARKING and NO STANDING signs have a blue background instead of yellow. The end of restriction signs are marked with black bars.

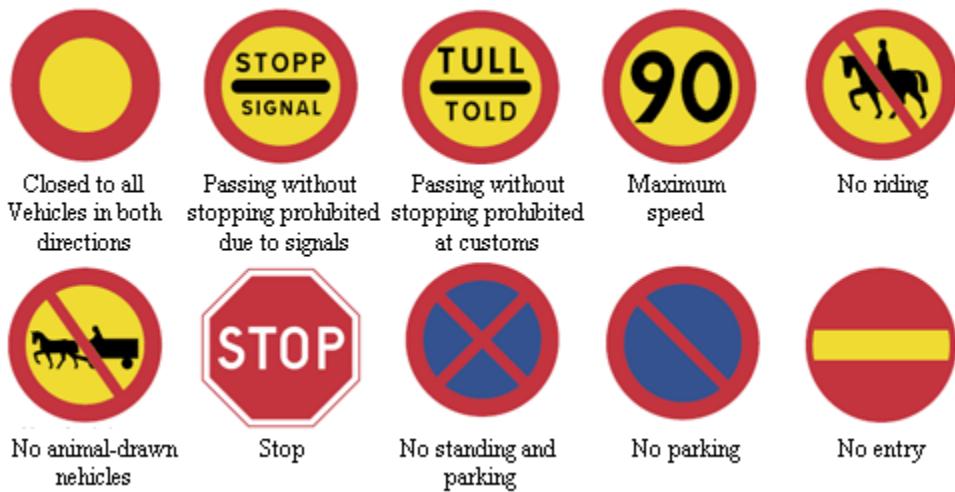


Figure 2.2: Prohibitory signs.

3. Mandatory signs: They are characterised by a complete blue circle and a white arrow or pictogram, Figure 2.3. They control the actions of drivers and road users. Signs ending obligation have a diagonal red slash.

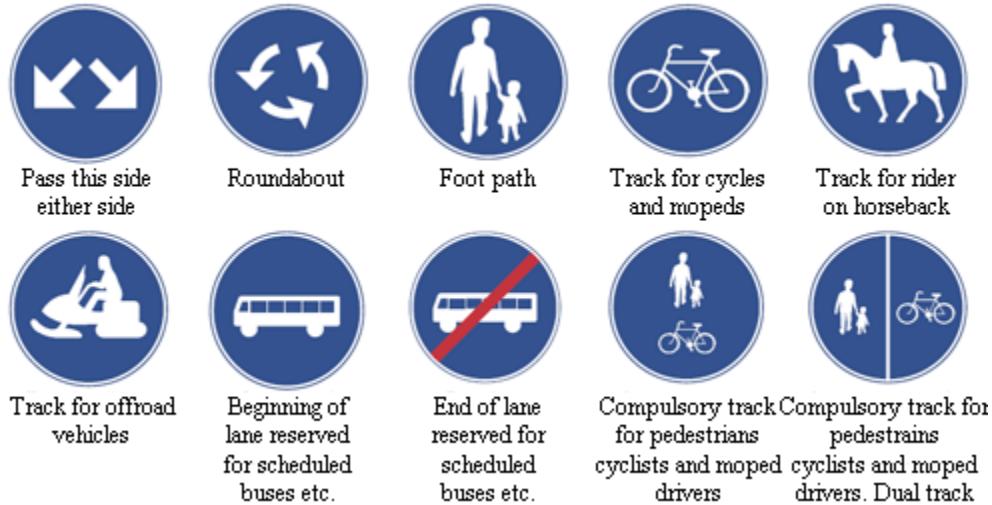


Figure 2.3: Mandatory signs

4. Indicatory and Supplementary signs: These types of signs are characterised by using rectangles with different background colours such as yellow, green, or blue. Figure 2.4 illustrates some example signs belonging to this category. The pictograms are either white or black. This category includes the diamond shaped rectangle and the signs which give information about road priority.



Figure 2.4: Indicatory and Supplementary signs

The full set of Swedish road and traffic signs, their meanings, and other information is depicted in appendix A.

As noted in the aforementioned discussion, the colour of the road sign together its shape determine its category. The colours used on road signs have specific wavelengths in the visible spectrum. They are selected to be distinguishable from the natural and man-made surroundings so that they can be easily recognisable by road users.

The National Road Administration in Sweden defines the standard colours used for road signs in CMYK colour space [13] , as described in Table 2.1. The meaning of each colour is given in Table 2.2. Moreover, Table 2.3 depicts the shapes used in Swedish road signs.

2.3 Summary

This chapter presents a description of road and traffic signs in general and the Swedish road and traffic signs in particular. It shows how they help drivers anticipate the road ahead in addition to any associated problems and hazardous situations. The chapter illustrates the general properties of road and traffic signs by describing the characteristics, colours and shapes of different categories of Swedish road signs

In the next chapter the problem of traffic sign recognition is addressed in detail and analysed from different perspectives.

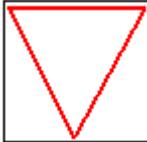
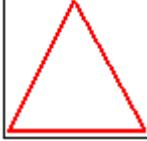
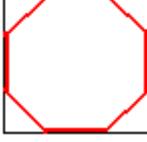
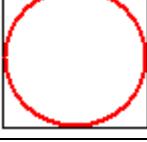
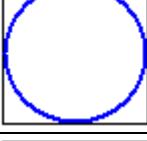
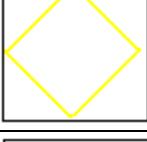
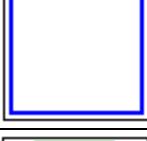
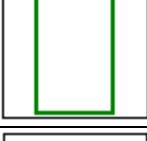
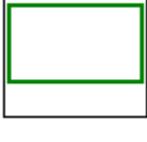
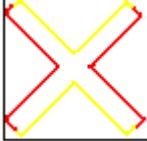
Table 2.1: Swedish standard colours.

Colour	Pantone	C	M	Y	K
Light Blue	294	82	56	0	18
Dark Blue	282	34	27	0	64
Green	335	70	0	65	30
Red	185	0	91	76	0
Yellow	116	0	15	94	0
Light Grey	444	9	0	6	47
Orange	152	0	51	100	0
Brown	469	0	27	32	61

Table 2.2: The meaning of colours used for traffic signs.

Colour	Meaning
Red	Exclusively for STOP and YIELD signs, DO NOT ENTER signs, and it is used in the warning signs and forbidden signs.
Black	Used as information colour in some of the warning signs, prohibitory signs and information signs.
White	Used as background for route markers, guide signs, and certain regulatory signs, and as message colour on signs with brown, green, blue, black, and red backgrounds.
Orange	Used as background colour for construction and maintenance signs.
Yellow	Used as background colour for Warning signs and Prohibitory signs, and some of the supplementary signs.
Green	Used as background colour for express roads.
Blue	Used as background colour for regulatory, information signs, and supplementary signs.

Table 2.3: Shapes of traffic signs and their meanings.

Shape	Example	Meaning
		Equilateral Triangle, Point Down -- Exclusively for YIELD signs
		Equilateral Triangle, Point Up – Exclusively for WARNING signs
		Octagon -- Exclusively for STOP signs
		Red Circle -- Exclusively for Forbidden signs
		Blue Circle -- Exclusively for Regulatory signs
		Diamond -- Used for PRIORITY ROAD
		Square --Used as additional sign or symbolic information sign
		Rectangle, Longer Dimension Vertical -- Used for EXPRESS ROADS
		Rectangle, Longer Dimension Horizontal -- Used for Information Express roads
		Cross buck -- Used for railroad crossing signs

3. Problem Formulation

Considering the object recognition and interpretation abilities of humans, it is a hard task to try to develop a computer based system which should be able to support people in every day life. There are a lot of conditions which are changing continuously such as luminance and visibility, which are handled by the human recognition system with ease but present serious problems for computer based recognition.

Looking at the problem of road and traffic sign recognition shows that the goal is well defined and it seems to be a simple problem. Road signs are located in standard positions and they have standard shapes, standard colours, and their pictograms are known.

To see the problem in its full scale, however, a number of parameters that affect the performance of the detection system need to be studied carefully. Road sign images are acquired using a digital camera for the purpose of the current analysis. However, still images captured from a moving camera may suffer from motion blur. Moreover, these images can contain road signs which are partially or totally occluded by other objects such as vehicles or pedestrians. Other problems, such as the presence of objects similar to road signs, such as buildings or billboards, can affect the system and make sign detection difficult. The system should be able to deal with traffic and road signs in a wide range of weather and illumination variant environments such as different seasons, different weather condition e.g. sunny, foggy, rainy and snowy conditions. Different potential difficulties are depicted in one section of this chapter.

Using the system in different countries can make the problem even worse. Different countries use different colours and different pictograms. The system should also be adaptive, which means it should allow continuous learning otherwise the training should be repeated for every country.

To deal with all these constraints, road sign recognition should be provided with a large number of sign examples to allow the system to respond correctly when a traffic sign is encountered.

3.1 What is Road Sign Recognition?

Road Sign Recognition is a field which is concerned with the detection and recognition of road and traffic signs in traffic scenes acquired by a camera. It is a technique which uses computer vision and artificial intelligence to extract the road signs from outdoor images taken in uncontrolled lighting conditions where these signs may be occluded by other objects, and may suffer from different problems such as colour fading, disorientation, and variations in shape and size.

The first paper on the subject was published in Japan in 1984 [14]. The aim was to try various computer vision methods for the detection of road signs in outdoor scenes. Since that time many research groups and companies have shown interest, conducted research in the field, and generating an enormous amount of work. Different techniques have been used to cover different application areas (see next Section), and vast improvements have been achieved during the last decade.

The identification of the road signs is achieved through two main stages:

- ❖ Detection
- ❖ Recognition.

In the detection phase, the image is pre-processed, enhanced, and segmented according to the sign properties such as colour or shape or both. The output is a segmented image containing potential regions which could be recognised as possible road signs. The efficiency and speed of the detection are important factors because they reduce the search space and indicate only potential regions.

In the recognition stage, each of the candidates is tested against a certain set of features (a pattern) to decide whether it is in the group of road signs or not, and then according to these features they are classified into different groups. These features are chosen so as to emphasize the differences among the classes. The shape of the sign plays a central role in this stage and the signs are classified into different classes such as triangles, circles, octagons. Pictogram analysis allows a further stage of classification. By analysing pictogram shapes together with the text available in the interior of the sign, it is easy to decide the individual class of the sign under consideration. A prototype of road

sign detection and recognition system is shown in Figure 3.1. The system can be implemented by either colour information, shape information, or both. Combining colour and shape may give better results if the two features are available, but many studies have shown that detection and recognition can be achieved even if one component, colour or shape, is missing. For the purpose of clarity, a full separation between recognition and classification is made in chapter 5 and 6.

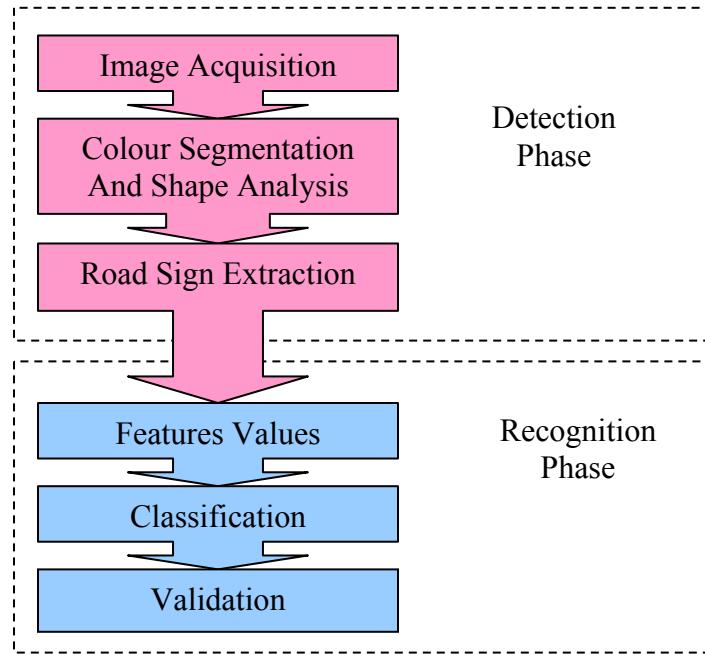


Figure 3.1: A block diagram of the road sign recognition and classification.

3.2 Road Sign Recognition Applications

Techniques for road sign detection and recognition have been developed in a range of application areas. These include:

- Driver Support System (DSS) can detect and recognise road signs in real time. This helps to improve traffic flow and safety [15, 16], and avoid hazardous driving conditions, such as collisions. Traffic sign detection and classification is one of the subjects which are not studied deeply. Research groups have focused on other aspects of sign detection, more related to the development of an automatic pilot, such as the detection of the road borders and/or the recognition of obstacles in the

vehicle's path e.g. other vehicles or pedestrians. Other systems are able to give warnings to drivers when they exceed the speed limit. Future Intelligent Vehicles would take some decisions about their speed, trajectory, etc. depending on the signs detected. Although, in the future, it can be part of a fully automated vehicle, now it can be a support to automatically limit the speed of the vehicle, send a warning signal indicating excess speed, warn or limit illegal manoeuvres or indicate earlier the presence of a sign to the driver. The general idea is to support the driver in some tasks, allowing him or her to concentrate on driving.

- Highway maintenance: This is used to check the presence and condition of the signs. Instead of an operator watching a video tape, which is a tedious work because the signs appear from time to time and the operator should pay a great attention to find the damaged ones, the road-sign detection and recognition system can do this job automatically for the signs with good conditions and alerts the operator when the sign is located but not classified.
- Sign inventory: The many millions of roadway signs necessary to keep roadways safe and traffic flowing present a particular logistical challenge for those responsible for the installation and maintenance of those signs. Road signs must be properly installed in the necessary locations and an inventory of those signs must be maintained for future reference.
- Mobile Robots: Landmarks similar to road and traffic signs can be used to automatically mobilise robots depending on the detection and recognition of these landmarks by the robot [16].

3.3 Potential Difficulties

In addition to the complex environment of the roads and the scenes around them illustrated in section 1.2, road signs can be found in different conditions such as aged, damaged, disoriented etc and hence the detection and recognition of these signs may face one or more of the following difficulties:

- ❖ The colour of the sign fades with time as a result of long exposure to sun light, and the reaction of the paint with the air, Figure 3.2 [3, 17].



Figure 3.2: Faded signs.

- ❖ Visibility is affected by the weather conditions such as the fog, rain, clouds and snow, as shown in Figure 3.3 [3].



Figure 3.3: Bad weather conditions (Rain and Snow).

- ❖ Visibility can be affected by local light variations such as the direction of the light, the strength of the light depending on the time of the day and the season, and the shadows generated by other objects, Figure 3.3 [11, 18, 19].
- ❖ Colour information is very sensitive to the variations of the light conditions such as shadows, clouds, and the sun, [3, 17, 20]. It can be affected by illuminant colour (daylight), illumination geometry, and viewing geometry, as shown in Figure 3.4 [21].



Figure 3.4: Bad lighting geometry.

- ❖ The presence of obstacles in the scene, such trees, buildings, vehicles and pedestrians or even signs which occlude other signs, as shown in Figure 3.5 [18, 20].



Figure 3.5: The presence of obstacles in the scene.

- ❖ The presence of objects similar in colour and/or shape to the road signs in the scene under consideration, such as buildings, or vehicles [17, 18]. They could be similar to the road sign in colour, shape or even both. Figure 3.6 shows two different cases; the first case shows a fence with a similar colour to the road sign. In the second case the post box has similar shape and colour to the signs and it is located very close to the road sign.



Figure 3.6: Similar objects in the scene or similar background colour.

- ❖ Signs may be found disoriented, damaged (Figure 3.7), or occluded by any kind of obstacles, even by some other signs, Figure 3.5. The signs in Figure 3.7 show two different damaged signs. The one on the left has damage to the red rim, while the one on the right is very old, rusted, damaged and the colour has faded due to aging.



Figure 3.7: Damaged signs.

- ❖ The size of the sign depends on the distance between the camera and the sign itself. Road signs may appear rotated due to the imaging orientation [22]. Figure 3.8 shows two images of the same sign take successively from two different positions on the motorway.



Figure 3.8: Size of signs depends on the distance between the camera and the sign.

- ❖ The acquired image often suffers from motion blur and car vibration [23]. This motion blur cannot be predicted above a certain level, because the car movements are not known to the recognition process. It is possible to make an assertion about the movements of objects in the future if the motion is continuous and unchanged. Figure 3.9 shows two images which suffer from a motion blur. The one on the left is clearly motion blurred. In the one on the right the sign is clear but the background suffers from a severe motion blur.



Figure 3.9: Motion blur problem.

- ❖ Sign boards often reflect the light from the sky or from an approaching car during the weak daylight hours or generate highlight, Figure 3.10. In the image on the left, the headlights of the vehicle can be clearly seen on the sign board, while the image

on the right shows a sign reflecting the light from the sky to the camera. This very often happens when collecting images at dawn or dusk.



Figure 3.10: Reflection from sign board.

- ❖ Different countries use different colours; and different pictograms. Figure 3.11 shows two images for the YIELD sign. The image on the left is taken in the Netherlands, and the image on the right is taken in Sweden. It is known that the interior of the warning sign is yellow in Sweden, but even the two sign rims are different in colour. In Figure 3.12, the same sign (Pedestrian crossing) is taken in four different European countries; Austria, the Netherlands, Poland, and Sweden. Comparing the four images shows that the pictograms differ from one country to another.



Figure 3.11: Colours in different countries: left, Netherlands; right, Sweden.



Figure 3.12: Different pictograms are used in different countries.

- ❖ People put stickers or write on the sign boards or damage the pictograms by changing the pictogram shapes. Figure 3.13 is an example taken in Barcelona, Spain. The number of stickers people have put on the sign is exceptional.
- ❖ The absence of a standard database for evaluation of the existent classification methods [12].



Figure 3.13: Stickers which damaged the pictogram of a sign.

It can be concluded from the above mentioned potential difficulties that it is extremely important for the algorithms to be developed for the detection and recognition of road and traffic signs to have high robustness of colour segmentation, high insensitivity to noise and brightness variations, and should be invariant to geometrical effects such as translation, in-plane and out-plane rotations and scaling changes in the image [24, 25].

3.4 Summary

In this chapter, the problem of road sign recognition was investigated. The problems that are faced when dealing with traffic signs were illustrated and potential difficulties were listed and described with the aid of images collected from real scenes. Road sign recognition as a concept was also presented in this chapter including its definition, its importance and applications. It is shown that sign recognition can be achieved by two basic stages; detection and recognition. In the detection stage, the sign is detected according to the sign's colour information, and then according to the sign's shape; features can be extracted and introduced to a certain classifier to decide the type of the sign according to the pictogram.

In the next chapter, the theoretical background including a literature review will be presented which covers the different techniques needed to solve the problem of traffic sign recognition.

4. Technical Overview

This chapter covers both the literature review and the technical survey of the techniques used in this research which includes colours, feature extraction and classification.

The chapter is divided into four sections. In the first section, a literature review is carried out to cover the last ten years of research in the field of traffic sign recognition. A wide range of techniques are studied and evaluated and conclusions are drawn about the state of the art of the road sign recognition systems.

The second section introduces colour as an important piece of information for a traffic sign recognition system, and describes why hue plays a central role in colour segmentation algorithms.

The third section covers feature extraction techniques, describing the different ‘moments’ that are used intensively in this field of research, namely, Zernike, Legendre, Orthogonal Fourier-Mellin moments, and the Binary Haar features.

The fourth section presents Support Vector Machine (SVM) as a classification stage. It is a linear classifier which belongs to a set of classifiers called ‘supervised learning classifiers’ and can be used for both classification and regression.

4.1 Literature Review

Since the appearance of the first paper in Japan in 1984, road sign recognition has become one of the important research fields. From that time until the present day many research groups have been active in the field and have tried to solve this problem using different approaches. Although initially the main steps towards a solution seem very well defined and straightforward, the details of the approaches used show that there are several alternatives and many ideas as to how better solutions, better robustness, or a better classification rate can be achieved. So far, no one solution method has dominated, and it will clearly take some time before systems are seen in the market.

The identification of road signs can be carried out by two main stages: detection, and recognition. In ‘detection’ research groups are categorised into three groups. The first

group of researchers believes that traffic sign colours are important information by which traffic signs can be detected and classified. The second group believes that detection of traffic signs can be achieved by traffic sign shape only, and the third believes that colour together with shape make the backbone for any road sign detection. Thus, there are three major approaches to detecting traffic signs: detection using colour information, detection using shape information, and detection using both colour and shape information. All of the reviewed papers used images from real traffic scenes which are similar to the images collected during this research.

4.1.1 Colour-Based Detection of Traffic Signs

Techniques invoked to carry out traffic sign detection varies from one author to another. There is a wide range of techniques used to solve this issue.

- **Ghica et al.** [26] used thresholding to segment pixels in a digital image into object pixels and background pixels. The technique is based on calculating the distance in RGB space between the pixel colour and a reference colour. The unknown pixel is considered as an object pixel if it is close enough to the reference colour.
- **Estevez and Kehtarnavas** [27] suggested an algorithm capable of recognising the Stop, Yield, and Do-Not-Enter traffic warning signs. It consists of six modules: colour segmentation, edge localisation, RGB differencing, edge detection, histogram extraction, and classification. Colour segmentation is only used to localise red edge areas; the segmentation is performed sparsely; and interpixel segmentation distance is determined.
- **Yuille et al.** [28] designed a sign finder system to help visually impaired people. The author assumed that signs consist of two colours (one for the sign, and another for the text), and sign boundaries are stereotyped (rectangle, hexagonal). Based on a set of tests to determine seeds, a region growing algorithm is used to detect hypothesis regions.
- **Yabuki et al.** [29] proposed a method to detect the road sign by using the colour distribution of the sign in XYZ colour space. They constructed a colour similarity map from the colour distribution, which is then incorporated into the image function of an active net model. It is possible to extract the road sign when it is wrapped up in an active net.

- **Fang et al.** [3] calculated the hue value of the HSI colour space for every pixel, and the similarity between this hue and the stored hue values of particular colours in road signs is calculated. The maximum degree of similarity is then considered. This result is fed into a perceptual analyser to specify the colour of the sign.
- **Shadeed et al.** [30] proposed an algorithm to detect road signs using the HSV and YUV colour spaces. The system is implemented in two stages. In the first stage, the RGB image is converted into YUV colour space, and a histogram of the Y channel is equalized, and then a new RGB is constructed. Colour segmentation is achieved in the second stage by converting the RGB image generated by the first stage into HSV and YUV colour spaces, and then applying a suitable value of threshold to H and UV values. Then the two results are combined by an AND operation.
- **Bénallal and Meunier** [31] developed a computer vision system which is embedded in a car and capable of identifying road signs. Many experiments were carried out with several road signs to study the stability of colours under different illumination conditions. Segmentation is achieved by the RGB colour space. It is shown that differences between red and green and blue components respectively are high and could be used with an appropriate threshold for segmentation.

4.1.2 Shape-Based Detection of Traffic Signs

Techniques using shapes could be a good alternative when colours are missing or when it is hard to detect colours. Shape-based techniques should be able to avoid difficulties related to invoking colours for sign detection and robust to handle in-plane transformations such as translation, scaling and rotation. Much effort has been exerted to develop these techniques and the results are very promising.

In the following reviewed papers the authors used shapes as the major source of information to detect traffic signs:

- **Piccioli et al.** [32] and Parodi and Piccioli [7] detected road signs by using a priori information of the supposed position of the sign in an image. A Canny edge detector was applied to the search region, and geometrical analysis was carried out on clusters of edge-points to extract the desired shape. The inner region of each candidate was tested against the database of signs by template matching. The correlation of the edge

pixels with an appropriate set of circular masks was used to detect circles. Triangles were detected by grouping edges in vertical, horizontal, and oblique segments.

- **Priesc et al.** [33] suggested a real-time traffic sign recognition system in which traffic signs are identified by the interpretation of their ideograms realised by different modules in the recogniser. There are modules for the position and direction of arrows, a module for the numerals, and another for prohibition signs, speed limits, and arrows on mandatory signs.
- **Aoyagi and Asakura** [34, 35] proposed a method to detect the traffic signs using brightness only. The object is extracted from the background using the Laplacian filter after using a smoothing filter to remove the noise. To obtain the binary image, a certain threshold is applied and detection is carried out by genetic algorithms with search ability for the circular pattern which is given as gene information.
- **Adorni et al.** [36] used Cellular Neural Networks to identify road traffic signs. A gradient operator is used to extract the border pixels from the image, followed by the application of a low value threshold to remove small gradient intensity pixels, and then using a 5x5 CNN single-iteration filter to perform the pre-selection of pixel with respect to the neighbouring pixels.
- **Gavril** [37] described a method to classify road signs based on template matching using distance transforms. The method could detect circular and triangular signs. Edge orientations are used as features which the algorithm depends upon. Different templates with radii between 7-18 pixels are used for circles and triangles. Each template is partitioned into 8-typed templates based on edge orientation. The method is used to detect road signs both on-line and off-line with a detection rate of about 90%.
- **Schiekel** [38] addressed the problem of recognising road signs in poor light conditions such as twilight, where colour information is not sufficient. The original image is processed by two Sobel filters, the gradient magnitude and orientation are calculated, and edge pixels are identified by thresholding the magnitude image. Edge pixels are segmented in two steps organised hierarchically, in which low-level features of pixels such as gradient orientations are linked to high-level features such

as triangles and ellipses. The recognition rate is 95% for triangular and elliptical signs, and 93% for rectangular signs.

- **Huang and Hsu** [39-41] developed a road sign detection and recognition system based on the Matching Pursuit (MP) method. In the detection phase a region of interest is selected according to a priori information. A search to extract triangular or circular shapes is achieved in the region of interest (ROI) area by using template matching. The detection rate of triangular signs is 93% and 95% for circular signs.
- **Paclík and Novovičová** [23] developed a classification module based on Hierarchical Spatial Feature Matching (HSFM) method. In the detection stage, a list of regions where the signs are likely to exist was generated. This list is passed to the classification module, by which each region is either labelled with the sign type found in this region, or marked as a rejected region.
- **Perez and Javidi** [42] carried out a road sign detection system which is scale and illumination invariant. The stop sign was chosen to be the reference target. It was tested in different light conditions and different backgrounds. Two methods are tested: the composite filter, and the filter bank.
- **Perez and Javidi** [24] proposed a road sign recognition system by using a composite filter bank. The system is developed to allow in-plane and out-plane rotations of the road sign. It detected road signs even when they were slightly tilted, and out of plane rotations due to different angles of the acquisition system.
- **Sandoval et al.** [43] developed a method to detect traffic signs by using angle-dependent edge detection. The method is based on the generation of position-dependent convolution mask, which uses the angular position of the pixels under consideration. The method is applied as a filter and used to detect circular edges.
- **Puntavungkour et al.** [44] developed an automatic recognition system for road signs. A region of interest containing the road sign is specified by using affine geometric correction. Sign identification based on the grey-scale image processing is applied. Normalised Euclidean distances between the target image and the template images are calculated, and the template with the least value is chosen to represent the sign.

- **Hirose et al.** [45], Liu et. al. [46] and Liu et. al. [47] proposed a method for the recognition of traffic signs from motion pictures. The technique is based on applying an edge filter which is a normal Laplacian filter used to extract the sign edges, followed by another filter called the Simple Vector Filter (SVF) to extract the specific colour from the image. Genetic algorithms with search ability are applied to search for the circular traffic signs
- **Loy and Barnes** [48] developed a fast shape-based method for road sign detection. The method uses an extension of fast symmetry transform to detect regular polygons. It is tested on a range of sign images and it performs over 95%. The method is invariant to in-plane rotation and can be used for real time applications.

4.1.3 Colour-Shape-Based Detection of Road Signs

By invoking a combination of colour and shape, it is possible to take advantage of both techniques to detect traffic and road signs. Each approach has its own positive properties and difficulties. However, an adaptive hybrid approach can invoke one technique under certain circumstances and invoke the other under different circumstances. Even when this adaptive approach is not in use, combining colour and shape in any sign detection method has the advantage of using the information available from both sides of the problem. As both colour and shape represent information which should not be neglected, it is also possible to avoid many problems and disadvantages. Colour-shape-based systems were used in the following papers:

- **Hibi** [49] used hue and saturation in an improved HSL colour space to recognise road signs in night images. Dynamic thresholds are used for both hue and saturation histograms. The final binary image is generated by combining hue and saturation images by logical addition. Pixels of the binary image are allocated into seven boundaries depending on the target pixel and its neighbours. These boundary patterns are used to specify the outline shapes of the road sign.
- **Piccioli et al.** [50] showed two different algorithms for the detection of road signs. In the first one, grey-levels are used to detect road signs according to simple geometrical criteria. In the second one, hue and saturation in a HSV colour space are used. The

image is divided into 16×16 pixel regions, and each region is classified as 1 or 0 depending on whether the number of labelled pixels exceeds a certain threshold. A search is carried out only for regions labelled with ‘1’. Shape detection is based on the geometrical analysis of edge contours.

- **Azami et al.** [51] used the HSV colour space to detect the route guidance sign (RGS). Automatic threshold is chosen for hue, saturation, and value. A connected component analysis is applied to choose the RGS candidate according to size and shape.
- **Jiang and Choi** [9] used Fuzzy rules to transform the colour image into a grey-scale function, and a binary image is obtained to find any landmark in the enhanced colour image, in which enhancement is achieved by hue invariance. They used Nrgb colour space, thresholds, and fuzzy rules to detect the red and blue colours. Warning signs, which are considered here, are identified by extracting the three corners of triangles. A fuzzy method is developed to detect these corners by defining two member functions to specify the possibility of pixels inside two masks to create a corner. The other two corners are detected in the same way. The masks are rectified to eliminate the problem caused by damaged signs.
- **Vitabile and Sorbello** [6] proposed a road sign recognition system which consists of two modules: a sign detection module and a sign classification module. Detection is based on sign colour and shape. RGB images are converted into HSV colour space which is divided into a number of subspaces (regions). The S and V components are used to find the region in which the hue is located. The binary image generated by the former step is sent to the shape extractor to extract the road sign depending on the shape knowledge of the road sign. The image is compared with an internal database containing different sign shape templates and the template with maximum correlation is selected.
- **Miura et al.** [17] used area filters to binarise white regions in speed signs. Since binarization is sensitive to threshold, they binarised multiple times using different thresholds. By analysing the actual distribution of data in the YUV colour space, they could determine the right threshold. Shape information is used for the screening of

candidates. A search area is set around each candidate detected by the previous step and edges are extracted and tested for the presence of a specific shape.

- **Paclik et al.** [12] segmented colour images by using HSV colour space and selecting a certain threshold. The thresholds are setup by using real scene images which are collected under different illumination conditions. Shape analysis is carried out by calculating several moment invariant features such as unscaled spatial moments, unscaled central moments, normalised unscaled central moments, and compactness which are used to construct the feature vector used in the shape analysis.
- **Vitabile et al.** [20] proposed a dynamic, optimised HSV sub-space, according to the s and v values of the processed images. Colour segmentation is achieved by applying standard HSV colour filtering, generating sub-images to calculate seed pixels, and aggregating pixels depending on the seed saturation values by applying a region growing algorithm. Shape detection took place by using similarity coefficients between the segmented region and sample images for road signs. A segmentation rate of 94.6% for red circular signs, 86.3% for red triangular signs, and 95.7% for blue circular signs is achieved.
- **Lauzière et al.** [52] detected traffic sign colours using colour formation equations (CFE). The road signs' material spectral reflectance was measured with a spectrophotometer. The camera spectral response with monochromator and statistical daylight model was used. Regions of interest are extracted from the image on the bases of colour space labelling and connectivity of the pixels of similar colours. Features such as aspect ratio and road sign colours are extracted from the image and used to compute normalised templates. The template matching technique relies on an encoding of different road sign colours.
- **Vitabile et al.** [11, 19] designed an automatic road sign recognition system using multi-layer perceptron mapping on a SIMD architecture. Colour segmentation is achieved by using a priori knowledge about colour signs in the HSV system. Standard colour filters are applied and sub-images are generated using the seeded region growing techniques. Pixel aggregation with dynamic threshold is then applied depending on the seed saturation values. Candidate sign regions are selected

according to a priori knowledge about the shape of the sign by exploiting a similarity function.

- **de la Escalera et al.** [16] and de la Escalera et al. [53] built a colour classifier based on two look-up tables derived from hue and saturation of an HSI colour space. The two images are multiplied and normalised to 256 grey levels to create one binary image representing the classified road sign. In [16] Genetic Algorithms were used for shape analysis. In [53] two algorithms are used for shape detection: Genetic Algorithms (GA) and Simulated Annealing (SA) for shape analysis. Sign detection is achieved by normalised correlation and classification is achieved by neural networks. A detection rate of 90.4% is achieved by GA compared to 82.9% for SA.
- **Fang et al.** [4] developed a road sign detection and tracking system in which colour images from a video camera are converted into the HSI system. Colour features are extracted from the hue by using a two-layer neural network. Gradient values in specific colour regions are acquired by an edge detection method to construct an edge image which is fed to another two-layer neural network to extract shape features. Colour and shape features are combined by using a fuzzy approach to form an integration map.
- **Ohara et al.** [54] used a small and simple neural network (NN) to detect the colour and the shape of road signs. The original colour image is first treated by a Laplacian of Gaussian filter (LOG). A colour NN classifier is then used to segment the image according to the colour under recognition in RGB colour space. A shape NN is used after that to check whether each image contains an object with the shape of a road sign. When a shape is found, template matching is applied for final recognition.
- **Shirvaikar** [55] carried out an automatic system for road sign detection and interpretation. The system used RGB colour space and spectral feature analysis of the images under consideration to create binary images of candidates' pixel locations. Candidate regions are tested by a spatial feature analyser. Features employed to select the probable signs include area in pixels, aspect ratio, and fill factor. A relational feature analyser is used to refine the results. The system is used to detect stop, yield, and speed limit signs in real-time.

- **Nakamura et al.** [56] developed a recognition system of speed limit signs. First, the red area in the input image is detected by colour chart processing in the RGB colour space. Then the position and the size of the circular rings are detected by a multi-layered neural network. A neural network classifier is used for character recognition of speed limits. A recognition rate of 98.3% is achieved.

4.1.4 Recognition and Classification

Generally, word recognition is used to point out that a sign is identified while word classification is invoked to indicate that the sign is assigned in a certain category based on certain features. Sometimes recognition implies classification; therefore, these two terminologies are used interchangeably in this chapter. In the rest of the thesis, a complete separation between recognition and classification is made.

There are several techniques used for the recognition and classification of road and traffic signs. These techniques are summarised follows:

Neural Networks

Neural networks are widely used to classify traffic signs. There are many reasons for this, but primarily because of the high accuracy achieved by this classifier. Research in neural networks was at its peak in the 1990s as knowledge of using neural networks was very fashionable at that time. Since a number of the reviewed papers belong to that period, it is not a surprise to see a number of them using neural networks. In addition, the majority of researchers wanted to avoid using traditional classifiers and wanted to try new techniques. A review of the papers using neural networks as a classifier is given here:

- **Kellmeyer and Zwahlen** [57] used back propagation neural network to recognise warning signs. The input to the network which was a 10x10 boundary square representing the yellow region inside the warning sign, is fed to a 100 neuron input layer. The output-layer contains two outputs either “sign” or “non-sign”. A hidden layer of 30 nodes was used. The system could detect 55% of warning signs in 55 images. For large signs, 86% of the signs could be detected.

- **Ghica et al.** [26] carried out recognition by a neural network which consisted of three sub-networks, a classification sub-network, winner-takes-all sub-network (Hopfield network), and a validation sub-network.
- **Aoyagi and Asakura** [34] used neural networks with an input pattern of 18x18 pixels fed to a three-layer network consisting of 324 neurons in the input layer, fifteen neurons in the hidden layer, and three neurons in the output layer. The system could detect and classify 23 out of 24 speed signs, and 23 out of 24 other signs.
- In **Vitabile and Sorbello's** [6] system, classification is carried out by normalising the sign image to 36x36 pixels, and using two different multi-layer neural networks designed with a similar topology of 432-144-10 to extract the pictograms of the road sign under consideration. The first one is used to extract the circular red signs, and the second to extract the red triangular warning signs. Depending on the shape of the sign which is extracted in the former stage, one of these two classifiers is invoked.
- **Vitabile et al.** [20] used a neural network to classify the candidate sign regions according to the information inside it. Classification is carried out by a feed forward neural network classifier, where a 36x36 pixels candidate is fed to the neural network input. A classification rate of 84% for red circular signs, 88% for red triangular signs, and 100% for blue circular signs are achieved respectively.
- **Vitabile et al.** [11, 19] used a multi-layer perceptron neural network to classify the road signs. The system consists of three unrelated MLP neural network classifiers. The adopted topology is 432-144-O, where O=11 for warning sign, O=8 for prohibitory signs, and O=5 for mandatory direction signs. The system was tested on 620 outdoor images in 24 pictogram classes.
- **de la Escalera et al.** [16, 53] used neural networks for the classification of the traffic signs following the Adaptive Resonance Theory ART1.
- **Fang et al.** [3] carried out classification using the conceptual component module in which an ART2 network with a configurable long term memory was used to extract certain patterns from the categorical features fed from the perceptual module. These patterns are fed to another two-layer neural network to extract the road signs.

- **Nakamura et al.** [56] used neural networks to detect the position and size of speed limit signs. A neural network classifier is used for character recognition of speed limits. A detection rate of 100% and a recognition rate of 98.3% were achieved.

Template Matching

The second alternative classification technique is Template Matching. It is used either in its direct form or in a derived form i.e. by either matching a histogram or by matching a pattern generated from the unknown image with the standard patterns. The papers using this technique are reviewed here:

- **Piccioli et al.** [32] classified the recognised road sign according to its similarity to one of the road signs in a database. The inner region of each candidate is tested against the database of signs by template matching. To improve recognition scheme, Piccioli et al. [32] used Kalman-filter-based temporal integration to the extracted information from a sequence of images. A high rate of correct classification (about 98%) was achieved.
- **Huang and Hsu** [39-41] implemented a recognition process by using a robust and flexible Matching Pursuit MP filter, which was used to decompose the training pattern into a 2-D wavelet expansion. Here the information is represented locally; unlike template matching which encodes the information globally. The filter is used to extract the feature, which differentiates each class of road signs. The MP filter is trained off-line, and a conventional template matching is used to compare the input signals with the actual template to find the best match. A recognition rate of 94% for triangular signs and 91% for circular signs is achieved.
- **Piccioli et al.** [50] achieved road sign recognition by classifying road signs according to their shapes, normalising the sign to a size of 50x50 pixels by linear interpolation, and applying cross-correlation between the road sign and the template of the database related to the proper shape. The best N templates which fall within a fixed length of the maximum value are chosen. To increase the robustness, a temporal integration is applied to the selected sign from the database which best fits the unknown sign.

- **Azami et al.** [51] used a threshold to separate character regions from Route Guidance Signs (RGS). Since the characters are of the same size and placed horizontally, the search is done horizontally. Arrow candidates have larger areas compared with characters. They are recognised by their topology features using thinning and template matching. Seven kinds of arrows are recognised. The RGS and its contents are identified by comparing this structure with an RGS database. The method is applied to driving assistance systems.
- **Miura et al.** [17] identified road signs by using normalised correlation-based pattern matching between a test image and a template image. According to the proposed method, two thresholds are used for the correlation value. The candidate is considered if the correlation value of that candidate is above the first threshold, and the ratio of correlation values for the best and the second best candidates are above the second threshold. The method achieved a detection and recognition rate of 100% for the Guidance Sign Recognition, and 97.2% and 46.5% respectively for Speed signs.
- **Lauzière et al.** [52] achieved the recognition in three steps. In the first one, the road signs are divided according to their principal colour. They are then subdivided within the same colour class, according to their shapes and contents, into several classes. Finally, the individual road sign is found at the third level. Template matching is used in the recognition of the unknown sign with respect to the templates stored in the template database. Approximately 90% of the road signs are correctly recognised.
- **Ohara et al.** [54] used template matching for sign recognition. A sub-area of size NxN is selected, and small defects and noises are filled in or deleted. The pixel values are normalised by the maximum and minimum values of the input sub-area. The size is also normalised depending on the template to be matched, and the closeness with the template is calculated. A recognition rate of over 95% is achieved.
- **Hibi** [49] transformed target images and reference models to k-l coordinates using the complex-log mapping method which is scale and rotation invariant. Complex-log mapped images are then transformed into frequency domain by 2-D FFT. The target image is matched with reference images, and the reference with minimum judgement value is chosen. A recognition rate of 100% is achieved for Stop signs, 94% for No

Parking sign, and 95% for No Entry sign. The method is tested in daylight and night conditions.

Other Classifiers

The remainder of the reviewed papers use different classifiers, among them weighted distance metrics, nearest neighbour classifier, and kernel based classifiers. The following papers were reviewed:

- **Estevez and Kehtarnavaz** [27] performed the recognition by using an angular histographic attribute extracted by a semi-rectangular histographic mask. 50% of all stop signs were correctly identified, 41% of all stop signs were incorrectly identified. 37% of all yield signs were correctly identified, and 94% of all do-not-enter signs were correctly identified.
- **Paclík and Novovičová** [23] used two different features for classification. The first feature, which is extracted from the grey image histogram, includes the mean, energy, and entropy of the region containing the sign. The second feature set includes the unscaled central moments, normalised unscaled central moments, and Hu moments which are used for shape description. Five separate classifiers working in different feature spaces are used. It is tested in a wide range of lighting conditions which vary from full sunshine to cloudy twilight.
- **Paclík et al.** [12] used the Laplace kernel classifier to classify road signs. The signs are divided into nine groups depending on their shapes and colours. This kernel is based on Laplace probability density, and the smoothing parameters of Laplace kernel were optimised by the pseudo-likelihood cross-validation method. The Expectation-Maximisation algorithm is used to maximise the pseudo-likelihood function. The algorithm is tested on more than 4900 noisy images.
- **Dahyot et al.** [58] carried out a pattern recognition system based on an appearance-based representation of colour images. The approach relied on the M-estimators involving a non-quadratic and non-convex energy function. The approach is tested on triangular road signs along with their rotation in the image plane. The training set consists of 1548 (76x76) images representing 43 different road signs. The Euclidian distance between the estimation and all the training images in the eigenspace is calculated, and the closest model is selected.

- **Paclik et al.** [59] designed a classifier by using a trainable similarity measure base on normalised cross correlation. Images of traffic signs are normalised to 32x32 pixels and row binary pixel representation is used and PCA algorithm is invoked to reduce the dimension of the input feature. Two types of classifiers are used: Fisher linear discriminant (FLD) and Soft Independent Modelling of Class Analogy (SIMCA) classifier.
- In 2005, **Lafuente-Arroyo et al.** [60] used Support Vector Machines (SVM) to classify traffic sign shapes based on the distance of the edge of the sign from the border of the normalised image. The method achieved 82% success rate for standalone signs and 54% for occluded signs.

4.1.5 OCR and Pictograms Recognition

A number of authors do not pay special attention to pictograms because they consider pictograms as part of the sign interior. However, the following reviewed papers show special attention to pictogram recognition and text recognition for traffic signs:

- **Jiang et. al.** [61] developed a method to recognise traffic signs by using binary morphological operations and rank statistics. They were interested in recognising warning signs (triangular shape), and indicatory signs (circular shape). They concentrated on extracting the inner signs, assuming that the outer contours of the signs are extracted by some pre-processing.
- **Jiang et al.** [62] performed a study to analyse the morphological skeletons of the inner shapes of traffic signs and developed a method based on a morphological skeleton to detect the inner shapes of traffic signs. The relative functions and distance functions of the morphological skeletons were used to recognise the inner shapes and then the traffic signs. Ten warning signs are detected by this method.
- To extract the symbols and characters, **Miura et al.** [17] applied intensity transformation to avoid lighting variations. These characters and symbols were segmented by projecting character regions on the vertical axis, and calculating the histogram for prominent peaks, which are indications of the existence of characters. Horizontal projection was applied to specify the character position. The identification of each character is achieved by normalised correlation-based matching.

- **Nakamura et al.** [56] developed a recognition system of speed limit signs. The system recognises characters by multi-layer perceptron neural network. Back propagation is used as the training method. First it recognises the first digit to the right. If this digit is not zero the sign is rejected as a speed limit sign. Then it recognises the digit to the left. The speed limit is obtained by multiplying the digit to the left by 10.

4.1.6 Analysis of the Literature Review

I. Colour-Based Approaches

Colour is an important source of information in traffic signs recognition. The first part of colour detection is colour space conversion in which colour gathered by a camera in RGB form can be converted into another colour space so that the colour information can be separated from the intensity information. Some researchers prefer to use RGB colour space or a modified version of this colour space while others prefer to undertake colour space conversion to get better results. The major colour-based techniques are summarised below:

1. **Colour Thresholding Segmentation:** This is one of the earliest techniques used for segmentation of colour images. The method uses a threshold value to classify image pixels to traffic sign pixels or background. A reference colour is used to judge whether a pixel is considered as a traffic sign pixel or not [26, 63].
2. **Dynamic Pixel Aggregation:** Segmentation in this method is performed by introducing a dynamic threshold in the pixel aggregation process on HSV colour space. The main advantage of dynamic threshold is to reduce hue instability in real scenes depending on external brightness variation[20].
3. **HSI/HSV Transformation:** These two colour spaces separate colour information (Hue and Saturation) out of the overall intensity value which makes them more immune to light changes. The transformation from RGB colour space to HSI colour space makes the separation between chromatic information and intensity information useful for colour segmentation as the HSI colour space is very similar to human perception of colours.

4. **Region Growing:** This approach uses a seed in a region as a starting point and expands as groups of pixels with a certain colour similarity. The approach can be implemented in the HSI colour space. As it requires a seed to start and ends when certain criteria are met, it may run into a problem when ending conditions are not satisfied [28].
5. **Colour Indexing:** Colour histograms are used to compare colours in two images. The method is fast and straightforward. The colour histogram is used to index the images stored in a database. Computations will increase greatly in complex traffic scenes [29].

From the literature review it is clear that road signs use colour to represent the key information provided to drivers. Colours are an important source of information in the detection and recognition of traffic signs. As colours are distinguishing features of traffic signs, they can simplify this process. In addition, colour processing can significantly reduce the amount of false edge points produced by low-level image processing operations. An important part of the colour-based detection system is ‘colour space conversion’, which converts RGB images into other forms which simplify the detection process. This means separating the colour information from the brightness information by converting the RGB colour space into another colour space. This gives good detection abilities depending on the colour cue. There are many colour spaces available in the literature, namely the HSI, HSB, HSV, HLS, L*a*b*, YIQ, and YUV colour systems [64]. The hue-saturation systems are the most used in road sign detection, but the other colour systems are also used for this task.

Colour detection and segmentation is the next stage of colour based analysis. In this stage special algorithms are developed to separate objects with certain colours from the scene under observation. Many segmentation algorithms have been developed during the last decade. A wide range of colour spaces have also been used. All colour spaces are invoked by different authors. The methods used for colour segmentation vary depending on whether neural networks, fuzzy sets or statistical methods are used. It is clear that there is no one standard colour segmentation method and a large variety of methods are presented [3, 4, 16, 27, 53, 65, 66].

The accuracy achieved by invoking these methods varies between 37% up to 100%. Many of these algorithms are developed for real time applications. One or two systems have been tested for rain conditions.

Four major issues are missing in these algorithms and they are necessary to be investigated in future work.

- The absence of algorithms dealing with poor light conditions. As is the case for Sweden and other Scandinavian countries where winter is long and daylight hours are few.
- The absence of algorithms dealing with severe rain showers and snow fall.
- There are no algorithms that handle road signs located under trees in which different parts of the sign are exposed to different levels of illumination,
- There are no colour segmentation algorithms which are immune to ‘highlights’. In ‘highlights’ signs function as a mirror to reflect some of the source light to the camera.

II. Shape-Based Approaches

In shape-based sign detection grey scale images are used to avoid different problems faced when dealing with colours. The outer edges of the signs are used in many studies in this field. Among the techniques used to extract road signs are the following:

- 1. Hierarchical Spatial Feature Matching:** The search for geometrical shapes is carried out based on spatial features of signs within the traffic scene. Once these shapes are found a list is created and passed to a classification module.
- 2. Hough Transform:** The classical Hough transform has been used to detect regular features such as lines and circles. It is used because of its ability to isolate features of a particular shape within an image. The method is computationally complex and memory hungry which does not make it a good choice for real-time applications. However, these constraints are not crucial issues for road sign inventory.
- 3. Similarity Detection:** This approach is performed by finding a similarity factor between a segmented region and a set of binary images which represent each road sign shape. The method assumes that both sampled and segmented image have the same dimensions.

4. Distance Transform Matching: In this approach a template hierarchy is used to capture the variety of object shapes. Efficient hierarchy can be generated offline for given shape distributions using stochastic optimisation techniques. In the online mode, a simultaneous coarse-to-fine approach is involved over the shape hierarchy and over the transformation. The approach is capable of checking objects of arbitrary shapes which is an advantage over other techniques when dealing with non-rigid objects.

It has been proved that it is enough to use road sign shapes to detect them. One of the points to support this theory is the lack of a standard colour system among the different countries even within the European Union. Systems relying on colour need to be tuned when moving from one country to another. The other point in this argument is the fact that colours vary as daylight and reflectance properties vary. In situations in which it is difficult to extract colour information such as twilight time and night time shape detection will be a good alternative. However, using shapes to detect road and traffic signs may suffer from some difficulties. Among these difficulties are the following:

- Objects similar to traffic signs, such as windows, mail boxes and cars, may exist in the scene.
- Road signs may appear damaged and/or occluded by other objects, and they may appear disoriented vertically or horizontally.
- As the distance between the camera and the sign varies, the size of the sign also varies. When the sign is very small, it may be unrecognisable.
- When the viewing angle is not head-on, the aspect ratio may also change. The camera in this case takes a perspective projected image of the sign.
- Working with shapes necessitates robust edge detection and a matching algorithm. This is difficult when the road sign appears relatively small in the image.

III. Colour-Shape-Based Approaches

It is clear that combining colour information with that of shapes gives a good source of information for traffic sign detection. Shape information is valuable as much as that of colour. This kind of combination reduces the number of false alarms since every object,

not just road and traffic signs, can possess these specifications. As mentioned above, adaptive traffic sign detectors can be built by using this colour-shape combination. When colours are available they can be used to detect the signs. Otherwise a shape-based algorithm can be invoked. This kind of traffic sign detector may need some kind of rule to control which method is used depending on the availability of colour information or shape information. In addition to that, combining colour and shape in one algorithm can also reduce false alarms by avoiding some of the problems which can arise due to the nature of either of these approaches.

IV. Recognition and Classification

From the review a number of parameters should be taken into consideration when a classifier is designed:

1. The recogniser should present a good discriminative power and low computational cost.
2. It should be robust to the geometrical status of sign, such as the vertical or horizontal orientation, the size, and the position of the sign in the image.
3. It should be robust to noise.
4. The recognition should be carried out quickly if it is designed for real time applications.
5. The classifier must be able to learn a large number of classes and as much priori knowledge as possible about road signs should be employed into the classifier design.

Neural Networks are a suitable alternative for the recognition and classification of road signs. There are two distinct advantages of using neural networks. First, the input image does not have to be transformed into another representation space. Second, the result depends only on the correlation between the network weights and the network. However, neural networks have their own problems. The training overhead still exists, and the multi-layer neural networks cannot be adapted for on-line application due to their architecture. Since this architecture is fixed, there is no provision for an increase in the number of classes without a severe redesign penalty, and they cannot recognise the new

patterns without retraining with the entire network. In this respect, they do not offer significant advantages over template matching.

Other types of neural networks, such as reconfigurable neural networks, ART1, ART2, Hopfield, Cellular neural networks, try to offer more flexibility and adaptation of neural networks [6, 11, 19, 22, 67]. Kohonen maps were trained for signs partially occluded by other objects and signs which were rotated by small angles in the outdoor images. They can have capabilities to adapt to new signs without need of new training.

Template matching is a second alternative in the recognition stage. It is used to classify the inner regions of traffic signs, and in some cases, combined with wavelets to extract the local features of the sign. Complex-log transform and 2D-FFT are also combined with template matching to achieve better classification results [32, 49, 50, 68].

Genetic Algorithm can be used to search for traffic sign in a scene image. The image is matched by giving the gene information. The gene of individuals can be represented by expression using a set of equation to determine its characteristic.

Nearest Neighbour Classification is a straightforward and classic type of classification. An image in the test set is recognised by assigning to it the label of most of the closest points in the learning set. All images are then normalised to certain value. The image in the learning set that best correlates with the test image is then the result.

Support Vector Machines classifier is a potential classifier which shows good abilities to classify patterns for different applications. This classifier which has only been introduced in recent years to the field of traffic sign recognition is chosen as a classifier in this thesis. Details of the classifier are presented in section 4.4.

Other classifiers such as classical classifier, weighted distance classifier, Angular histogramic, matching pursuit classifier, Laplace kernel classifier, and Euclidian distance have also been used for road sign classification [12, 69].

V. OCR and Pictogram Recognition

The amount of research achieved clearly shows that optical character recognition (OCR) is not a basic problem any more. The availability of commercial packages and the robustness of these packages indicate that research in OCR for sign recognition has reached stability. The reason is that OCR techniques available nowadays have passed the

requirement for road sign recognition. As mentioned earlier, the text on the road signs has standard fonts and standard sizes, which makes the recognition of this text a simple task.

Extracting pictograms is more difficult than extracting text. A good quality close-up image is needed to extract the pictogram from a sign under consideration,. When the road sign is small in the test image, the details of the pictogram will not be clear. This leads to an unrecognisable pictogram. Since most of the pictograms contain complex shapes, and a variety of symbols, the task of the recognition will be extremely tedious when only a few pixels are available to represent the pictogram.

However, traffic sign detection can benefit from pictogram extraction, because the position and shape of the sign are already specified in an earlier stage. In addition, the whole classification operation can be carried out off-line, which provides good quality, because very sophisticated algorithms can be used.

VI. Discussions

Various commonly used techniques on road sign recognition have been presented. Some of these techniques can be combined with others to realise a hybrid recognition system.

Comparing to colour-based recognition, shape-based recognition faces more limitations. However, the shortcomings of colour-based recognition such as weather conditions and faded colour on road sign can be compensated through the use of shape-based recognition giving more superior performance. For colour-based recognition, most approaches can operate considerably fast except colour indexing. Even though colour indexing can segment an image when the road sign is slightly tilted or partially occluded, its computation time increases greatly in a complicated traffic scenes. Hence, colour indexing is not ideal for real-time implementation. Camera captures images in RGB colour format of which is not suitable for image segmentation as the colour information varies greatly under different weather condition. Often it is transformed into other colour space to like HSV, XYA space for processing. However, these transformations may become expensive in hardware realisation. Whilst, the Colour - Thresholding is easy, it may not be robust when the weather is poor or when the paint on road sign is faded.

The slow computation time of Hough Transform is a disadvantage, as it makes the method improper for real-time implementation. Template matching is popular but additional techniques need to be used with it to compensate for the imperfect sign shape problem. Similarity detection is like the Colour Thresholding in colour-based, simple and straightforward. However, this method may not give reasonable hit rate.

Neural Networks classification is popular amongst pattern recognition applications; road sign recognition makes no exception. Neural network approach has the most number of references, and yet there are several more that have not been included. The recognition rate varies quite significantly (550% 100%), this could be due to how well the networks are trained, and what network architecture are employed. The networks that achieved the highest hit rate used feedforward MLP Neural Networks. Neural networks are more adaptable to changes, and more flexible.

Growing number of experimental results prove their robustness, which makes them stand out from the rest of the techniques. Some of the techniques are robust but computationally costly, while others are simple but unable to handle changes in road sign patterns. None of the existing approaches can be totally immune to the problems faced by road signs recognition system.

4.2 Colour

Colour provides powerful information for object recognition. Segmentation based on colour provides good discrimination between material boundaries. As mentioned earlier, road signs use colours to represent the key information provided to road users. As colours are distinguishing features of traffic signs, they can simplify the recognition. In addition, colour processing can significantly reduce the amount of false edge points produced by low-level image processing operations [70].

Colour space is often used in computer vision. It is defined as a mathematical representation of a set of colours. Although the number of existing colour spaces is large, many of them are related to some major groups of colour spaces. The main models of colour spaces are given below:

- The RGB colour model is used for computer graphics and CMYK for printing.

- HSI, HSV and HLS are designed to emulate the way human perceive colour.
- YIO, YUV and YCbCr are designed for video systems.
- XYZ for standard primary colours.

As the information regarding colour spaces are scattered, a full description of colour spaces is presented in appendix B.

Figure 4.1 depicts a model for a normal scene when an image is taken. Light from the source strikes a surface is reflected and enters the camera where it is sampled by the red, green, and blue sensitive receptors.

In this Figure, the vectors $\mathbf{n}, \mathbf{s}, \mathbf{v}$ are unit vectors which represent the direction of the normal vector to the surface of the sign, the direction of the source of illumination, and the direction of the viewer, respectively.

The response of the RGB sensors at position \mathbf{P}_s measuring the light reflected from a Lambertian surface is given by [71]:

$$C(\mathbf{P}_s) = m_b(\mathbf{n}, \mathbf{s}) \int_{\lambda} f_C(\lambda) e(\lambda) c_b(\mathbf{P}_o, \lambda) d\lambda \quad \text{for } C = \{R, G, B\} \quad (4.1)$$

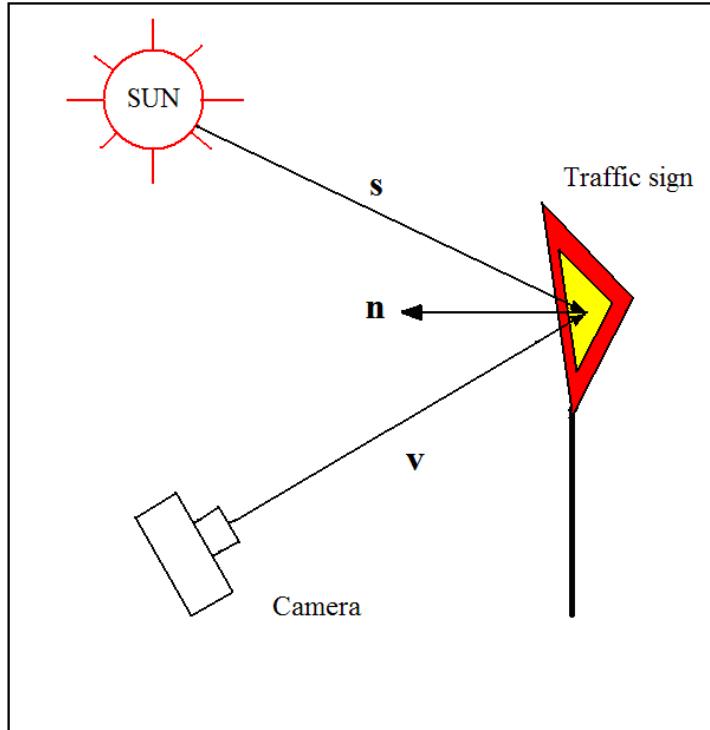


Figure 4.1: Traffic scene model.

Where $C(\mathbf{P}_s)$ is the response of the sensor which is located at position \mathbf{P}_s to the RGB colours, m_b denotes the geometric dependencies on the body reflection component, \mathbf{s} is a unit vector pointing in the direction of the light source, \mathbf{n} a unit vector corresponding to surface normal, $e(\lambda)$ is the spectrum of the incident illumination, $c_b(\mathbf{P}_o, \lambda)$ is the spectral reflectance of the body located at position \mathbf{P}_o , and $f_C(\lambda)$ is the spectral sensitivity of the camera in the RGB colour. The integration is done over all wavelengths to which the sensor responds.

One of the key points of equation (4.1) is that the responses of the camera sensors are related to the spectral characteristics of the incident light and the surface which reflects that light.

4.2.1 Variation of Colour in Outdoor Images

As shown in Figure 4.1, light rays come from the source of light and are reflected on the sign board's surface to the imaging device. The apparent colour of any object in outdoor images varies according to changes in the daylight colour (illuminant colour), illumination geometry, viewing geometry and imaging parameters.

The irradiance of any object in a colour image depends on three parameters:

1. *The colour, intensity and position of the light source.*

One of the difficult problems in using colours in outdoor images is the chromatic variation of daylight. As a result of this chromatic variation, the apparent colour of the object varies as daylight changes. Daylight colour changes according to the angle of the sun, cloud cover, and local weather conditions along the characteristic curve in the CIE model, as shown in Figure 4.2. It is given by the following equation:

$$y = 2.87x - 3.0x^2 - 0.275 \text{ for } 0.25 \leq x \leq 0.38 \quad (4.2)$$

Where x and y are the coordinates of the CIE colour model. According to this equation, the variation of daylight colour can be expressed by a single parameter “colour temperature” which is independent of the intensity.

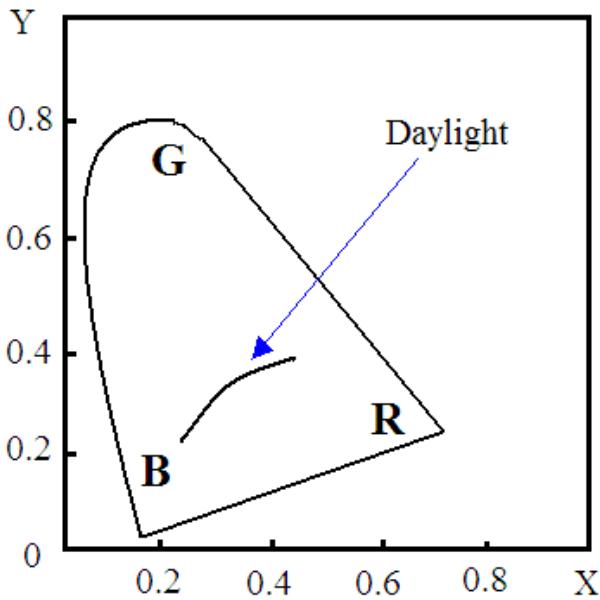


Figure 4.2: The CIE diagram.

2. *The reflectance properties of the object.*

A Lambertian surface is that surface which reflects light with equal intensity in all directions. When a Lambertian colour model is assumed, the surface reflectance of an object $s(\lambda)$ is strictly a function of the wavelength λ of the incident light. It is given by following equation:

$$s(\lambda) = e(\lambda) c_b(\lambda) \quad (4.3)$$

Where $e(\lambda)$ is the intensity of the light at wavelength λ , and $c_b(\lambda)$ is the object's albedo function giving the percent of the light reflected at each wavelength. Although this model does not take the extended light sources, inter-reflectance effects, shadowing or peculiarities into consideration, however, it is the best available working model of colour reflectance.

3. *Camera properties.*

Given the radiance of an object $L(\lambda)$, the observed intensities depend on the lens diameter d , the focal length f_l of the camera, and the image position of the object measured as angle a off the optical axis. This is given by the standard irradiance equation:

$$E(\lambda) = L(\lambda).(\pi/4)(d/f_l)^2 \cos(4a) \quad (4.4)$$

According to equation (4.4), the radiance $L(\lambda)$ is multiplied by a constant function of the camera parameters. This means that it will not affect the observed colour of the object. Assuming that the chromatic aberration of the camera's lens is negligible, only the density of the observed light will be affected.

As a result, the colour of the light reflected by an object located outdoors is a function of the temperature of daylight and object's albedo, and the observed irradiance is the reflected light surface scaled by the irradiance equation [21, 72].

4.2.2 Hue Invariance

Hue is defined by the CIE as the attribute of a visual sensation according to which an area appears to be similar to one of the perceived colours, red, green and blue, or a combination of two of them. In practical terms, hue is the name of the colour [73]. By mimicking humans, computer vision invokes hue in various applications such as colour segmentation and suppresses the effects of cloud shadows for remote sensing applications. RGB to HSV and HLS colour transformation find a variety of applications in conversion applications such as object recognition and face recognition.

The human vision system ascribes fairly constant hues to surfaces viewed in different visual contexts. It has been shown in the former section that hue is invariant to brightness. It is also invariant to highlights. Thus hue plays a central role in colour detection and segmentation for many applications because of these properties. Some more interesting properties of hue which make it invariant to the variations in light conditions are given below:

- Hue is multiplicative/scale invariant: $hue(R, G, B) = hue(aR, aG, aB)$ for all values of a such that $(aR, aG, aB) \in [0, 255] \times [0, 255] \times [0, 255]$
- Hue is additive/shift invariant: $hue(R, G, B) = hue(R + b, G + b, B + b)$ for all values of b such that $(R + b, G + b, B + b) \in [0, 255] \times [0, 255] \times [0, 255]$
- Hue is invariant under saturation changes. Practically this means that it is still possible to recover the tint of an object when it is lit with intensity varying

illumination. It has been shown in the aforementioned section that hue is unaffected by shadows and highlights when illumination is white.

However hue coordinates are unstable, and small changes in the RGB can cause strong variations in hue [10]. Furthermore, hue suffers from three problems:

1. When the intensity is very low or very high, hue is meaningless.
2. When saturation is very low, hue is meaningless.
3. When saturation is less than a certain threshold, hue becomes unstable.

Vitabile et al. [11] defined three different areas in the HSV colour space:

- o The *achromatic* area: characterised by $s \leq 0.25$ or $v \leq 0.2$ or $v \geq 0.9$.
- o The *unstable chromatic* area: characterised by $0.25 \leq s \leq 0.5$ and $0.2 \leq v \leq 0.9$.
- o The *chromatic* area: characterised by $s \geq 0.5$ and $0.2 \leq v \leq 0.9$.

In order to obtain robustness to changes in external light conditions, these areas should be taken into consideration in the design of any colour segmentation system based on HSV colour space.

4.2.3 Colour Constancy

Colour constancy represents the ability of a visual system to recognise an object's true colour across a range of variations of factors extrinsic to the object, such as light conditions [74]. This definition summarises the purpose of colour constancy algorithms in generating illumination-independent descriptors of the scene colours measured in terms of the camera's RGB coordinates.

As mentioned earlier, the camera output is affected by the temperature of daylight (colour of the daylight), and the object's albedo.

By referring to equation (4.1) and assuming ideal sensors for the RGB light, and light source which illuminates the surface at a right angle, the aforementioned equation can be simplified to:

$$C(\mathbf{P}_s) = e(\lambda)c_b(\mathbf{P}_o, \lambda) ; \text{for } C = \{R, G, B\} \quad (4.5)$$

From the above mentioned equation, colour constancy can be achieved by independent scaling of the RGB colour bands, if it is assumed that the camera sensors are close to ideal [71].

Colour constancy is an important issue as far as detection and recognition of road signs are concerned. This is due to the illumination's variation, both in brightness and spectrum, in the outdoor scenes where the traffic signs are located.

A study by Funt et al. [75] showed that machine colour constancy algorithms are not good enough for colour-based object recognition. In spite of this disappointing result, many new algorithms were developed after Funt's paper, namely Finlayson [76], Tsin et al. [77], Rosenberg et al.[78] ,Török and Zarányi[79], Sridharan and Stone [74], Ebner [71, 80], Stanikunas et al.[81]. The community concerned with traffic sign recognition has not taken the colour constancy problem seriously, and hence these algorithms have not been tested and assessed yet. Thus, no one knows how applicable they are for traffic and road sign detection and recognition.

4.3 Moments and Invariants

Moments and moment functions are widely used in many applications such as image analysis, invariant pattern recognition, object classification and image reconstruction. They are useful for feature extraction because moments represent the global characteristic of the shape of an object in the image and they provide information about the geometrical features of that object. The main advantage of moments is that they are computationally very simple. Therefore, they have been used in several areas of computer vision and robotics. In the recent past, different types of moments have been introduced and each of them has its advantage in specific application areas.

4.3.1 Zernike Moments

Zernike Moments were introduced by Teague [82] based on orthogonal functions called Zernike polynomials. Although the calculation of Zernike moments is very complex, they proved to be superior in terms of their feature representation and low noise sensitivity.

Zernike moments are classed as orthogonal moments as they possess a useful rotation invariance property. Rotating the image does not change the magnitude of its Zernike moments. However, Zernike moments are translation and scale variant and to obtain an invariant version of these moments, a normalisation process is needed.

The kernel of Zernike moments is a set of orthogonal Zernike polynomials defined over the polar coordinate space inside a unit circle. The two-dimensional Zernike moments of order p with repetition q of an image intensity function $f(r, \theta)$ are defined as [83]

$$Z_{pq} = \frac{p+1}{\pi} \int_{\theta=0}^{2\pi} \int_{r=0}^1 V_{pq}^*(r, \theta) f(r, \theta) r dr d\theta \quad (4.6)$$

where $0 \leq r \leq 1$, p is the order of Zernike polynomials with repetition q and $V_{pq}(r, \theta)$ is defined as

$$V_{pq}(r, \theta) = R_{pq}(r) e^{jq\theta} \quad (4.7)$$

and the real-valued radial polynomial $R_{pq}(r)$ is given as follows [84]

$$R_{pq}(r) = \sum_{k=0}^{(p-|q|)/2} (-1)^k \frac{(p-k)!}{k! \left(\frac{p+|q|}{2} - k\right)! \left(\frac{p-|q|}{2} - k\right)!} r^{p-2k} \quad (4.8)$$

In this equation, p is a non negative value such that $p = 0, 1, 2, \dots, \infty$, q should be in the range $0 \leq |q| \leq p$ and $p - |q|$ is even, and $R_{p,-q}(r) = R_{pq}(r)$.

In its discrete form, Zernike moments of order p with repetition q of a discrete binary image with image intensity function $f(x, y)$ inside a unit circle is given by [85]

$$Z_{pq} = \frac{p+1}{\pi} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} V_{pq}^*(x, y) f(x, y), \quad x^2 + y^2 \leq 1 \quad (4.9)$$

where N is the number of pixels along each axis of the image, and the Zernike polynomials are given by

$$V_{pq}(x, y) = V_{pq}(r, \theta) = R_{pq}(r)e^{jq\theta} \quad (4.10)$$

And

$$r = \sqrt{x^2 + y^2}, \quad \theta = \tan^{-1}(y/x)$$

and $R_{pq}(r)$ is the radial polynomial given by equation (4.8).

A big advantage of Zernike moments is that they have a simple property. If Z_{pq} and Z'_{pq} are the p th order Zernike moments of an image $f(x, y)$ and $f'(x, y)$ rotated by an angle ϕ , respectively, then the moments of the two images are related by

$$Z'_{pq} = Z_{pq}e^{-jp\phi} \quad (4.11)$$

The magnitude of the Zernike moments is then given by

$$|Z'_{pq}| = |Z_{pq}e^{-jp\phi}| = |Z_{pq}| \quad (4.12)$$

Hence, the magnitude of the Zernike moments could be taken as a rotation invariant feature of the underlying image [86].

Zernike moments are computationally very expensive. Equation (4.8) has many factorial terms which are computationally very exhaustive. In addition to this, the moments themselves are scale and translation variant which need a normalisation process. Many methods have been suggested to reduce the time of computation by reducing the redundant terms in the Zernike radial polynomial [87, 88]. Nevertheless, the computation time of Zernike moments can be reduced dramatically by using the explicit form of $R_{pq}(r)$ [89] which is shown in Appendix C. The terms in Appendix C are used in this research to implement Zernike moments.

Moment-based features of an object are said to be translation invariant if they remain the same even when the object is located in different positions in the image. The translation invariance of Zernike moments is achieved by moving the origin of the image to the centroid of the object. This is done by transforming the image $f(x, y)$ into another image $f(x + x_{cen}, y + y_{cen})$ where (x_{cen}, y_{cen}) are the coordinates of object's centroid.

Scale invariance is achieved by scaling the original object through setting its zero-order geometric moment (area) to a certain value. This method is more suitable to

continuous space rather than digital images. In digital images, this process produces some errors due to re-sampling the image. An algorithm proposed by Kamila et. al. [85] eliminated this problem by avoiding any scaling of the original image and replacing that with calculating Zernike moments of the object centred at the origin of the image. Normalisation is then carried out by dividing the Zernike moments by the area of the object under consideration.

The modified Zernike moments which are rotation, translation and scale invariant, are carried out by the following steps:

1. To carry out the translation invariance, the binary image is normalised by calculating object's area and centroid (x_{cen}, y_{cen}) from the following equations:

$$a = \sum_x \sum_y f(x, y) \quad (4.13)$$

$$x_{cen} = \frac{1}{a} \sum_x x f(x, y) \quad (4.14)$$

$$y_{cen} = \frac{1}{a} \sum_y y f(x, y) \quad (4.15)$$

2. Calculate object's pixel coordinates of the new image shown in Figure 4.3:

$$x' = y - y_{cen}, \quad y' = x_{cen} - x \quad (4.16)$$

where (x', y') are the transformed coordinates and (x, y) are the original coordinates of the object.

3. Find the radius of the minimum circle containing the object by calculating the furthest object's pixel from the centroid (x_{cen}, y_{cen}) , denoted r_{min} , using Euclidean distance.

$$r_{min} = \sqrt{(x' - x_{cen})^2 + (y' - y_{cen})^2} \quad (4.17)$$

Use r_{min} to map the object's coordinates to the modified coordinates.

4. Map the coordinates of every object's pixel to be within a unit circle by calculating

$$x'' = x' / r_{min}, \quad y'' = y' / r_{min} \quad (4.18)$$

5. Calculate Zernike moments using x'' and y'' values achieved from the former step.
6. Calculate the modified Zernike moments z'_{pq} by dividing the Zernike moments by the area of the object.

$$z'_{pq} = z_{pq} / a \quad (4.19)$$

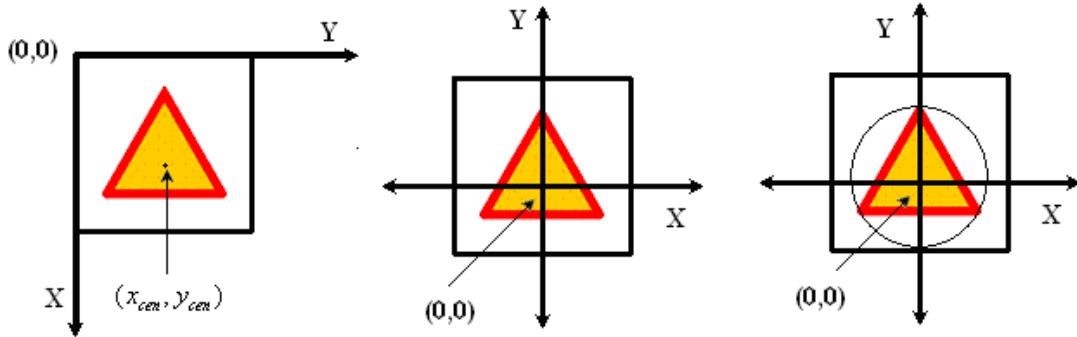


Figure 4.3: Main steps to perform translation and scaling invariance.

4.3.2 Legendre Moments

The set of Legendre moments was proposed by Teague [82] as a set of orthogonal moments for image analysis. Legendre moments are used in different applications such as pattern recognition, image indexing and face recognition.

The kernel of Legendre moments are the product of Legendre polynomials defined along rectangular image coordinate axes inside a unit circle. Legendre moments of order $(m+n)$ are defined as [90]

$$L_{mn} = \frac{(2m+1)(2n+1)}{4} \int_{-1}^1 \int_{-1}^1 P_m(x) P_n(y) f(x, y) dx dy \quad (4.20)$$

where $m, n = 1, 2, 3, \dots, \infty$ and $x, y \in [-1, 1]$. The n th order Legendre polynomials are defined as [83]:

$$P_n(x) = \sum_{k=0}^n (-1)^{(n-k)/2} \frac{1}{2^n} \frac{(n+k)! x^k}{\left(\frac{n-k}{2}\right)! \left(\frac{n+k}{2}\right)! k!} \quad (4.21)$$

where $|x| \leq 1$ and $(n-k)$ is even.

The above series expansion of Legendre polynomials can be obtained from the equation

$$P_n(x) = \frac{1}{2^n n!} \left(\frac{d}{dx} \right)^n \left[(1-x^2)^n \right] \quad (4.22)$$

The set of Legendre polynomials $P_n(x)$ forms a complete orthogonal basis set on the interval $[-1, 1]$, and the Legendre moments L_{mn} generalizes the geometric moments m_{pq} in the sense that the monomial $x^p y^q$ is replaced by the orthogonal polynomial $P_m(x) P_n(y)$ of the same order.

As mentioned in the previous discussion, the region of definition of Legendre polynomials is inside the interval $[-1, 1]$. An $N \times N$ pixel image with intensity $f(i, j)$ such that $0 \leq i, j \leq (N-1)$ should be scaled to fit the region $-1 \leq x, y \leq 1$. The discrete version of the Legendre moments can be given as [91]

$$L_{mn} = \frac{(2m+1)(2n+1)}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} P_m(x_i) P_n(y_j) f(i, j) \quad (4.23)$$

where x_i and y_j denote the normalised pixel coordinates in the range $[-1, 1]$ given by

$$x_i = \frac{2i}{N-1} - 1, \quad y_j = \frac{2j}{N-1} - 1 \quad (4.24)$$

To calculate the Legendre moments for digital binary images, a similar procedure to that used to calculate Zernike moments is invoked. The steps in this procedure are as follows

1. Find the centre of mass (x_{cen}, y_{cen}) of the object from equation (4.14) and (4.15).
2. Find the minimum bounding circle and calculate its radius denoted r_{min} from

$$r_{min} = \sqrt{(i - x_{cen})^2 + (j - y_{cen})^2}$$

Where $0 \leq i, j \leq (N-1)$ and (i, j) is the position of the current pixel.

3. Normalise the coordinates of the image such that $-1 \leq x_i, y_j \leq 1$ as follows

$$x_i = \frac{j - y_{cen}}{r_{min}}, \quad y_j = \frac{x_{cen} - i}{r_{min}} \quad (4.25)$$

4. Calculate Legendre moments for equation (4.36).

In this procedure, equation (4.38) differs from equation (4.37). The reason is that it is assumed in equation (4.37) that the object is located in an NxN pixels image and it fits the whole area of this image, while in equation (4.38) it is not necessary that the object fits the whole area. This means that the object will fit the unit circle only when it is inside the image.

4.3.3 Orthogonal Fourier-Mellin Moments

There is a large number of complete sets of polynomials which are rotation invariant and are orthogonal over the interior of a unit circle. Among them are the orthogonal Fourier-Mellin moments (OFMMs) which were introduced by Sheng and Arsenault [92] and later by Sheng and Shen [93] for character recognition.

The kernel of the orthogonal Fourier-Mellin moments is the set of orthogonal Mellin polynomials over the polar coordinates inside a unit circle. The two-dimensional OFMMs of order p with repetition q of an image intensity function $f(r, \theta)$ are defined as

$$\Phi_{pq} = \frac{1}{2\pi a_p} \int_0^{2\pi} \int_0^1 Q_p(r) e^{-jq\theta} f(r, \theta) r dr d\theta \quad (4.26)$$

where $0 \leq |q| \leq p$, $p \geq 0$, $j = \sqrt{-1}$.

The p th Mellin radial polynomials are given in the following forms:

$$Q_p(r) = \sum_{k=0}^p \alpha_{pk} r^k \quad (4.27)$$

with

$$\alpha_{pk} = (-1)^{p+k} \frac{(p+k+1)!}{(p-k)!k!(k+1)!} \quad (4.28)$$

It can be verified that the set $Q_p(r)$ is orthogonal over the range $0 \leq r \leq 1$

$$\int_0^1 Q_p(r) Q_s(r) r dr = a_p \delta_{ps}, \quad a_p = \frac{1}{2p+2} \quad (4.29)$$

where δ_{ps} is the Kronecker delta, and a_p is a normalisation constant with $r=1$ as a maximum radial size of the underlying object[89].

The discrete version of OFMMs can be expressed in rectangular coordinates (x, y) as

$$\Phi_{pq} = \frac{p+1}{\pi} \sum_x \sum_y f(x, y) Q_p(r) e^{-jq\theta} \Delta x \Delta y \quad (4.30)$$

$$x^2 + y^2 \leq 1, \quad x = r \cos \theta, \quad y = r \sin \theta$$

The OFMMs are integrable when the degree p of $Q_p(r)$ is $p \geq 0$.

If the image is rotated through an angle ϕ , the relationship between the moments of the two images are as follows

$$\Phi'_{pq} = \Phi_{pq} e^{-jq\phi} \quad (4.31)$$

Therefore, the magnitude $|\Phi_{pq}|$ of the OFMMs could be taken as a rotation invariant feature of the underlying image.

To calculate the orthogonal Fourier-Mellin moments for digital binary images, a similar procedure to that used to calculate Legendre moments is invoked.

4.3.4 Binary Haar Features

In many cases during image retrieval, the exact position and orientation of objects in an image are only of secondary value. Thus, it is desirable to have features which are invariant to certain transformations, say translation and rotation. Invariant image features based on integration over that transformation group which were introduced by Schulz-Mirbach [94] can be invoked for this purpose.

Let $I = \{I(i, j)\}, 0 \leq i < N, 0 \leq j < M$ be an image, with $I(i, j)$ representing the grey-value at the pixel coordinate (i, j) . Let G be the transformation group of translations and rotations with elements $g \in G$ acting on the image, such that the transformed image is gI . An invariant feature must satisfy $F(gI) = F(I), \forall g \in G$.

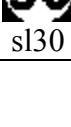
Original Image	(0, 1)	(1, 0)	(1, 1)	(0, 2)	(1, 2)	(2, 0)	(2, 1)	(2, 2)	(3, 3)	(4, 4)
 rcx										
 rcb										
 yield										
 warning										
 stop										
 rc										
 noentry										
 sl110										
 sl90										
 sl70										
 sl50										
 sl30										

Figure 4.4: Binary Haar features applied to different road signs.

Such invariant features can be constructed by integrating $f(gI)$ over the transformation group G .

$$F(I) = \frac{1}{|G|} \int_G f(gI) dg \quad (4.32)$$

For binary road sign images, transformations will be restricted to a certain group of translations. As a kernel function, binary operations among neighbour pixels are proposed. For example, a 2-point kernel evaluated at a point (x, y) would be

$$k(x, y) = I(x, y) \text{ XOR } I(x + \Delta_x, y + \Delta_y) \quad (4.33)$$

where the pair (Δ_x, Δ_y) determines the local support.

The i_{th} invariant feature is then given by

$$F_i = \frac{1}{MN} \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} \text{XOR}(I(x, y), I(x + \Delta_x^i, y + \Delta_y^i)) \quad (4.34)$$

where the pair (Δ_x^i, Δ_y^i) are the translation parameters for the i_{th} kernel. The boundary pixels may optionally be discarded if the corresponding translated point falls outside the image.

Theoretically, the values of (Δ_x^i, Δ_y^i) are $0 \leq (\Delta_x^i, \Delta_y^i) < \infty$, but practically they should not exceed the size of the image. Figure 6.17 illustrates the features which are essentially discriminative for this classification task. The first column is the original image. The other columns depict the result of XORing of the original image with its translated version by the amount mentioned in the first row.

4.4 Support Vector Machines

Support Vector Machines (SVM) are pattern classification and regression techniques based on mathematical foundations of statistical learning theory. This theory was first proposed by Vapnik and his group at AT&T Bell Laboratories in 1992 [95]. The SVM learns a separating hyperplane to maximise the margin and to produce good generalisation ability. Due to the good generalisation performance on a large number of real-life data and due to the fact that the approach is properly motivated, it has been used for a wide range of applications.

It has been shown to correspond to a linear method in a high-dimensional feature space nonlinearly related to the input space. High-dimensionality usually leads to expensive computation steps and therefore to an algorithm unsuitable for time critical applications. By using Kernels, the so-called kernel trick, SVM algorithms do not need to make computations in the high dimensional space. All necessary computations can be done in the input space, which significantly reduces the complexity.

4.4.1 Linear Classification with Maximal Margin Classifier

Suppose there are two given classes of objects. The goal is to find a plane that separates the two classes and helps to assign new objects to the right class. The simple case, with only two different classes is binary classification. Figure 4.5 illustrates an example of a pattern recognition problem with two classes. Clearly the two classes are linearly separable. Therefore, by finding a suitable plane, the training data can be successfully separated. However, the choice of the hyperplane is not trivial, as is demonstrated in Figure 4.5-b and 4.5-c.

Even though the lines in both of these figures separate the two classes without a single training error, they may not perform equally well on unseen test data. Figure 4.5-d must be the most promising or even optimal choice.

The basic idea of SVM can therefore be described as follows: Find the optimal hyperplane to linearly separate two classes. The optimal hyperplane is a maximum-margin hyperplane, that is, a hyperplane which separates the two classes and is equidistant from both. To obtain the maximal margin, one needs to solve a quadratic optimisation problem.

As can be seen in Figure 4.6, using the optimal hyperplane, the margin intersects a few training patterns from each class. These training patterns are called Support Vectors (SV). Solely they determine the position of the hyperplane. All other patterns have no influence on the calculation of the hyperplane.

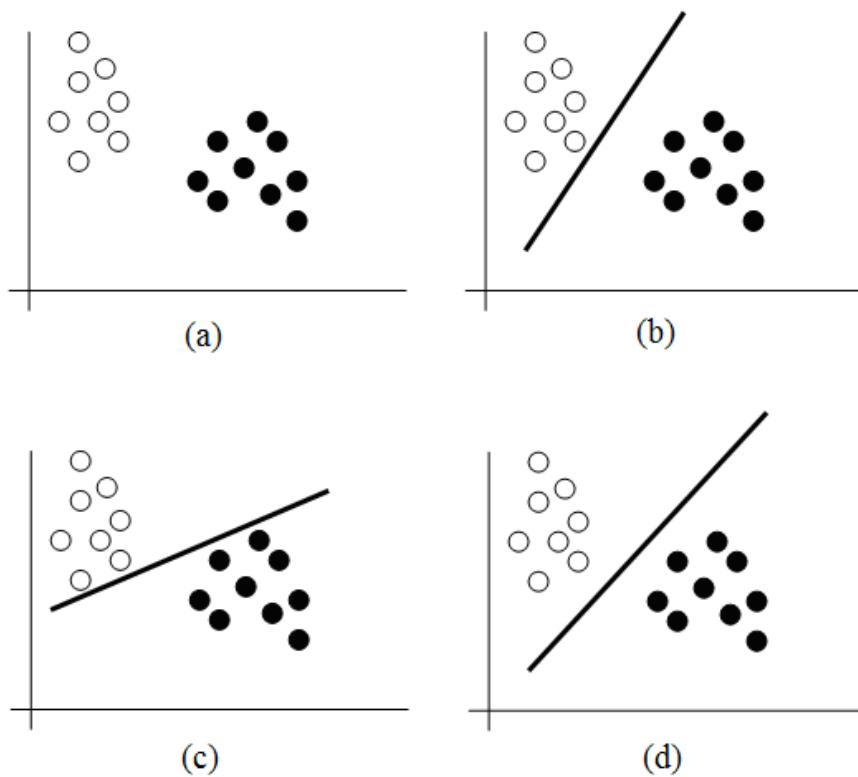


Figure 4.5: (a) Pattern recognition problem with two classes. (b)(c) Separating hyperplane separates the two classes without training errors. (d) Optimal separating hyperplane.

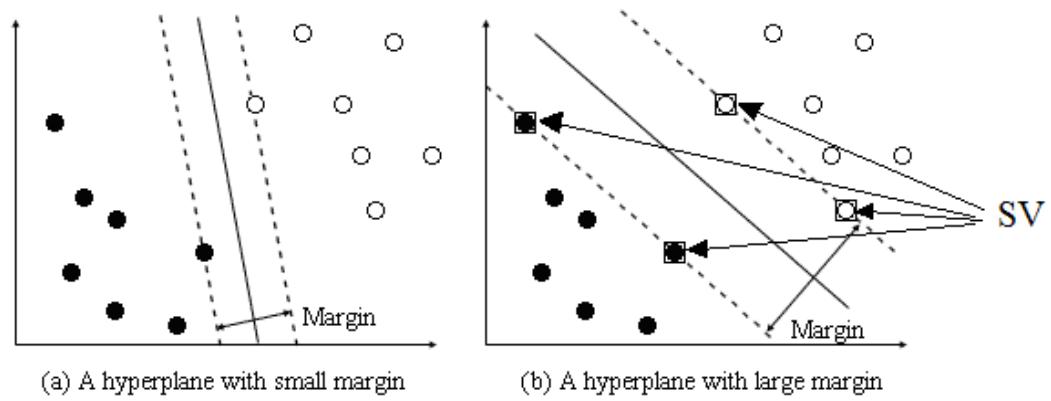


Figure 4.6: Optimal solution can be obtained by maximizing the margin.

In a binary classification problem the training data is given as a data set S of points $\mathbf{x}_i \in \Re^d$ with the label $y_i \in \{-1, +1\}$, for all training data $i = 1, \dots, l$, where l is the number of training examples, and d is the dimension of the problem. When training SVM, the goal is to construct a separating hyperplane as the decision plane, which successfully separates the positive (+1) and the negative (-1) classes with the largest margin.

When the two classes are linearly separable in \Re^d , the goal is to find the maximal-margin hyperplane. The margin can be seen as twice the distance from the hyperplane to the nearest of the positive and negative examples (see Figure 4.7). The data points closest to the margins are called the Support Vectors (SV). They are the most important training points, since they finally determine the position of the hyperplane[96].

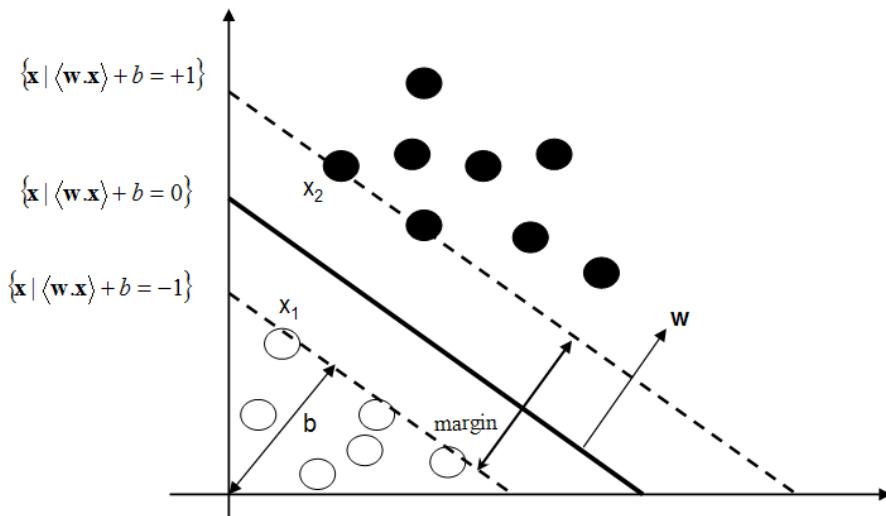


Figure 4.7: Linear classification for two-dimensional input vectors.

Linear classification is normally performed by using a linear function of its input vectors. This function can be written as

$$f(\mathbf{x}) = \langle \mathbf{w} \cdot \mathbf{x} \rangle + b = \sum_{i=1}^l w_i x_i + b \quad (4.35)$$

where x_i is the i th attribute value of an input vector \mathbf{x} , w_i is the weight value for the attribute x_i and b is the bias. For a binary classification, the decision rule is given by

$$f(\mathbf{x}) = \text{sign}(\langle \mathbf{w}, \mathbf{x} \rangle + b) \quad (4.36)$$

The hyperplane can be defined as

$$\langle \mathbf{w}, \mathbf{x} \rangle + b = 0 \quad \mathbf{w} \in \Re^d, b \in \Re \quad (4.37)$$

The input vector $\mathbf{x} = (x_1, \dots, x_l)'$ is assigned into the positive class if $f(\mathbf{x}) \geq 0$, otherwise it is assigned into the negative class.

A hyperplane with a margin of width= 1 is called a canonical hyperplane. A separating hyperplane in canonical form must satisfy the following constraints

$$\gamma_i = y_i (\langle \mathbf{w}, \mathbf{x}_i \rangle + b) \geq 1, \quad i = 1, \dots, l \quad (4.38)$$

If $d(\mathbf{w}, b; \mathbf{x}_i)$ is defined as a distance of a point x from the hyperplane (\mathbf{w}, b) and it is given by:

$$d(\mathbf{w}, b; \mathbf{x}_i) = \frac{|\langle \mathbf{w}, \mathbf{x}_i \rangle + b|}{\|\mathbf{w}\|} \quad (4.39)$$

Then, the margin is given by

$$\gamma(\mathbf{w}, b) = \min_{x_i, y_i = -1} \frac{|\langle \mathbf{w}, \mathbf{x}_i \rangle + b|}{\|\mathbf{w}\|} + \min_{x_i, y_i = 1} \frac{|\langle \mathbf{w}, \mathbf{x}_i \rangle + b|}{\|\mathbf{w}\|} \quad (4.40)$$

$$\gamma(\mathbf{w}, b) = \frac{1}{\|\mathbf{w}\|} \left(\min_{x_i, y_i = -1} |\langle \mathbf{w}, \mathbf{x}_i \rangle + b| + \min_{x_i, y_i = 1} |\langle \mathbf{w}, \mathbf{x}_i \rangle + b| \right) = \frac{2}{\|\mathbf{w}\|} \quad (4.41)$$

The optimal hyperplane can then be found by maximising the margin which leads to the following optimisation problem:

$$\min \tau(w) = \frac{\|\mathbf{w}\|^2}{2} \quad (4.42)$$

under the constraint

$$y_i (\langle \mathbf{w}, \mathbf{x}_i \rangle + b) - 1 \geq 0, \quad \forall i \quad (4.43)$$

The optimal separating hyperplane can thus be found by solving the Quadratic Programming problem in (4.42) and (4.43). However, it turns out that it is more suitable to work with the dual problem of the above optimisation problem. To derive it, the introduction of Lagrangian is necessary

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{\|\mathbf{w}\|^2}{2} - \sum_{i=1}^l \alpha_i (y_i (\langle \mathbf{w}, \mathbf{x}_i \rangle + b) - 1) \quad (4.44)$$

where $\alpha_i \geq 0$ are Lagrange multipliers and $\boldsymbol{\alpha} = \{\alpha_i\}_{i=0}^l$. L needs to be minimised with respect to the primal variable \mathbf{w} and b and maximised with respect to the dual variable α_i . Therefore, in order to calculate the derivatives with respect to \mathbf{w} and b :

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\alpha})}{\partial b} = \sum_{i=1}^l y_i \alpha_i = 0 \quad (4.45)$$

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\alpha})}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^l y_i \alpha_i \mathbf{x}_i = 0 \quad (4.46)$$

The corresponding Karush-Kuhn-Tucker (KKT) [97] complementary conditions are given in (4.47). The KKT is necessary to be able to compute the parameter b , which is, other than \mathbf{w} , not explicitly determined by the training procedure.

$$\alpha_i [y_i (\langle \mathbf{w}, \mathbf{x}_i \rangle + b) - 1] = 0 \quad (4.47)$$

Rewriting equation (4.44),

$$\begin{aligned} L(\mathbf{w}, b, \boldsymbol{\alpha}) &= \frac{\|\mathbf{w}\|^2}{2} - \sum_{i=1}^l y_i \alpha_i \langle \mathbf{w}, \mathbf{x}_i \rangle - b \sum_{i=1}^l y_i \alpha_i + \sum_{i=1}^l \alpha_i \\ L(\mathbf{w}, b, \boldsymbol{\alpha}) &= \frac{1}{2} \mathbf{w}^T \mathbf{w} - \sum_{i=1}^l y_i \alpha_i \mathbf{w}^T \mathbf{x}_i - b \sum_{i=1}^l y_i \alpha_i + \sum_{i=1}^l \alpha_i \end{aligned} \quad (4.48)$$

By substituting equations (4.45) and (4.46) in the right hand side of (4.48), the primal variables vanish and we obtain

$$\begin{aligned}
L(\mathbf{w}, b, \mathbf{a}) &= \frac{1}{2} \left(\sum_{i=1}^l y_i \alpha_i \mathbf{x}_i \right)^T \left(\sum_{j=1}^l y_j \alpha_j \mathbf{x}_j \right) - \sum_{i=1}^l y_i \alpha_i \left(\sum_{j=1}^l y_j \alpha_j \mathbf{x}_j \right)^T \mathbf{x}_i + \sum_{i=1}^l \alpha_i \\
L(\mathbf{w}, b, \mathbf{a}) &= -\frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l y_i y_j \alpha_i \alpha_j \mathbf{x}_i^T \mathbf{x}_j + \sum_{i=1}^l \alpha_i
\end{aligned} \tag{4.49}$$

which leads to the following quadratic programming problem

$$\text{maximise } W(\alpha) = -\frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l y_i y_j \alpha_i \alpha_j \mathbf{x}_i^T \mathbf{x}_j + \sum_{i=1}^l \alpha_i \tag{4.50}$$

$$\text{subject to } \sum_{i=1}^l y_i \alpha_i = 0, \quad \alpha_i \geq 0 \tag{4.51}$$

The dual objective function $W(\alpha)$ has a quadratic form which can be solved using different optimisation methods. One particular method for training SVMs is the Sequential Minimal Optimisation (SMO) algorithm [98], which breaks the large quadratic programming problems into a series of small possible QP problems. The small QP problems are then solved analytically.

During the maximisation process, most of the dual variables α_i become zero. The remaining non-zero parameters are the Support Vectors.

The solution for \mathbf{w} can be derived from equation (4.46)

$$\mathbf{w} = \sum_{i=1}^l y_i \alpha_i \mathbf{x}_i \tag{4.52}$$

Instead of minimizing (4.42) it is preferable to maximise the equivalent dual function mainly for two reasons: only a quadratic form has to be maximised (with respect to α_i) and the data vectors \mathbf{x}_i appear in the form of dot products $\langle \mathbf{x}_i \cdot \mathbf{x}_j \rangle$ only, which will be important when introducing the kernel functions.

This means that \mathbf{w} , which describes the separating hyperplane, is determined by the sum of the input vector weighted with the Lagrangian multipliers α_i . Since the patterns

closest to the hyperplane (the support vectors) only obtain non-zero values for α_i , they solely determine \mathbf{w} , and thus the separating hyperplane. All the remaining examples of the training set have no influence on equation (4.52).

The threshold value b can be computed using the KKT in (4.51). It can for instance be obtained by averaging

$$b = y_i - \sum_{i=0}^l y_j \alpha_i (\mathbf{x}_i \cdot \mathbf{x}_j) \quad (4.53)$$

for all points with $\alpha_i > 0$; in other words, all Support Vectors.

The classification of a new pattern \mathbf{x} can now be obtained by solving the decision function $f(\mathbf{x})$

$$f(\mathbf{x}) = \text{sign}(\langle \mathbf{w}, \mathbf{x} \rangle + b) = \text{sign}\left(\sum_{i=1}^l y_i \alpha_i (\mathbf{x} \cdot \mathbf{x}_i) + b\right) \quad (4.54)$$

4.4.2 Non-Linear Classification

In real-life problems often the classes involved are not linearly separable. Figure 4.8 depicts such a problem where two classes overlap each other. Even if it is impossible to derive a separating hyperplane in the input space, it might be possible to do so in some high-dimensional feature space. The input data is mapped into a higher-dimensional feature space, where the classification can be performed by a linear SVM. Finally by transforming back into the input space, a non-linear decision surface is obtained, which successfully separates the two classes.

The problem with high-dimensionality is that computations may become intractable. To avoid this, the kernel trick is invoked. To compute the separating hyperplane, a dot product is mainly used as a similarity measure. Instead of actually performing the mapping into the feature space and computing dot products there, it is possible to use kernel functions which combine both steps. The ingenuity of the kernel trick is that mapping is not actually performed. The similarity value is in fact obtained from the kernel function as it would be first mapped and then the dot product is performed.

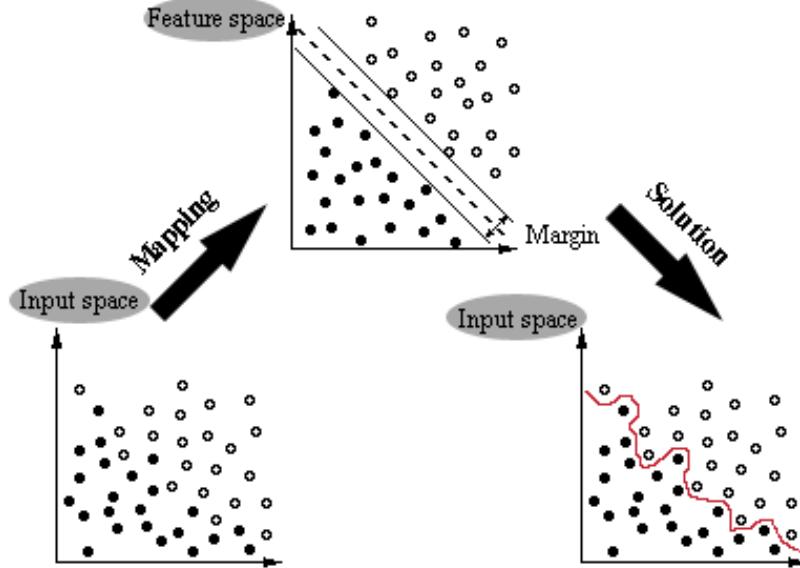


Figure 4.8: A non-linear problem. It is impossible to linearly separate the patterns in the input space.

Linear learning solves problems by linear functions. However, a simple linear function defined by the given attributes cannot achieve a target task in a flexible manner. There are two main limitations of linear learning:

- ✧ First, the functions that are tried to learn may not have a simple representation and may not be easily verified in this way.
- ✧ Second, normally the training data are noisy and so there is no guarantee that there is an underlying function which correctly classifies the training data.

Therefore, complex real world problems require more expressive hypothesis spaces than linear functions. The following sections discuss a method which constructs a non-linear machine to classify the data more flexibly.

4.4.3 Learning in Feature Space

The complexity of the target function to be learned depends on the way it is represented, and the difficulty of the learning task can vary accordingly. Kernel representations offer a solution by constructing a mapping from the input space to a high dimensional feature space to increase the power of linear learning for complex applications.

Figure 4.9 shows an example of a mapping from a two-dimensional input space to a two-dimensional feature space. In the input space the data cannot be separated by a linear function. A feature mapping simplifies the classification task since the data in the feature space is linearly separable.

The quantities introduced to describe the data are usually called features, while the original quantities are sometimes called attributes. The task of choosing the most suitable representation is known as feature selection. The space X is referred to as the input space, while $F = \{\phi(x) : x \in X\}$ is called feature space.

By selecting an appropriate kernel function, a non-linear mapping is performed between the input space and high dimensional feature space. This means each input vector in the input space matches a feature vector in the feature space. However, this mapping does not increase the number of the tuneable parameters. This technique also overcomes the problem of dimensionality in both computation and generalisation [96].

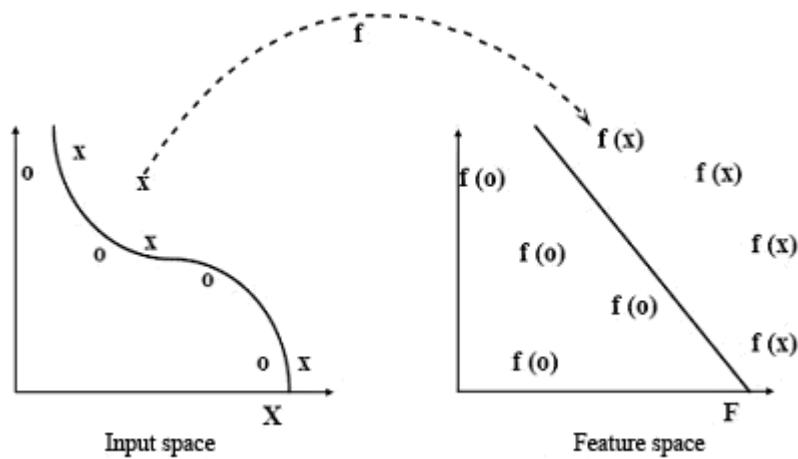


Figure 4.9: A mapping from a two-dimensional input space to a two-dimensional feature space.

4.4.4 Implicit Mapping to Feature Space

The implicit mapping from input space into feature space expresses the data in a new representation in which the nonlinear decision function in the input space is equivalent to a linear function in the feature space. In such a way the linear learning machine can be used. The function (4.35) which was represented in linear learning section is modified as:

$$f(\mathbf{x}) = \langle \mathbf{w} \cdot \phi(\mathbf{x}) \rangle + b = \sum_{i=1}^l w_i \phi_i(\mathbf{x}) + b \quad (4.55)$$

where $\phi(\mathbf{x}) : X \rightarrow F$ is the mapping function from the input space to some feature space. Therefore, there are two steps to constructing a non-linear machine: first, a fixed non-linear mapping transforms the data from input space into a feature space; second, in the feature space a linear machine is used to classify them.

One important feature of linear machines is that the decision function can be expressed as a linear combination of the training points so that the decision rule can be evaluated using just the inner products between the test points and the training points as follows:

$$f(\mathbf{x}) = \sum_{i=1}^l \alpha_i y_i \langle \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}) \rangle + b \quad (4.56)$$

It is possible to merge the previous two steps to build a non-linear learning machine in which the inner product $\langle \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}) \rangle$ can be computed directly in the feature space as a function of the original input points.

4.4.5 Kernels

A kernel is a function K , such that for all $\mathbf{x}, \mathbf{z} \in X$

$$K(\mathbf{x}, \mathbf{z}) = \langle \Phi(\mathbf{x}) \cdot \Phi(\mathbf{z}) \rangle$$

where Φ is a mapping from X to an (inner product) feature space F [99].

A kernel constructs an implicit mapping from the input space into a feature space and a linear machine is trained in the feature space. The Gram matrix or kernel matrix describes the training data information in the feature space. The key to this approach is to find a kernel function to be evaluated efficiently. The decision rule can be evaluated using this kernel function by at most ℓ times of the kernel:

$$f(\mathbf{x}) = \sum_{i=1}^l \alpha_j y_j K(\mathbf{x}_i, \mathbf{x}) + b \quad (4.57)$$

4.4.6 Properties of Kernels

Kernel functions are used to avoid the feature space in the computation of inner products. In order to ensure that the kernel function is useful for the feature space, the kernel function must be symmetric [99]

$$K(\mathbf{x}, \mathbf{z}) = \langle \Phi(\mathbf{x}) \cdot \Phi(\mathbf{z}) \rangle = \langle \Phi(\mathbf{z}) \cdot \Phi(\mathbf{x}) \rangle = K(\mathbf{z}, \mathbf{x}) \quad (4.58)$$

and satisfy the inequality which follows from the Cauchy-Schwartz inequality

$$K(\mathbf{x}, \mathbf{z})^2 = \langle \Phi(\mathbf{x}) \cdot \Phi(\mathbf{z}) \rangle^2 \leq \|\Phi(\mathbf{x})\|^2 \|\Phi(\mathbf{z})\|^2$$

$$K(\mathbf{x}, \mathbf{z})^2 = \langle \Phi(\mathbf{x}) \cdot \Phi(\mathbf{x}) \rangle \langle \Phi(\mathbf{z}) \cdot \Phi(\mathbf{z}) \rangle = K(\mathbf{x}, \mathbf{x}) K(\mathbf{z}, \mathbf{z}) \quad (4.59)$$

4.4.7 Examples of Kernels

To date many kernels have been proposed by researchers. The most frequently used kernels are:

1. Linear

The linear kernel is the simplest linear model,

$$K(\mathbf{x}, \mathbf{z}) = \langle \mathbf{x}, \mathbf{z} \rangle \quad (4.60)$$

2. Polynomial

Polynomial mapping is a popular method for non-linear modelling,

$$K(\mathbf{x}, \mathbf{z}) = \langle \mathbf{x}, \mathbf{z} \rangle^d \quad (4.61)$$

A more preferable expression is given by,

$$K(\mathbf{x}, \mathbf{z}) = (\gamma \langle \mathbf{x}, \mathbf{z} \rangle + r)^d \quad (4.62)$$

Where γ, r, d are kernel parameters and $\gamma > 0$.

3. Gaussian Radial Basis Function

The radial basis function is one of the kernels which has been given significant attention. The form of the Gaussian Radial Basis Function (GRBF) is,

$$K(\mathbf{x}, \mathbf{z}) = \exp(-\gamma \|\mathbf{x} - \mathbf{z}\|^2) \quad (4.63)$$

The RBF kernel has fewer numerical difficulties. Comparing the GRBF kernel with the polynomial kernel, the value of $K(\mathbf{x}, \mathbf{z})$ in the GRBF kernel is in the interval of $[0, 1]$, while the value of $K(\mathbf{x}, \mathbf{z})$ in the polynomial kernel is in the interval $[0, \infty)$.

4. Exponential Radial Basis Function

The form of the Exponential Radial Basis Function (ERBF) is,

$$K(\mathbf{x}, \mathbf{z}) = \exp(-\gamma \|\mathbf{x} - \mathbf{z}\|) \quad (4.64)$$

It produces a piecewise linear solution which can be attractive when discontinuities are acceptable.

5. Sigmoid

The Sigmoid function is so far the most common form of activation function used in artificial neural networks. It is a strictly increasing function which exhibits a graceful balance between linear and nonlinear behaviour. A SVM model using a sigmoid kernel function is equivalent to a two-layer, feed forward neural network,

$$K(\mathbf{x}, \mathbf{z}) = \tanh(\gamma \langle \mathbf{x}, \mathbf{z} \rangle + r) \quad (4.65)$$

4.4.8 Soft Margin Classifier

A maximal margin classifier can only be used for data which is linearly separable in the feature space. To satisfy real-world problems where the data are not linearly separable in the feature space, a soft margin classifier which tolerates noise and outliers is extended from a maximal margin classifier.

Revising the constraints equation (4.43) by adding a slack variable ξ_i , the new constraints are,

$$y_i (\langle \mathbf{w} \cdot \mathbf{x}_i \rangle + b) + \xi_i \geq 1 \quad \forall i \quad (4.66)$$

where

$$\xi_i \geq 0 \quad \forall i$$

In this case some data are allowed to be misclassified; Figure 4.10 depicts an example of a soft margin classifier.

The optimisation problem for soft margin classifier is,

$$\text{Minimise}_{\xi, \mathbf{w}, b} \quad \langle \mathbf{w} \cdot \mathbf{w} \rangle + C \sum_{i=1}^{\ell} \xi_i^k \quad (4.67)$$

where C and k define the cost of constraint violation. For positive integers k , the above optimisation problem is a convex programming problem. If $k = 1$ or $k = 2$, it will be a quadratic programming (QP) problem. C is the upper bound of α , it is a trade-off between maximum margin and classification error. A higher value of C gives a larger

penalty for classification error, and this means α is allowed to have a large value. Particular data maybe classified correctly by increasing the α value.

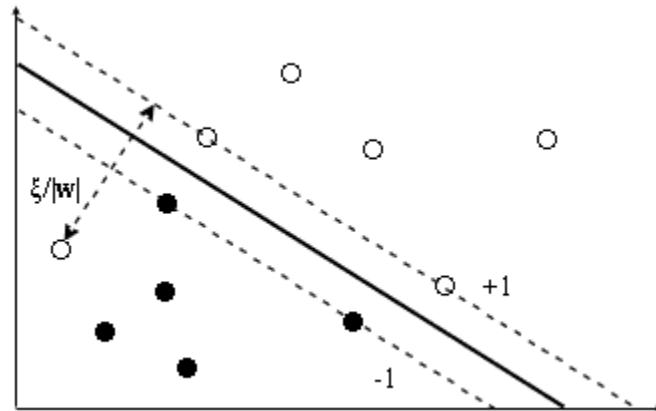


Figure 4.10: An example of soft margin classifier.

Applying the dual formulation, the 1-norm soft margin optimisation problem is

$$\text{Maximise}_{\alpha} \quad \sum_{i=1}^{\ell} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{\ell} y_i y_j \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j) \quad (4.68)$$

$$\text{Subject to} \quad \sum_{i=1}^{\ell} y_i \alpha_i = 0 \quad C \geq \alpha_i \geq 0 \quad (4.69)$$

4.4.9 Multi-class Classifier

So far the classification problems which have been discussed are binary classification problems. Some methods have been proposed to solve multi-class problems. Here one of them which is most widely used will be briefly introduced.

Among the existing multi-class approaches, “one against one” is one of the most suitable methods for practical problems. It constructs $k(k-1)/2$ classifiers where k is the number of categories. Each classifier trains data from two different classes, m th and n th classes solving the following binary classification problem,

$$\text{Minimise}_{\xi, \mathbf{w}, b} \quad \frac{1}{2} \langle \mathbf{w}^{mn} \cdot \mathbf{w}^{mn} \rangle + C \sum_{i=1}^{\ell} \xi_i^{mn} \quad (4.70)$$

$$\text{Subject to} \quad y_i \left(\langle \mathbf{w}^{mn} \cdot \mathbf{x}_i \rangle + b^{mn} \right) + \xi_i^{mn} \geq 1, \text{ if } \mathbf{x}_i \text{ in the } m\text{th class} \quad (4.71)$$

$$\text{Subject to } y_i \left(\langle \mathbf{w}^{mn} \cdot \mathbf{x}_i \rangle + b^{mn} \right) - \xi_i^{mn} \leq -1, \text{ if } \mathbf{x}_i \text{ in the } k\text{th class} \quad (4.72)$$

where $\xi_i^{mn} \geq 0$.

A voting strategy is used in this approach: each binary classification is considered to be a vote where votes can be cast for all data points \mathbf{x} . The point \mathbf{x} is assigned into a class with the maximum number of votes. If there are two classes with the same number of votes, a simple strategy is to select the one with the smallest index.

4.4.10 Types of SVM

1. C-SVM Classification

Given training vector $\mathbf{x}_i \in \Re^n$, C-SVC solves the following primal problem for binary classification $y_i \in \{-1, 1\}$ [95, 100]:

$$\text{Minimise}_{\xi, \mathbf{w}, b} \quad \frac{1}{2} \langle \mathbf{w} \cdot \mathbf{w} \rangle + C \sum_{i=1}^{\ell} \xi_i \quad (4.73)$$

$$\text{Subject to } y_i (\mathbf{w} \cdot \Phi(\mathbf{x}_i) + b) + \xi_i \geq 1 \quad \xi_i \geq 0, i = 1, \dots, \ell \quad (4.74)$$

Its dual formulation is

$$\text{Maximise} \quad \sum_{i=1}^{\ell} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{\ell} y_i y_j \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j) \quad (4.75)$$

$$\text{Subject to } \sum_{i=1}^{\ell} y_i \alpha_i = 0 \quad 0 \leq \alpha_i \leq C \quad (4.76)$$

where C is the upper bound of α_i . The decision function is

$$\text{sgn} \left(\sum_{i=1}^{\ell} y_i \alpha_i K(\mathbf{x}_i, \mathbf{x}) + b \right) \quad (4.77)$$

2. ν -SVM Classification

The ν -SVM was introduced by Schölkopf et al. [101]. The ν -support vector classification introduces a new parameter ν which controls the number of support vectors and margin of errors. Schölkopf proved that ν actually sets the upper bound for the number of margin errors and the lower bound for the number of support vectors. Chang and Lin [102, 103] investigated the relation between ν -SVM and C-SVM. In particular, they showed that solving the two approaches amounts to solving two different

problems with the same optimal solution. Increasing the parameter C in C-SVM is equivalent to decreasing ν in ν -SVM.

In Schölkopf's ν -SVM it turns out that as ν becomes smaller, the total number of support vectors decreases. When ν is decreased, fewer training errors are allowed, therefore the hyperplane is created in a way that it fits as many training examples as possible.

By increasing ν , more training errors take place and the margin of the hyperplane is increased and thus more examples lie within the margin and are considered as support vectors. Due to this, the model of the separating hyperplane is simplified. The interval of $\nu \in [0, 1]$ is much easier than $C \in [0, \infty]$.

The primal problem for this approach to solve binary classification is

$$\text{Minimise}_{\xi, \mathbf{w}, b} \quad \frac{1}{2} \langle \mathbf{w} \cdot \mathbf{w} \rangle - \nu \rho + \frac{1}{\ell} \sum_{i=1}^{\ell} \xi_i \quad (4.78)$$

$$\text{Subject to} \quad y_i (\mathbf{w} \cdot \Phi(\mathbf{x}_i) + b) + \xi_i \geq \rho \quad \xi_i \geq 0, i = 1, \dots, \ell, \rho \geq 0 \quad (4.79)$$

Its dual formulation is

$$\text{Maximise} \quad \frac{1}{2} \sum_{i,j=1}^{\ell} y_i y_j \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j) \quad (4.80)$$

$$\begin{aligned} \text{Subject to} \quad & \sum_{i=1}^{\ell} y_i \alpha_i = 0 \quad 0 \leq \alpha_i \leq \frac{1}{\ell} \\ & \sum_{i=1}^{\ell} \alpha_i \geq \nu \end{aligned} \quad (4.81)$$

4.5 Summary

A large number of articles have been published in the field of road sign recognition in the last ten years and a substantial number of them have been reviewed in this research. Detection of signs is achieved by using colour information, shape information or a mixture of the two. The recognition of signs, however, is carried out mostly by neural network or template matching. Other techniques are also employed for classification.

This chapter focuses on the study of colour as key information in the recognition of traffic signs. Since hue is the core of many colour segmentation algorithms in the reviewed articles, it is also studied in depth.

The shape of a traffic sign represents another key factor which plays a central role in traffic sign recognition. This chapter presents different methods to calculate features from the shape of the traffic signs. Several types of features and invariants are presented and discussed together with the method to make these features invariant to the in-plane transformations.

The SVM is presented as a classifier to be used in traffic sign recognition in this research. In this context, linear and non-linear classification is presented together with a discussion of the way to convert a non-linear problem to a linear one. Several types of SVM classifiers, such as the maximum margin classifier, the soft margin classifier, and multi-class classification are also discussed.

The next chapter shows how the various techniques may be brought together in designing an integrated recognition system that covers the image database, the design of the colour segmentation algorithms, the recognition system and the classification stage.

5. Road Sign Recognition System Design

This chapter examines the practical aspects and experimental setup which cover the algorithms developed in this research. The chapter is divided into 8 sections and a summary. Section 5.1 presents an overview of the Road Sign Recognition System (RSRS) which comprises several modules working together to achieve recognition. In section 5.2 the camera used for data collection is described and in section 5.3 the raw images database created for the purpose of developing and testing the different algorithms in this research are depicted. Section 5.4 presents the colour segmentation algorithms which are developed for traffic sign applications in outdoor images. Section 5.6 describes the Fuzzy shape recogniser invoked to recognise the traffic signs based on the combination of colours and shapes. Section 5.7 illustrates the training database used for the training and validation of the classification using the SVM classifier. Finally, section 5.8 presents the classification of traffic signs based on the SVM classifier and the training and testing normalised image library.

5.1 System Overview

A system to detect and recognise road and traffic signs should be able to work in two modes; the training mode in which a database can be built by collecting a set of traffic signs for training and validation, and a prediction mode in which the system can recognise a traffic sign which has not been seen before. A system to recognise road and traffic signs is depicted in Figure 5.1. It consists of a number of modules which work together to perform this recognition. These modules are as follows:

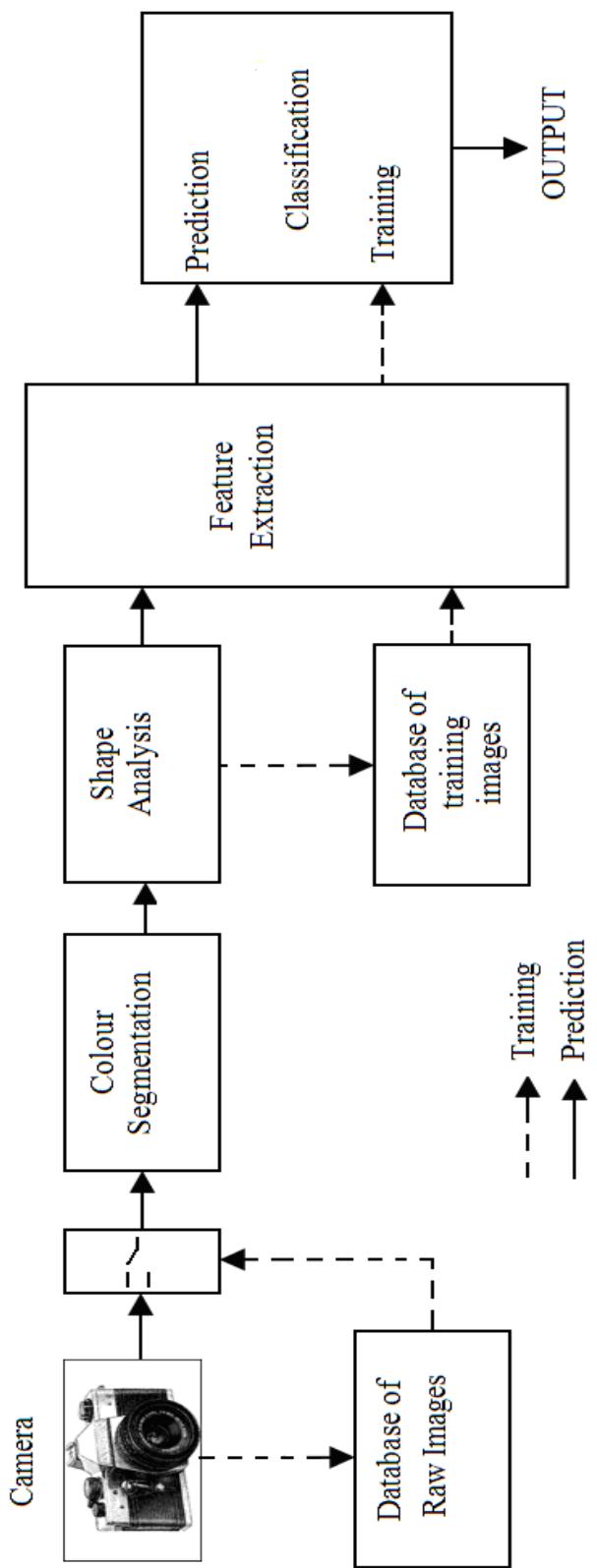


Figure 5.1: A block diagram of the RSRS.

a. The Camera:

A good digital camera which gives clear and sharp still images with different sizes is necessary. No special equipment is needed for this purpose. Images collected by this camera are used in later stages to develop and validate the colour segmentation algorithm, the recognition stage, and to build the classification system.

b. Colour Segmentation:

Colour segmentation is an important step to eliminate all background objects and unimportant information in the image. It generates a binary image containing the road signs and any other objects similar to the colour of the road sign. This step reduces the amount of calculation needed in the following steps as it radically reduces the number of probable objects. A colour segmentation algorithm should be robust enough to work in a wide spectrum of environmental conditions and be able to generate binary images even when traffic sign colours are attenuated.

c. Shape Analysis:

The main task of this module includes cleaning the binary image from noise and small objects, applying connected components labelling algorithm, and recognising the traffic sign. This module normalises the recognised traffic sign so that it becomes invariant to the in-plane transformations. This means that the resultant sign has a fixed size and it is located in a standard position where its centre of gravity is located in the centre of the image. This module works in two modes; the training and the prediction mode. In the training mode it is invoked to create or update the training image database. In the prediction mode it prepares every object in the binary image to be in standard format and ready for feature extraction.

d. Raw Image Database

This database is simply a collection of traffic scenes gathered by the camera. Images in this database are categorised according to the type of the sign, condition of the sign such as occluded, damaged, or faded, weather conditions, light geometry, and defects in the images such as blurring. The database represents the main source of images from which another database called the “training database” is built. The number of images in this database should be extensive enough to cover the large

variety of traffic signs which are used for training the system. In addition to this, the number of images containing the same type of sign should be vast to provide a good number of examples for training. An Access database program manages the images and the information about them so that selecting images with certain characteristics is straightforward.

e. Training Database

The training database consists of binary images of a normalised size such as 36x36 pixels. The database is created and updated in the training mode in such a way that binary images of the desired traffic signs are selected from a set of images. This database is used either directly or by extracting some features to train and validate the classifier. In the prediction mode, the database is invoked to train the classifier before any classification task takes place.

f. Feature Extraction:

This module contains algorithms which are used to extract features from either the training images in the training database or images directly from the shape analysis unit. It allows the classifier to be trained by either binary images or by features. Among the features which can be used are geometric moments, Zernike moments, Legendre moments, Orthogonal Fourier-Mellin Moments and Binary Haar features.

g. Classification:

Classification is carried out using a Support Vector Machine (SVM) described in Section 4.4.

5.2 The Camera

In this research a Minolta DiMAGE 7Hi camera shown in Figure 5.2 was used for data collection. It was selected because of the price and the specification which could be gained. The camera has a 5 mega pixel resolution with 7X optical zoom and 2X digital zoom capability. The full specifications of this camera are listed in Table 5.1.

Table 5.1: Specifications of Minolta DiMAGE 7Hi.

Features	Specifications
Lens	Minolta GT Lens, 7.2 – 50.8mm, $\phi 49\text{mm}$
Image formats	<ul style="list-style-type: none"> • RAW • Super Fine (TIFF) • Extra Fine (JPEG) • Fine (JPEG) • Standard (JPEG)
Max Resolution	2560 x 1920 (5 Megapixel)
Zoom Capability	7X optical + 2X digital
Max. Aperture	F2.8 - F3.5
Focal length	208mm -200mm
ISO sensitivity	100/200/400/800/Auto
Movie mode (format)	Yes (QuickTime)
Timed slow shutter	<ul style="list-style-type: none"> • Timed up to 15 seconds depending on selected ISO • Bulb up to 30 seconds
Max shutter speed	<ul style="list-style-type: none"> • Program AE : 1/4000 sec • Aperture Priority: 1/4000 sec • Shutter Priority: 1/2000 sec • Manual: 1/2000 sec
White Balance	<ul style="list-style-type: none"> • Auto, Daylight, Tungsten, Fluorescent 1, Fluorescent 2, Cloudy, Manual preset (1, 2, 3)
Continuous drive	<ul style="list-style-type: none"> • UHS - 7 fps @ 1280 x 960 • High-continuous - 3 fps (9 frames) At 2560 x 1920 resolution • Continuous - 2 fps (9 frames) At 2560 x 1920 resolution
Colour modes	<ul style="list-style-type: none"> • Natural colour (sRGB) • Vivid colour (sRGB) • B&W (neutral or toned) • Solarisation • Adobe RGB
Embedded profile	Yes, optional embedded ICC profile
Video out	Yes
USB Support	Yes
Sound Recording	Yes

5.3 The Raw Images Database

During this research, images were collected for the development and verification of the algorithms used for traffic sign recognition. A total of 3415 images were collected in Sweden and 330 images in other countries. These images comprise one of the biggest databases of traffic signs in Europe [104]. All images collected in Sweden were taken from the same position in the vehicle, but different vehicles were used depending on

availability. All still images were taken manually when traffic signs were seen by the camera operator. They were collected in different light conditions, in different weather conditions and in different road conditions including different speeds. For all images and without any exception, the camera was set to 640x480 pixels; Extra Fine JPG format and Continuous drive which allows a sequence of images to be taken at different distances between the vehicle and the sign.



Figure 5.2: Minolta DiMAGE 7Hi.

Images in this database are classified into 30 categories depending on weather conditions, type of the sign, sign condition, image condition and light geometry. Table 5.2 presents these categories together with the number of images in each category. Some of the images can be classified into different categories depending on the condition of the image.

Furthermore, this database includes additional traffic sign images taken from different places around the globe. Most of them are collected in Europe but some of them are taken in Canada, the USA, Singapore and Japan. These images are taken using different cameras depending on availability. Some of them are taken by a pedestrian and they can be of different sizes and different resolutions. One of the objectives of collecting images from different countries is to study the differences and compare the colours and

pictogram shapes used by these countries. Another objective is to test the compatibility of the algorithms developed in the research.

Table 5.2: Raw images database comprises different categories.

Category	Number of Images	Category	Number of Images
Bad light geometry	77	SL-5	2
Blurred images	290	SL-10	5
Closed to all Vehicles	62	SL-15	3
Faded signs	158	SL-20	21
Fog	27	SL-30	94
Highlight	40	SL-50	196
Information Signs	508	SL-70	326
No Entry sign	125	SL-90	351
No Parking sign	92	SL-110	81
No Standing	93	Snowfall	80
Noisy images	48	Stop sign	120
Occluded signs	96	Sunny	922
Physically Damaged	61	Sunrise and Sunset	489
Prohibitory signs	227	Warning signs	443
Rain	122	Yield sign	157

A Microsoft Access application was developed to manage these images and associated information. It consists mainly of two tables: the pictures table and the signs table. In the pictures table, each record consists of the picture's name, date, weather condition under which the picture was taken, defects in the picture, light geometry, and time of day. The signs table contains all the signs gathered during the development stage of this research. As any picture may contain one or more sign depending on the nature of the sign, each record in this table is set to consist of the following fields: type of the sign, x and y coordinates of the centre of the sign, rim colour, interior colour, sign shape, sign

condition, and picture ID which represents the picture in which the sign is found. Other tables are created to prevent typing mistakes in the pictures and signs table. Such tables are called phrase tables and they contain all the phrases used in this database. A full description of this database is given in Appendix E.

5.4 Colour Segmentation Algorithms

The aim of the colour segmentation algorithm is to recognise multicolour traffic signs. Its success in doing so is judged according to the following criteria:

- Robustness to a change in the viewing direction.
- Robustness to a change in object geometry.
- Robustness to a change in the direction of illumination.
- Robustness to a change in the intensity of illumination
- Robustness to a change in the Spectral Power Distribution (SPD) of illumination.
- Robustness to changes in weather conditions.

Some of the above criteria are interrelated. The choice of a colour model not only depends on its robustness to the variation in the illumination across the scene, but also on its robustness to changes in the surface orientation of the object. Two other factors may affect the robustness of the colour model: robustness to noise and robustness to occlusions and cluttering.

The Swedish National Road Administration defined the colours used for traffic and road signs [105] in CMYK colour space. The original CMYK values are given in Table 2.1 in chapter 2. The values in this table are converted into Normalised Hue and Normalised Saturation and are listed in Table 5.3.

Table 5.3: Normalised Hue and Saturation.

Colour	Normalised Hue [0,255]	Normalised Saturation [0,255]
Red	250	207
Yellow	37	230
Green	123	255
Light Blue	157	255
Dark Blue	160	230

There now follows a discussion of the new colour segmentation algorithms developed by the author in this research. The evaluation of the performance of these algorithms will be presented in the next chapter.

5.4.1 The Dynamic Threshold Algorithm

In this algorithm colour segmentation is carried out by converting the RGB image into the IHLS colour space. Hue, Saturation and Luminance images are normalised to [0,255]. In the first step of the algorithm, the global normalised mean $Nmean$ of the luminance image is calculated. This is achieved in two steps. In the first step, the global image mean is calculated. In the next step the normalised global mean is computed by dividing the mean by 256, as shown in the following equations:

$$mean = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} L(i, j) \quad (5.1)$$

$$Nmean = mean / 256 \quad (5.2)$$

where m and n are the image dimensions, $L(i, j)$ is the luminance of the current pixel, and $Nmean$ is the normalised global mean in range [0,1].

The reference colour and the unknown colour are represented by two vectors on the hue colour circle by using hue and saturation values of these two colours, Figure 5.3.

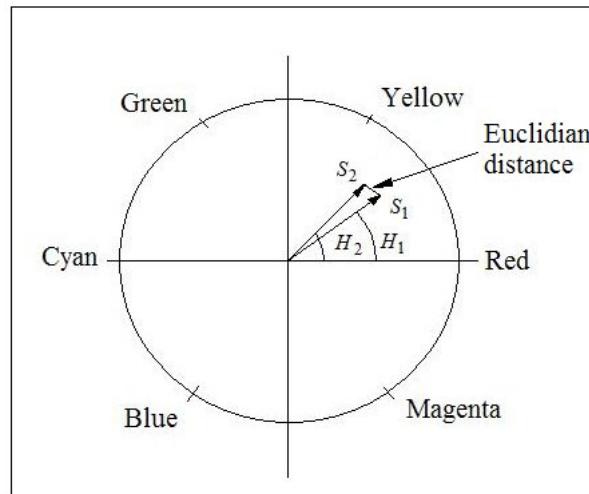


Figure 5.3: The vector model of the Hue and Saturation.

Let H_1 and H_2 be the hue angles of the reference colour and unknown colour respectively, and S_1 and S_2 be the saturation values of the reference colour and unknown colour respectively, then, the Euclidian distance between the two vectors is calculated using the following equation:

$$d = \left((S_2 \cos H_2 - S_1 \cos H_1)^2 + (S_2 \sin H_2 - S_1 \sin H_1)^2 \right)^{1/2} \quad (5.3)$$

Because the hue angle and saturation can be affected by the light conditions at the time the image is taken, a dynamic threshold is calculated in correspondence to the normalised global mean which reflects these dynamic light conditions. This threshold is calculated as:

$$thresh = e^{-Nmean} \quad (5.4)$$

The main idea here is to develop a dynamic threshold related to the brightness of the image. When the brightness of the image is high, the threshold is small, and vice versa. This allows the luminance image to control the relationship between the reference pixel and the unknown pixel. Hence, the Euclidian distance is related to the normalised global mean. When the image is bright, i.e. the value of the normalised global mean is high, a lower threshold is considered and vice versa.

The algorithm checks all the pixels in the image seeking for object pixels. The pixel is considered to be an object pixel if the Euclidian distance is less than or equal to the threshold; otherwise it is considered as background. The output of the algorithm is a binary image containing all pixels with similar colour to the reference colour. Figure 5.4 shows the results of colour segmentation using this method. This method was published by the author in 2004 [66] and its performance is presented in the next chapter.





Figure 5.4: Results of colour segmentation using dynamic threshold algorithm applied for Red, Yellow and Green signs.

5.4.2 A Modification of de la Escalera's Algorithm

This is a modified version of the algorithm described by de la Escalera et al. [16]. In the modified algorithm, the RGB image is converted into the IHLS colour space, and both saturation and hue are normalised to be [0,255]. The transfer functions of saturation, red, green, blue hues are shown in Figures 5.5 - 5.8. To avoid the achromatic hue subspaces defined by Vitabile et al. [11], the minimum and maximum values of saturation are chosen to be $S_{\min} = 51$, $S_{\max} = 170$ in the normalised scale i.e. [0-255], and saturation is then calculated as follows:

$$S_{out} = \begin{cases} 0 & 0 \leq S_{in} \leq S_{\min} \\ S_{in} & S_{\min} < S_{in} < S_{\max} \\ 255 & S_{\max} \leq S_{in} \leq 255 \end{cases} \quad (5.5)$$

Hue is calculated by:

$$H_{out} = \begin{cases} 255 & H_{min} \leq H_{in} \leq H_{max} \\ 0 & \text{otherwise} \end{cases} \quad (5.6)$$

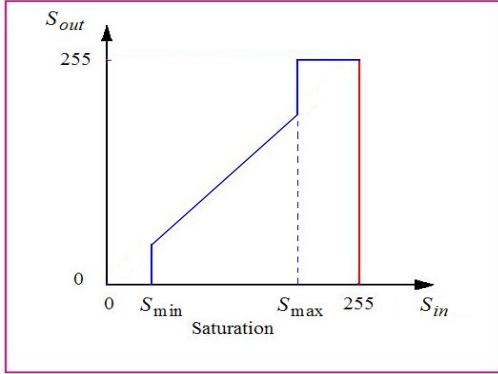


Figure 5.5: Saturation transfer function.

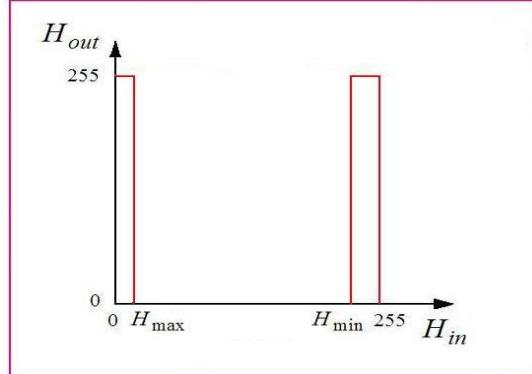


Figure 5.6: Hue transfer function of Red.

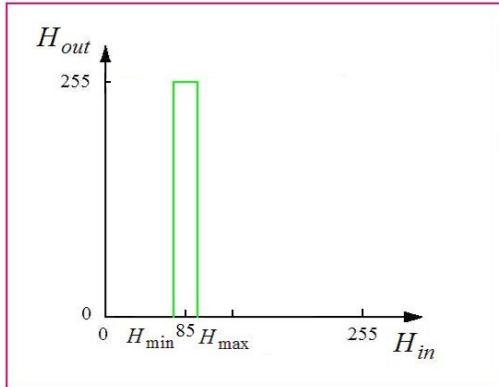


Figure 5.7: Hue transfer function of Green.

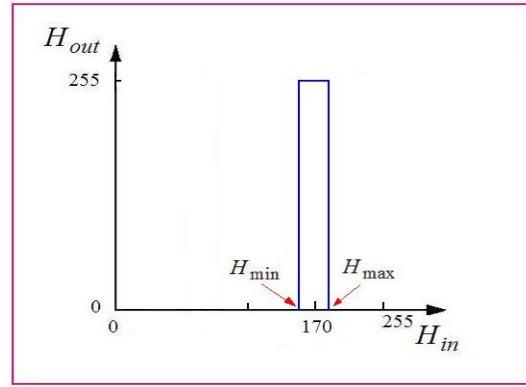


Figure 5.8: Hue transfer function of Blue.

When saturation is below a certain level, Figure 5.5, hue is meaningless and no colour information can be extracted. This is the achromatic region. In contrast, when saturation is above the upper limit, a pure hue is involved and there is no need to involve saturation in the calculations. However, saturation helps the hue in the region between the lower and upper limits. A logical AND between S_{out} and H_{out} will generate a binary image containing the road sign with the desired colour. Figure 5.9 shows segmented images produced by this algorithm. This method was published by the author in 2004 [66] and the performance tests are presented in the next chapter.



Figure 5.9: Results of applying the modified version of de la Escalera algorithm.

5.4.3 The Fuzzy Colour Segmentation Algorithm

The fuzzy colour segmentation algorithm published by the author [106] is carried out by converting RGB images into HSV colour space. The HSV colour space is chosen because Hue is invariant to variations in light conditions as it is multiplicative/scale invariant, additive/shift invariant, and it is invariant under saturation changes. In practice, this means that it is still possible to recover the tint of the object when it is lit with intensity varying illumination space.

Normalised Hue and Saturation which are mentioned in Table 5.2 are used as a priori knowledge to the fuzzy inference system to specify the range of each colour in this system. To detect and segment any of the colours used for traffic signs, seven fuzzy rules are applied. These rules are as follows:

1. If (*Hue* is **Red1**) and (*Saturation* is **Red**) then (*result* is **Red**)
2. If (*Hue* is **Red2**) and (*Saturation* is **Red**) then (*result* is **Red**)
3. If (*Hue* is **Yellow**) and (*Saturation* is **Yellow**) then (*result* is **Yellow**)
4. If (*Hue* is **Green**) and (*Saturation* is **Green**) then (*result* is **Green**)
5. If (*Hue* is **Blue**) and (*Saturation* is **Blue**) then (*result* is **Blue**)
6. If (*Hue* is **Noise1**) then (*result* is **Black**)
7. If (*Hue* is **Noise2**) then (*result* is **Black**)

The membership functions of Hue and Saturation are depicted in Figures 5.10 and 5.11, respectively. Hue is represented as an angle of a range [0-360].

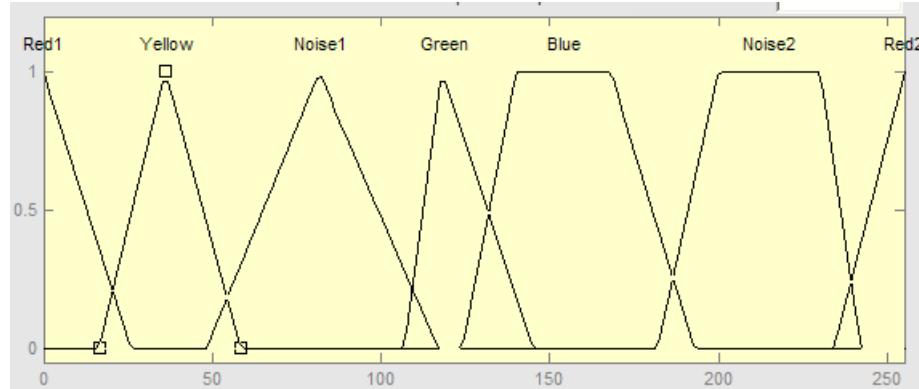


Figure 5.10: Hue membership functions.

By normalising hue values to [0-255] to fit the image processing demand, small changes are needed here. Since the range of the Hue of the red colour is around zero (above zero values or below 255), two fuzzy variables are defined for Hue. They are called **Red1** to represent colour values above zero, and **Red2** to represent colour values below or equal 255. Moreover, there are two regions of Hue values which are not used for road signs. They are defined as **Noise1** and **Noise2**. If any of these colours are faced by the fuzzy inference system, it responds by initiating a black pixel.

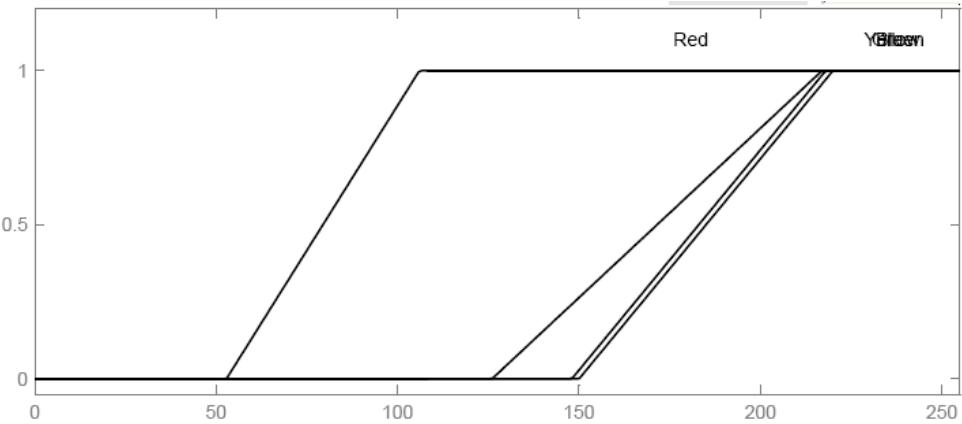


Figure 5.11: Saturation membership functions.

Membership functions of Saturation show that almost all of the values of the colours used in road signs are located on the right side and they are similar to each other with small differences in their ranges.

Figure 5.12 depicts the “*result*” output variable. There are five member functions, one for each colour. They represent a certain range of grey levels in the output image which correspond to the colours used in road signs. The fuzzy surface is shown in Figure 5.13, and it shows the relation among Hue, Saturation, and result variables. It is derived from the membership functions of Hue, Saturation and result variables.

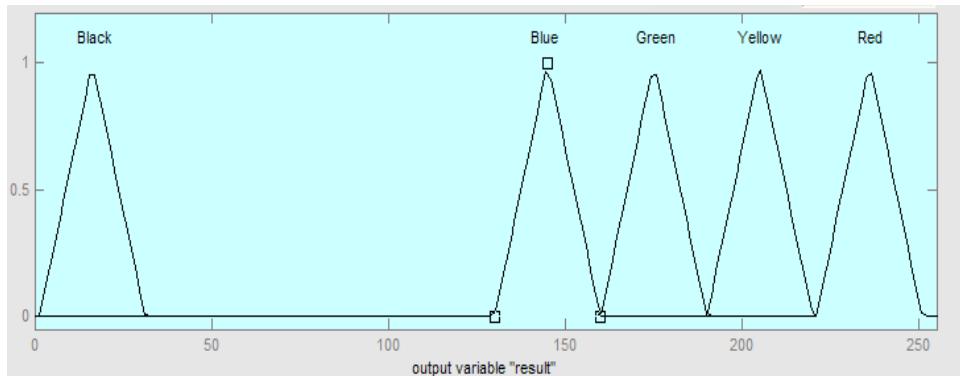


Figure 5.12: The Output functions.

Grey level slicing is used to separate different grey levels generated by the fuzzy inference system, which represent different colours. The output of this grey level slicing is a binary image containing the desired colour.

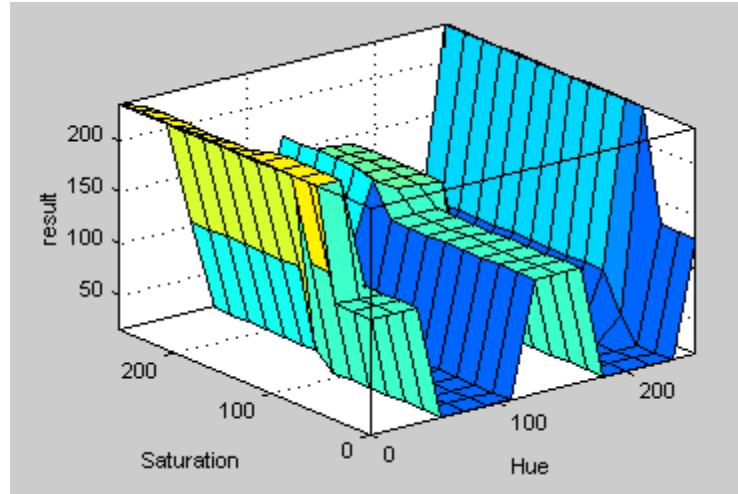


Figure 5.13: The fuzzy system surface.

The algorithm is based on invoking the HSV colour space, and uses hue and saturation to generate a binary image containing the road sign of a certain colour. The system uses a fuzzy inference to achieve colour segmentation and some of the sample images are shown in Figure 5.14.

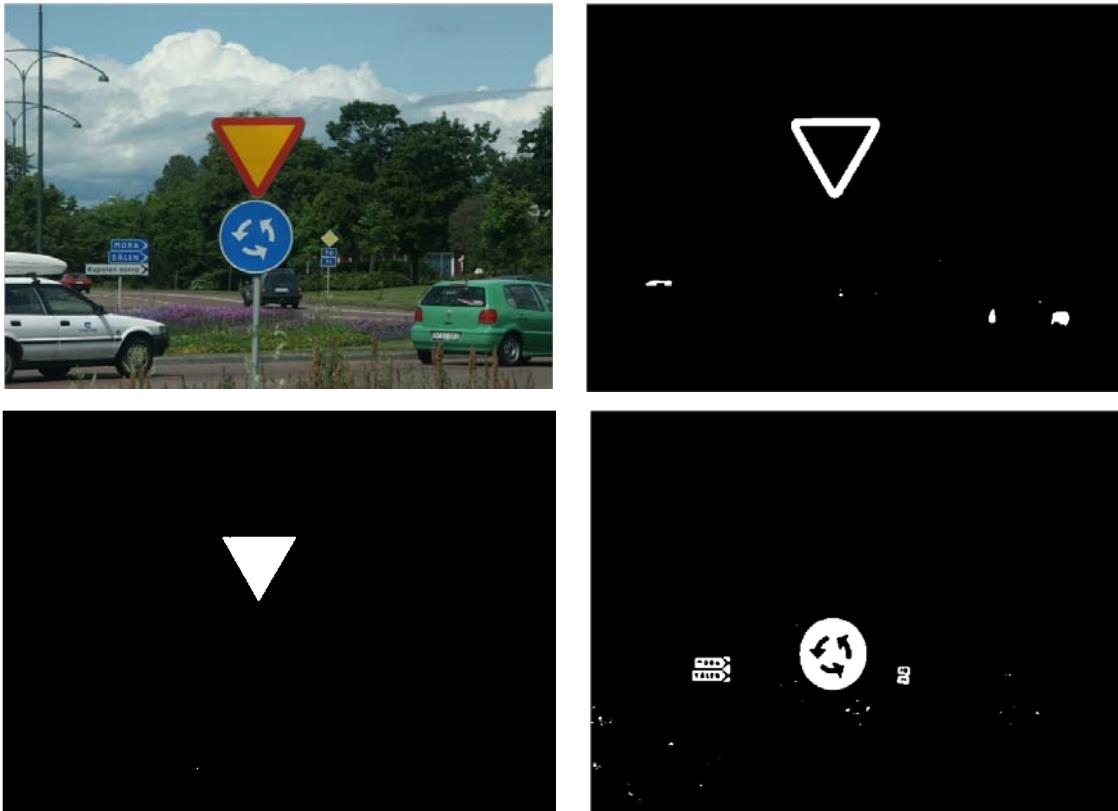




Figure 5.14: Segmentation results from fuzzy colour segmentation algorithm.

The algorithm is further tested on “Children” sign images collected from different European countries [23] and the results are shown in Figure 5.15. Further analysis of the performance of this algorithm is presented in the next chapter.

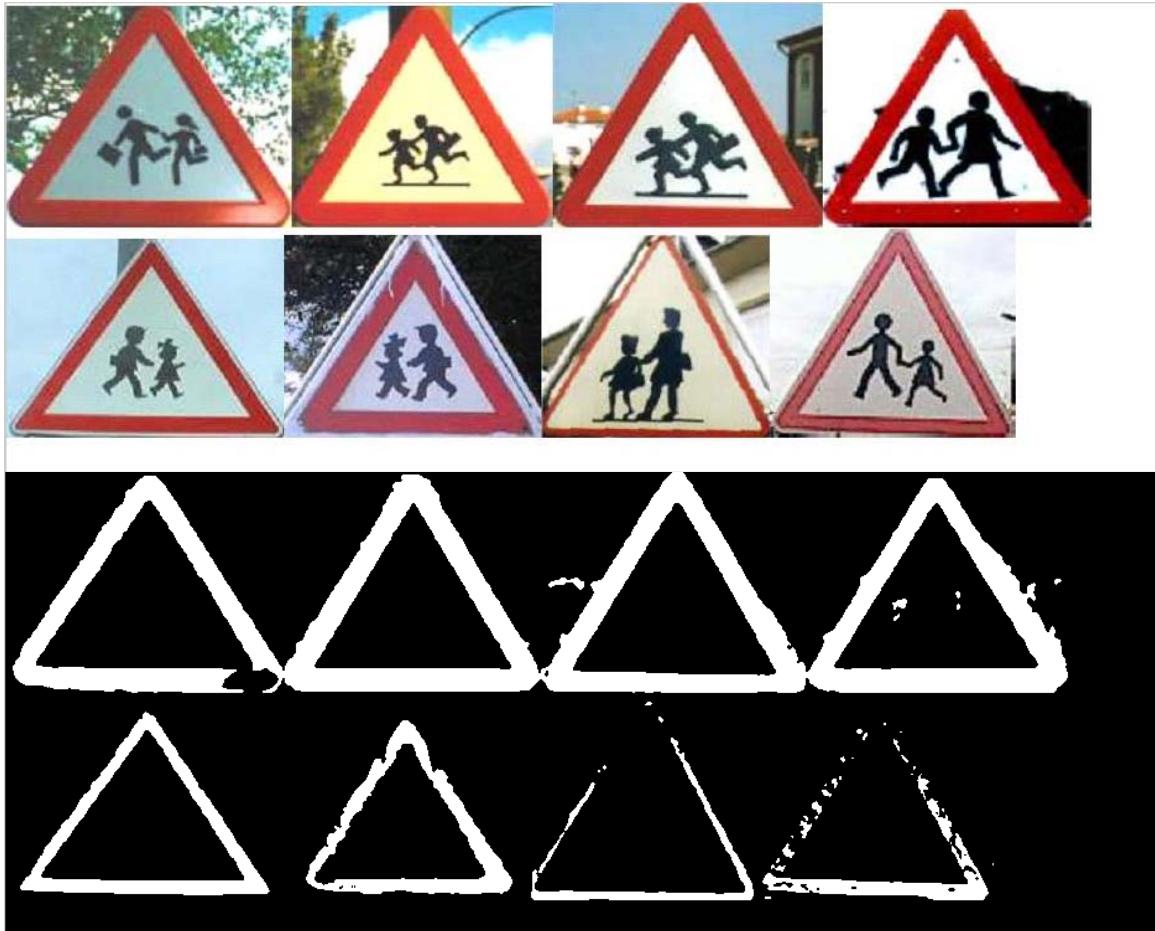


Figure 5.15: Children signs from different European countries.

5.4.4 Shadow and Highlight Invariant Algorithm

Daylight has two components, sunlight and the ambient component. The 3D direction of the traffic sign determines the amount of light received by the traffic sign from each source. Surface reflectance, based on the combined geometry of the illumination and viewing, affects the composition and amount of light reflected by the surface into the camera. As described earlier, when the angles between the vectors \mathbf{n}, \mathbf{s} and \mathbf{n}, \mathbf{v} approach each other, the traffic sign reflects more light from the source to the camera, otherwise ambient light is reflected by the sign. The model described in this section covers two major problems: the effect of shadows (different parts of the sign are exposed to different illumination levels) and the effect of highlights (light from sky or source is reflected to the viewer by the sign).

The Reflection Model

Consider a certain surface patch which is illuminated by an incident light with a certain Spectral Power Density (SPD) denoted $e(\lambda)$. The RGB sensors of the camera which is used to take an image of this surface patch are characterised by their spectral sensitivities $f_C(\lambda)$ for $C = \{R, G, B\}$. The C^{th} sensor response of the camera is given by:

$$C = m_b(\mathbf{n}, \mathbf{s}) \int_{\lambda} f_C(\lambda) e(\lambda) c_b(\lambda) d\lambda + m_s(\mathbf{n}, \mathbf{s}, \mathbf{v}) \int_{\lambda} f_C(\lambda) e(\lambda) c_s(\lambda) d\lambda \quad (5.7)$$

for $C = \{R, G, B\}$

where $c_b(\lambda)$ and $c_s(\lambda)$ are the surface albedo and Fresnel reflectance respectively, λ is the wavelength at which the sensor responds, and $\mathbf{n}, \mathbf{s}, \mathbf{v}$ are unit vectors which represent the direction of the normal vector to the surface patch, the direction of the source of illumination, and the direction of the viewer, respectively. Furthermore, the terms m_b and m_s denote the geometric dependencies on the body and surface reflection component, respectively [107].

Assuming that Fresnel albedo $c_s(\lambda)$ is constant and independent of the wavelength, and white illumination is used (white illumination means equal energy for all wavelengths within the visible spectrum), then $e(\lambda) = e$ and $c_s(\lambda) = c_s$ which are constants.

The sensors responses can be modified as:

$$C_w = em_b(\mathbf{n}, \mathbf{s})k_C + em_s(\mathbf{n}, \mathbf{s}, \mathbf{v})c_s \int_{\lambda} f_C(\lambda) d\lambda \quad \text{for } C_w = \{R_w, G_w, B_w\} \quad (5.8)$$

In this equation, C_w is the response of the RGB sensors under the assumption of white light source, and k_C is given by:

$$k_C = \int_{\lambda} f_C(\lambda) c_b d\lambda \quad (5.9)$$

where k_C is the compact formulation depending on the sensors and the surface albedo only.

If the assumption of white illumination holds, then

$$\int_{\lambda} f_R(\lambda) d\lambda = \int_{\lambda} f_G(\lambda) d\lambda = \int_{\lambda} f_B(\lambda) d\lambda = f \quad (5.10)$$

and the reflection of the surface can be given by:

$$C_w = em_b(\mathbf{n}, \mathbf{s})k_C + em_s(\mathbf{n}, \mathbf{s}, \mathbf{v})c_s f \quad (5.11)$$

Effect of Shadows on Colour Invariance

Based on the discussion in the previous section, colour invariant to shadows (surface illuminated by different levels of brightness) is tested for different colour spaces. Consider the body reflection term in equation (5.11)

$$C_b = em_b(\mathbf{n}, \mathbf{s})k_C \quad \text{for } C_b = \{R_b, G_b, B_b\} \quad (5.12)$$

where C_b is the sensor response for the RGB under the assumption of a white light source.

According to equation (5.12), the colour perceived by the sensor depends on three factors:

1. Sensor response and surface albedo represented by k_C .
2. Illumination intensity e .
3. Object geometry (surface orientation and illumination direction) $m_b(\mathbf{n}, \mathbf{s})$.

The following colour spaces were examined for variations of brightness.

RGB colour space:

The RGB colour space is easily affected by all of the factors mentioned above since

$$C_R = em_b(\mathbf{n}, \mathbf{s})k_R \quad (5.13)$$

$$C_G = em_b(\mathbf{n}, \mathbf{s})k_G \quad (5.14)$$

$$C_B = em_b(\mathbf{n}, \mathbf{s})k_B \quad (5.15)$$

Each of the RGB channels is affected by the sensor response, surface albedo, illumination intensity, surface orientation, and illumination direction.

The intensity I:

The intensity I is defined by:

$$I = \frac{R + G + B}{3} = \frac{em_b(\mathbf{n}, \mathbf{s})(k_R + k_G + k_B)}{3} \quad (5.16)$$

According to this equation, intensity I depends on illumination intensity, object geometry, and surface albedo.

The Nrgb colour space:

The Nrgb colour space is insensitive to surface orientation, illumination direction, and illumination intensity as:

$$r = \frac{em_b(\mathbf{n}, \mathbf{s})k_R}{em_b(\mathbf{n}, \mathbf{s})(k_R + k_G + k_B)} = \frac{k_R}{k_R + k_G + k_B} \quad (5.17)$$

$$g = \frac{em_b(\mathbf{n}, \mathbf{s})k_G}{em_b(\mathbf{n}, \mathbf{s})(k_R + k_G + k_B)} = \frac{k_G}{k_R + k_G + k_B} \quad (5.18)$$

$$b = \frac{em_b(\mathbf{n}, \mathbf{s})k_B}{em_b(\mathbf{n}, \mathbf{s})(k_R + k_G + k_B)} = \frac{k_B}{k_R + k_G + k_B} \quad (5.19)$$

Hue and Saturation:

Hue and saturation are invariant against variations of brightness which means that they only depend on the sensor response and surface albedo as long as white illumination condition holds.

From equations (B.29) and (B.30), the hue angle and saturation are given as:

$$\theta = \cos^{-1} \left\{ \frac{0.5 \times [(k_R - k_G) + (k_R - k_B)]}{\sqrt{(k_R - k_G)^2 + (k_G - k_B) \times (k_R - k_B)}} \right\} \quad (5.20)$$

$$S = 1 - \frac{3 \times \min(k_R, k_G, k_B)}{(k_R + k_G + k_B)} \quad (5.21)$$

Equation (5.20) and (5.21) show that hue and saturation are only affected by sensor response and surface albedo.

Effect of Highlights on Colour Invariance

Under certain conditions when the viewer is situated in a position so that the angle between the vectors \mathbf{nv} and \mathbf{ns} are approximately equal, the viewer receives two components of light. The first component is a light from the source reflected on the surface of the object, and another component which is the amount of light reflected by the object itself. The first component is called the highlight and is generated because the surface of the object acts as a mirror reflecting the light from the source to the viewer.

The contribution of the body reflection component C_b and the surface reflection component C_s is added together as shown in equation (5.11), and hence, the measured colour represents the contribution of the two components.

Testing different colour spaces according to the aforementioned argument shows that only the hue feature is invariant to the highlight component of the perceived colour. Some test results follow:

RGB colour space:

The RGB colour space is easily affected by the highlight component since

$$R_w = em_b(\mathbf{n}, \mathbf{s})k_R + em_s(\mathbf{n}, \mathbf{s}, \mathbf{v})c_s f \quad (5.22)$$

$$G_w = em_b(\mathbf{n}, \mathbf{s})k_G + em_s(\mathbf{n}, \mathbf{s}, \mathbf{v})c_s f \quad (5.23)$$

$$B_w = em_b(\mathbf{n}, \mathbf{s})k_B + em_s(\mathbf{n}, \mathbf{s}, \mathbf{v})c_s f \quad (5.24)$$

Each of the RGB channels is affected by the sensor response, surface albedo, illumination intensity, surface orientation, and illumination direction. They are also affected by the highlight component.

The intensity I:

The intensity I is defined by:

$$I = \frac{R_w + G_w + B_w}{3} = \frac{3em_s(\mathbf{n}, \mathbf{s}, \mathbf{v})c_s f + em_b(\mathbf{n}, \mathbf{s})(k_R + k_G + k_B)}{3} \quad (5.25)$$

According to this equation, intensity I depends on illumination intensity, object geometry, and surface albedo. Furthermore, it is also affected by the highlight component.

The Nrgb colour space:

The Nrgb colour space is sensitive to the highlight because it depends on the contribution of the surface reflection component as:

$$r = \frac{em_b(\mathbf{n}, \mathbf{s})k_R + em_s(\mathbf{n}, \mathbf{s}, \mathbf{v})c_s f}{3em_s(\mathbf{n}, \mathbf{s}, \mathbf{v})c_s f + em_b(\mathbf{n}, \mathbf{s})(k_R + k_G + k_B)} \quad (5.26)$$

$$g = \frac{em_b(\mathbf{n}, \mathbf{s})k_G + em_s(\mathbf{n}, \mathbf{s}, \mathbf{v})c_s f}{3em_s(\mathbf{n}, \mathbf{s}, \mathbf{v})c_s f + em_b(\mathbf{n}, \mathbf{s})(k_R + k_G + k_B)} \quad (5.27)$$

$$b = \frac{em_b(\mathbf{n}, \mathbf{s})k_B + em_s(\mathbf{n}, \mathbf{s}, \mathbf{v})c_s f}{3em_s(\mathbf{n}, \mathbf{s}, \mathbf{v})c_s f + em_b(\mathbf{n}, \mathbf{s})(k_R + k_G + k_B)} \quad (5.28)$$

Hue and Saturation:

Following the same procedure used in the aforementioned colour spaces, hue is the only invariant feature for the surface reflection component and so invariant for highlight, and hence it depends only on the sensor and the surface albedo.

$$\theta = \cos^{-1} \left\{ \frac{0.5 \times [(k_R - k_G) + (k_R - k_B)]}{\sqrt{(k_R - k_G)^2 + (k_G - k_B) \times (k_R - k_B)}} \right\} \quad (5.29)$$

In the case of saturation, it is at variance with the surface reflection component, and so cannot be used to develop any robust algorithm for highlights.

$$S = 1 - \frac{3 \times \min(em_s(\mathbf{n}, \mathbf{s}, \mathbf{v})c_s f + em_b(\mathbf{n}, \mathbf{s})k_R), (em_s(\mathbf{n}, \mathbf{s}, \mathbf{v})c_s f + em_b(\mathbf{n}, \mathbf{s})k_G), (em_s(\mathbf{n}, \mathbf{s}, \mathbf{v})c_s f + em_b(\mathbf{n}, \mathbf{s})k_B))}{3em_s(\mathbf{n}, \mathbf{s}, \mathbf{v})c_s f + em_b(\mathbf{n}, \mathbf{s})(k_R + k_G + k_B)} \quad (5.30)$$

Table 5.4 summarises the abilities of the different colour models to be invariant to different imaging conditions. In this table, the letter ‘Y’ denotes an invariant colour model to the imaging condition and ‘N’ denotes the sensitivity of the colour model to that imaging condition.

Table 5.4: The effect of imaging conditions on invariance of colours. ‘Y’ denotes invariance and ‘N’ denotes sensitivity of colour models to imaging conditions.

Colour feature	Viewing direction	Surface orientation	Highlight	Illumination direction	Illumination intensity
I	N	N	N	N	N
RGB	N	N	N	N	N
Nrgb	Y	Y	N	Y	Y
H	Y	Y	Y	Y	Y
S	Y	Y	N	Y	Y

The Algorithm

As a conclusion from the previous discussion, hue is the only component which is invariant to shadows and highlights. Therefore, this colour segmentation algorithm is carried out by converting the RGB images into HSV colour space. Hue, saturation, and value are normalised into [0,255].

While normalised Hue is used as a priori knowledge to the algorithm, normalised Saturation and Value are used to specify and avoid the achromatic subspaces in the HSV colour space described earlier. When the hue value of the colour of the pixel in the input image is within the specified colour, and its hue value is not in the achromatic area, then the corresponding value in the output image is set to white. The output image is then divided into a number of 16x16 pixel sub-images and used to calculate the seeds for the region growing algorithm. A seed is initiated if the number of white pixels in the output image is above a certain threshold level. The region growing algorithm is then applied to find all the objects in the output image which are large enough to initiate at least one seed. Noise and other small objects are rejected because of the region growing algorithm. This has the advantage that no more filtering is needed to delete these objects and the remaining objects are only the ones which potentially can be used for recognition. The flow chart of this algorithm is depicted in Figure 5.16. The colour segmentation algorithm [108] is given by the following pseudo code:

- Convert the RGB image into HSV colour space.
- Normalise the grey level of every pixel in the H image from [0,360] to [0,255].
- Normalise the grey level of every pixel in the S image from [0,1] to [0,255].
- Normalise the grey level of every pixel in the V image from [0,1] to [0,255].
- **For** all pixels in the H image
 - **If** ($H_pixel_value > 240 \text{ AND } H_pixel_value \leq 255$) **OR** ($H_pixel_value \geq 0 \text{ AND } H_pixel_value < 10$)
 - **Then** $H_pixel_value := 255$
 - **If** corresponding $S_pixel_value < 40$
 - **Then** $H_pixel_value := 0$
 - **If** corresponding ($V_pixel_value < 30$) **OR** ($V_pixel_value > 230$)
 - **Then** $H_pixel_value := 0$
- Divide the H image into 16x16 pixel sub-images.
- **For** every sub-image
 - Calculate number of white pixels
 - **If** number of white pixels ≥ 60
 - **Then** put a white pixel in the corresponding position in the seed image.
- Use seed image and H image, apply region growing algorithm to find proper regions with signs.

This method shows that hue and saturation are invariant to the effects of illumination variations and hence they can be used to develop shadow-invariant algorithms for colour segmentation. It also shows that hue is invariant to the effects of highlights and can be used to develop highlight-invariant colour segmentation algorithms. The algorithm invokes merely hue for colour segmentation, but saturation and value features of the HSV colour space are used to define the chromatic subspace in which hue can be used. Figure 5.17 depicts examples of images segmented using this method. The original images are shown in column a, and the results of segmentation using one shadow-highlight variant algorithm are shown in column b, while column c shows the results of the shadow-highlight invariant algorithm.

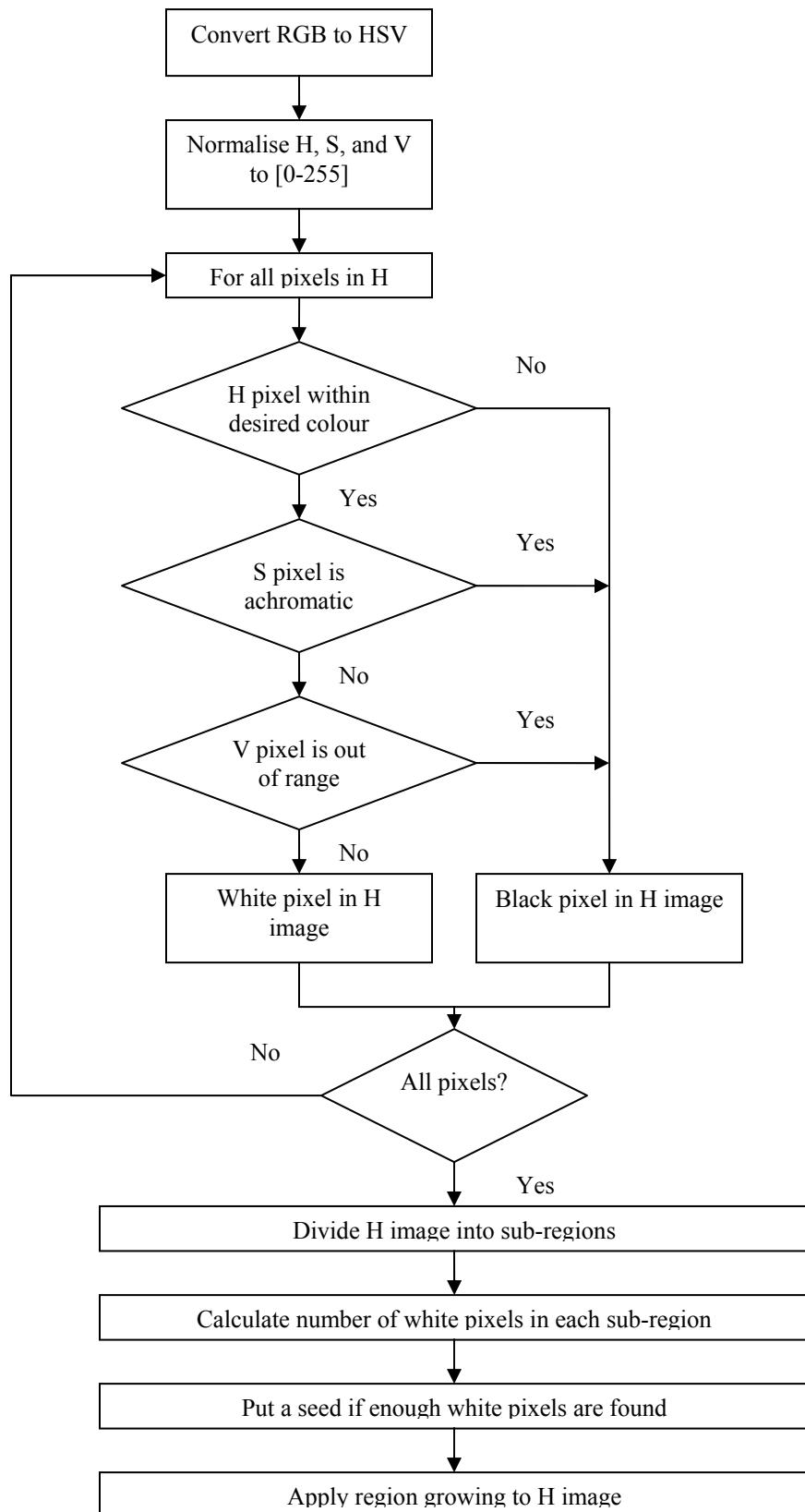
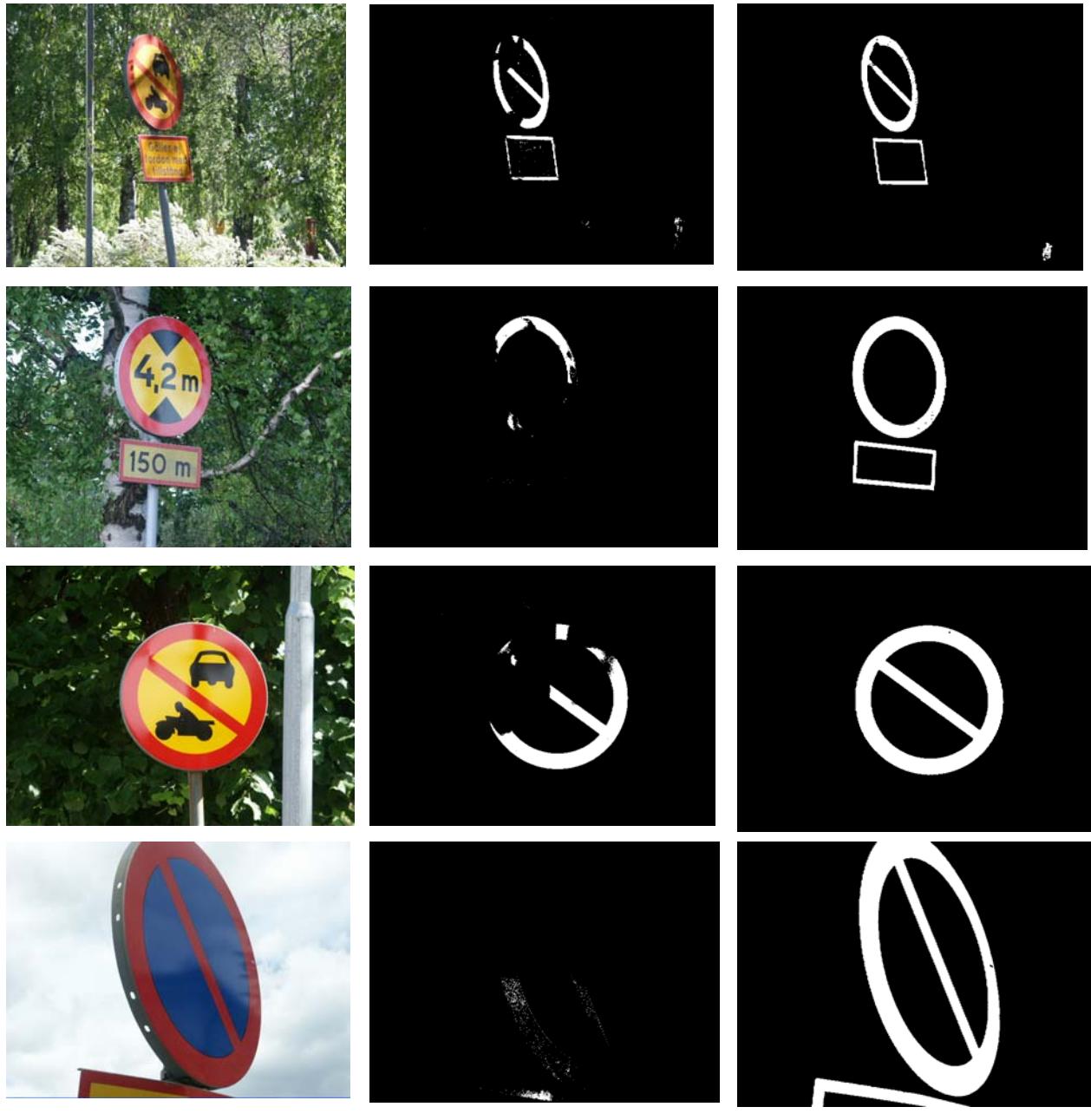


Figure 5.16: Flowchart of shadow and highlight invariant algorithm.



Original images

Results of colour segmentation using Hue-Saturation algorithm.

Results of Shadow-Highlight invariant algorithm.

Figure 5.17: Results from shadow and highlight invariant algorithm.

All noise and small objects which cannot be recognised as traffic signs are deleted by the algorithm. This reduces the chances of false alarms and improves detection. The algorithm was published by the author in 2006 [108] and its performance will be presented in the next chapter.

5.4.5 Colour Segmentation in Poor Light Conditions

In most parts of Europe, and especially in Scandinavia, winter is a long season, during which daylight hours are few, and weather conditions are generally foggy, rainy, or snowy. In other words, the entire environmental brightness is weak and consequently, the amount of incident light and reflected light from the road sign is also weak. Since colour information is very sensitive to the variations of light conditions, the need for improved algorithms to deal with such weather conditions represents a high priority in the future work of traffic sign recognition and computer vision.

In order to be able to change the levels of the RGB channels for each pixel, a colour segmentation algorithm is carried out by treating the RGB channels of the digital image separately. For this reason, the image acquired by the digital camera is separated into three different images, one in each of the RGB channels, and then histogram equalisation is applied to each channel. The resultant RGB images are then forwarded to the colour constancy algorithm to extract the true colour of the road signs. The block diagram depicted in Figure 5.18 illustrates this algorithm.

Although colour constancy is a parallel process in origin, it is carried out like a convolution process in serial computers. It is applied separately for each of the RGB channels. Let $a_i(x, y); i \in \{R, G, B\}$ be the current estimate of the local space average colour of channel i at position (x, y) in the image, $c_i(x, y)$ be the intensity of channel i at position (x, y) (this represents the input image in either of the RGB channels), and p be a small percentage of current pixel intensity greater than zero ($p = 0.0005$ is chosen). The algorithm is implemented in four steps:

1. Copy $c_i(x, y)$ to $a_i(x, y)$, and normalise both images to the range [0,1].

2. Iterate the following two operations a large number of times(10000 times):

$$a'_i(x, y) = (a_i(x - 1, y) + a_i(x + 1, y) + a_i(x, y - 1) + a_i(x, y + 1)) / 4.0$$

$$a_i(x, y) = c_i(x, y) \cdot p + a'_i(x, y) \cdot (1 - p)$$

3. Calculate the output image as

$$out_i(x, y) = c_i(x, y) / (2 \cdot a_i(x, y))$$

4. Normalise the RGB channels of the output image to the range [0,255], Ebner [71, 80].

The method is based on invoking the histogram equalisation, colour constancy, HSV colour space, and the use of hue, saturation, and value images to generate a binary image containing the road sign of a certain colour. Colour segmentation is carried out by converting the RGB image from the former step into the HSV colour space and then applying one of the segmentation algorithms discussed earlier.

The algorithm is tested on images collected in different poor light conditions. The first row of Figure 5.19 shows sample images. The second row shows images from the first row segmented by a normal colour segmentation algorithm without applying the algorithm described here. The third row shows the images after enhancement by histogram equalisation and colour constancy. It is clear that the colours of the signs are enhanced in these images. The fourth row of Figure 5.19 shows the results of the segmentation. The algorithm was published by the author in 2005 [109]. Further performance tests are shown in the next chapter.

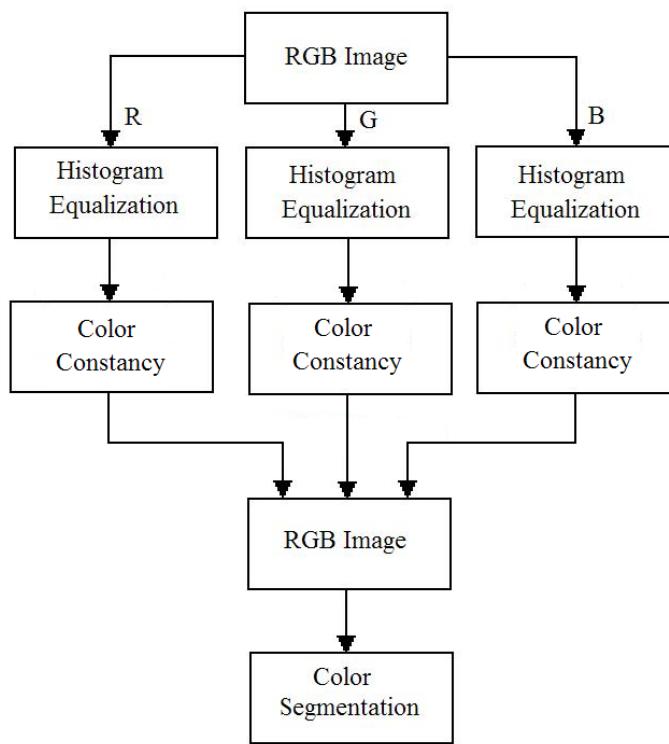
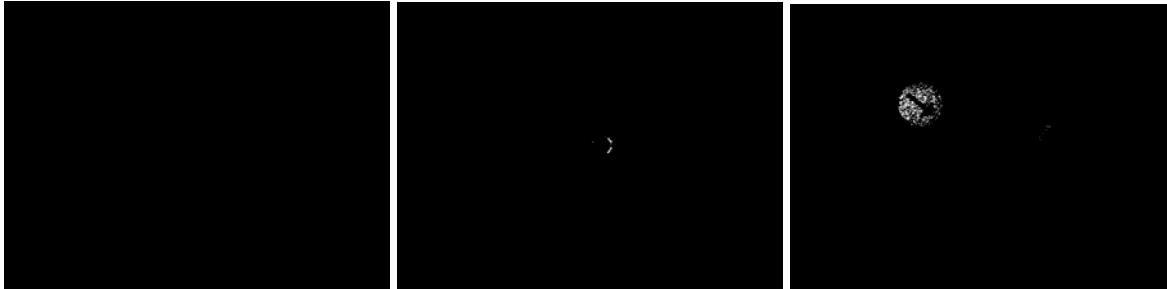


Figure 5.18: Block diagram of the colour segmentation algorithm for poor light conditions.



Original images.



Results before applying the new algorithm.



Test images after applying the histogram equalisation, and colour constancy.



The segmented images.

Figure 5.19: Results of applying colour segmentation in poor light.

5.5 Recognition by Combining Colours and Shapes

Dividing traffic signs into groups according to their colour combinations gives four alternatives, Figure 5.20:

- ◆ Red signs such as the Stop sign

- ◆ Red-Yellow signs such as Warning, Prohibitory, and Indicatory signs.
- ◆ Red-Blue signs such as No Parking and No-Stop signs.
- ◆ Blue signs such as Regulatory and Informative signs.



Figure 5.20: Colour combinations of traffic signs.

This colour grouping helps the recognition algorithm to perform in a better way and to reduce the number of false alarms which can be generated by this algorithm. Therefore, the detection and recognition of different signs requires testing the presence of different colour combinations in the image together with the presence of the specific shape. For example, if a stop sign is to be detected, a check for red colour combined with the octagon shape is searched for in the image. By contrast, if a warning sign is to be detected, a red-yellow combination together with a triangle rim would be the search criteria for the image. Figure 5.21 illustrates the details of this concept.

In the first stage colour segmentation is applied. Two rim colours exist for traffic signs in Sweden; red and blue. A traffic sign shape tree is built according to these two colours. In the blue branch, two shapes can exist; the circle and the rectangle. However, in the red colour branch, the categories shown in Figure 5.21 can be found. The seven sign rims are illustrated in the shapes box. They are the upward pointed triangle, downward pointed triangle, STOP, circle, circle with bar, circle with x, and no-entry signs. These signs can be described in three general shapes; circle, octagon and triangle. If the indicatory signs are added then another shape can be added to the list; a rectangle.

The Speed-Limit signs are on the third level and belong to the circular shape signs. Speed-Limit signs are chosen for classification because it is a priority of the traffic authorities in Sweden to keep a vehicle's speed within the indicated speed limit and because speed limit signs are very similar to each other; even the pictograms are similar to each other, and hence they represent a good challenge for research.

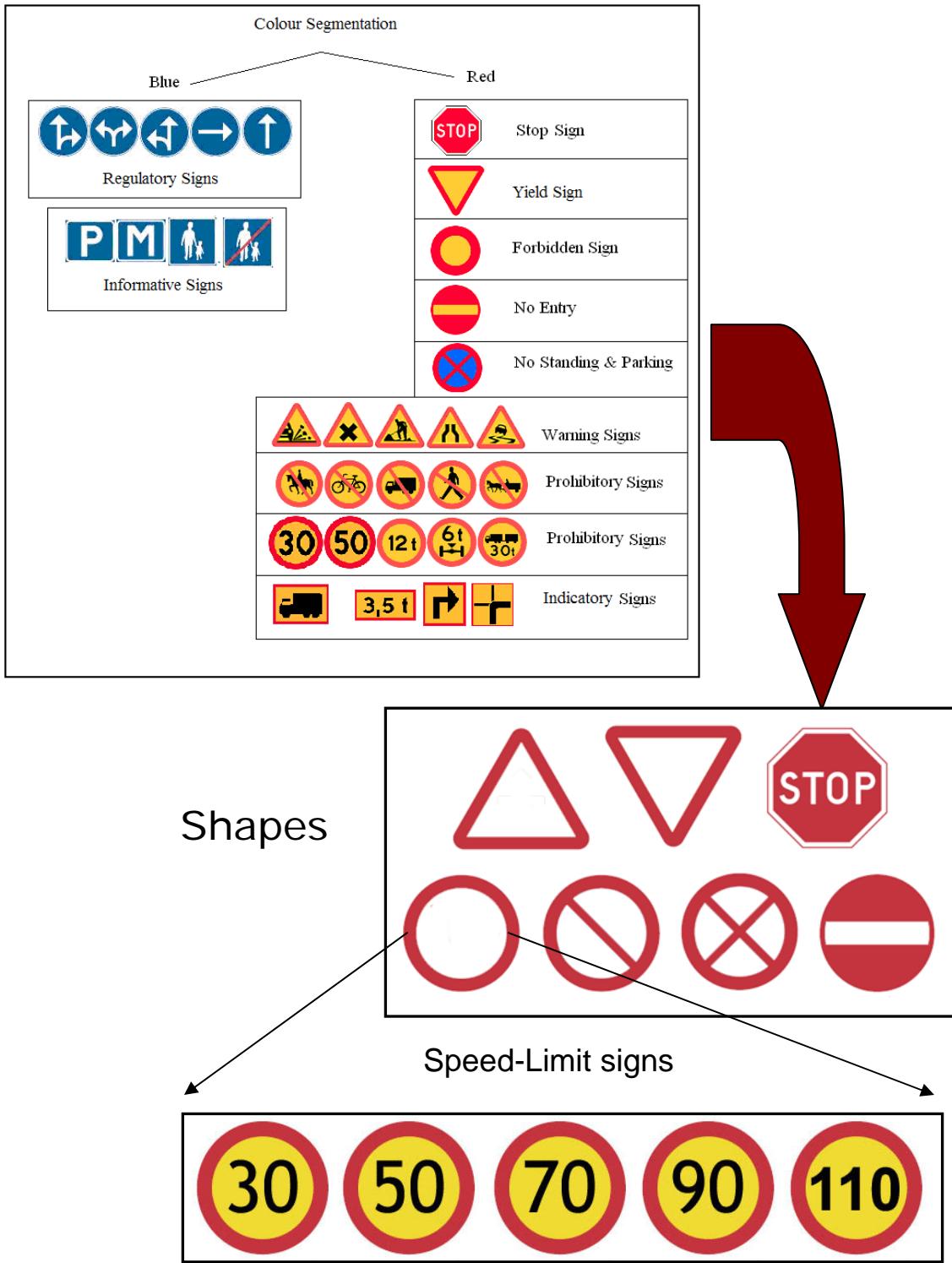


Figure 5.21: Traffic sign tree.

This model can be implemented by simple IF-THEN statements given by:

STOP:

IF (Colour is red
 AND Convex hull contains no yellow pixels
 AND Convex hull contains no blue pixels
 AND shape is octagon)
THEN sign is STOP

WARNING:

IF (Colour is red
 AND Convex hull contains yellow pixels
 AND Convex hull contains no blue pixels
 AND shape is triangle)
THEN sign is WARNING

PROHIBITORY:

IF (Colour is red
 AND (Convex hull contains yellow pixels
 OR Convex hull contains blue pixels)
 AND shape is circle)
THEN sign is PROHIBITORY

INDICATORY:

IF (Colour is red AND Convex hull contains yellow pixels
 AND shape is rectangle)
THEN sign is INDICATORY

Colour segmentation algorithms described earlier generate a segmented image in which objects that *could* be signs are represented by white pixels while all other pixels are black. However, this image may contain unwanted objects with colours similar to that of the signs. This colour-shape combination reduces the number of false alarms and

hence decreases the number of objects to be sent for classification. In this manner the whole efficiency of the system can be improved. The general shape of the sign is determined by a fuzzy system which is based on the use of what are called shape measures. These shape measures are used to determine whether the object under consideration belongs to one of the shapes that the sign could belong to i.e. circle, triangle, rectangle and octagon. Once a positive result is achieved the classification step is initiated.

5.5.1 Shape Measures

Four shape measures are used to decide the shape of the sign. They are ellipticity, triangularity, rectangularity, and octagonality.

Ellipticity can be obtained by applying an affine transform to a circle. The simplest Affine Moment Invariant I_1 [110] is given:

$$I_1 = (\mu_{20}\mu_{02} - \mu_{11}^2) / \mu_{00}^4 \quad (5.31)$$

Where μ_{20} , μ_{02} and μ_{11} are the second order central moments, and μ_{00} is the zero order central moment. To discriminate shapes more precisely higher order invariants should be involved. However, they are less reliable and very sensitive to the noise. In contrast I_1 is stable and more practical to use. To measure the ellipticity, the following equation is used:

$$E = \begin{cases} 16\pi^2 I_1 & \text{if } I_1 \leq 1/(16\pi^2) \\ 1/(16\pi^2 I_1) & \text{otherwise} \end{cases} \quad (5.32)$$

Ellipticity E ranges over $[0,1]$ and for a perfect ellipse its value is 1.

The aforementioned approach is used to characterise triangles. The triangularity measure T is given by:

$$T = \begin{cases} 108I_1 & \text{if } I_1 \leq (1/108) \\ 1/(108I_1) & \text{otherwise} \end{cases} \quad (5.33)$$

Triangularity has the same range as the ellipticity. A perfect triangle has triangularity T of 1.

Rectangularity is measured by calculating the ratio of the area of the region under consideration to the area of its minimum bounding rectangle (MBR)[111].

A new measure has been created during this research; octagonality. It gives the degree of similarity of any object with an octagon. Following [111], octagonality is given by:

$$O = \begin{cases} 15.932\pi^2 I_1 & \text{if } I_1 \leq 1/(15.932\pi^2) \\ 1/(15.932\pi^2 I_1) & \text{otherwise} \end{cases} \quad (5.34)$$

Octagonality O has the same range as ellipticity and triangularity which is $[0,1]$. A perfect octagon has an octagonality of 1.

5.5.2 Fuzzy Shape Recogniser

The four shape measures described in the previous subsection assume that the object is a solid one. This means that the object under consideration does not contain any holes. Since traffic signs have two different colours, one for the rim and the other for the interior, the rim colour is used for the segmentation and then the holes should be filled by the same grey level of the object to make it solid.

One problem which may arise here is that when the object under consideration is occluded by another object; it will be difficult to fill these holes. The reason is that the object does not form a closed shape. The first row of Figure 5.22 shows an object occluded by other objects and the second row shows the object after segmentation. This case can be treated by calculating the convex hull of the object which represents the actual solid shape of the object under consideration. A shape's convex hull is implemented using Graham's scan algorithm [112] and the results are shown in the third row of Figure 5.22. After getting the convex hull, the four shape measures are calculated and their values are forwarded to the fuzzy shape recogniser.

The fuzzy recogniser, which was published by the author as a part of this research [113], consists of five fuzzy input variables, and one output variable. The input variables are $R1$, $R2$, T , E , and O . The membership functions of these variables are shown in Figures 5.23 - 5.28.

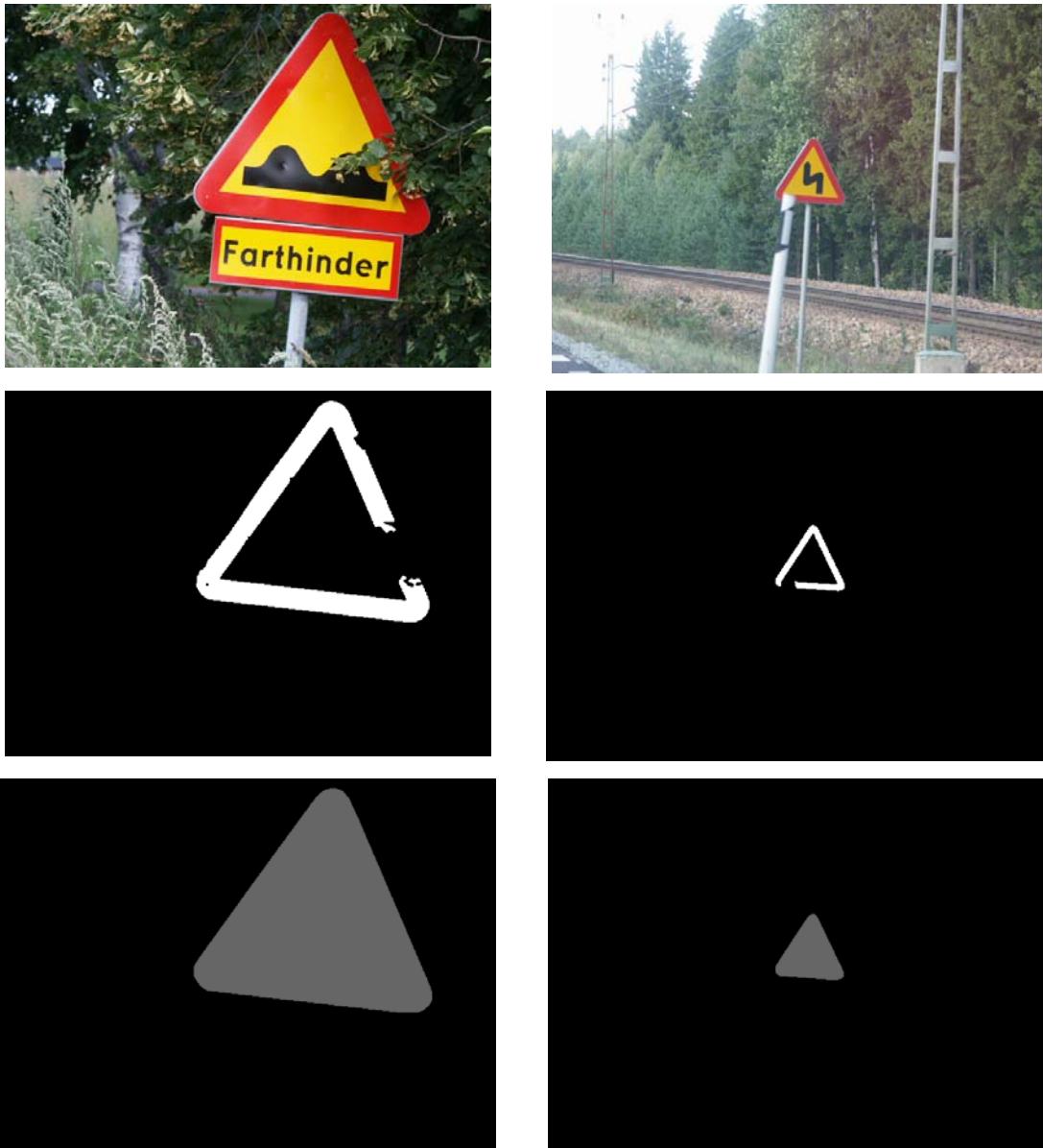


Figure 5.22: Occlusions and Convex hull. First row: Occluded objects. Second row: Results of Segmentation and Extraction. Third row: Convex hull.

To perform the recognition of traffic signs, five rules are used as follows:

1. If ($R1$ is **Low**) and ($R2$ is **Low**) and (T is **One**) and (E is **Low**) and (O is **High**) then
(*Shape* is **Triangle**)
2. If ($R1$ is **One**) or ($R2$ is **One**) then (*Shape* is **Rectangle**)
3. If ($R1$ is **Low**) and ($R2$ is **Low**) and (T is **High**) and (E is **Low**) and (O is **One**) then
(*Shape* is **Octagon**)

4. If ($R1$ is **Low**) and ($R2$ is **Low**) and (T is **High**) and (E is **One**) and (O is **Low**) then ($Shape$ is **Circle**)
5. If ($R1$ is not **One**) and ($R2$ is not **One**) and (T is not **One**) and (E is not **One**) and (O is not **One**) then ($Shape$ is **Undefined**)

Where $R1$ is rectangularity calculated for horizontally aligned objects, $R2$ is the rectangularity of objects oriented in any other angle, T is triangularity, E is the ellipticity, and O is the octagonality.

Shape measures of five different samples such as the Stop sign (octagon), the Yield sign (triangle), the No Entry sign (circle), and different rectangular signs are calculated and shown in Tables 5.5-5.8. These values are used to implement the fuzzy shape recogniser and to verify the results.

Since the shape measures are computed using the Affine Moment Invariants which are invariant to the general affine transformation, the algorithm is invariant to rotation, scaling and translation. It is also invariant to the distortion of objects by perspective projection, which takes place when the viewing angle between the camera and the sign is not zero, and there is no need to normalise the detected signs. Evaluation of the performance of this recogniser is presented in the next chapter.

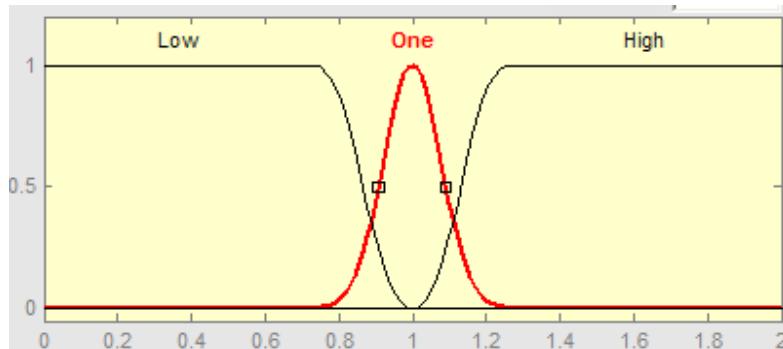


Figure 5.23: The R1 Membership Functions.

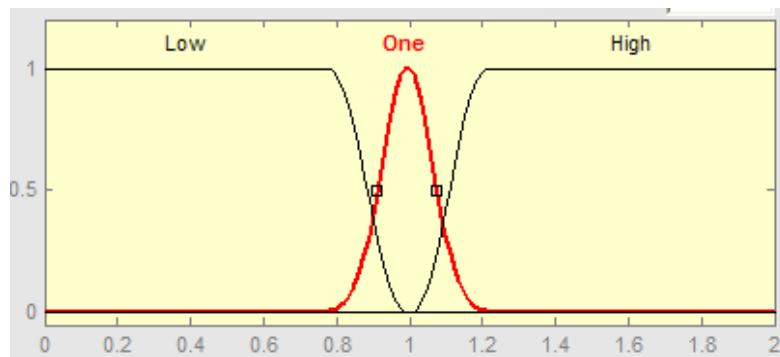


Figure 5.24: The R2 Membership Functions.

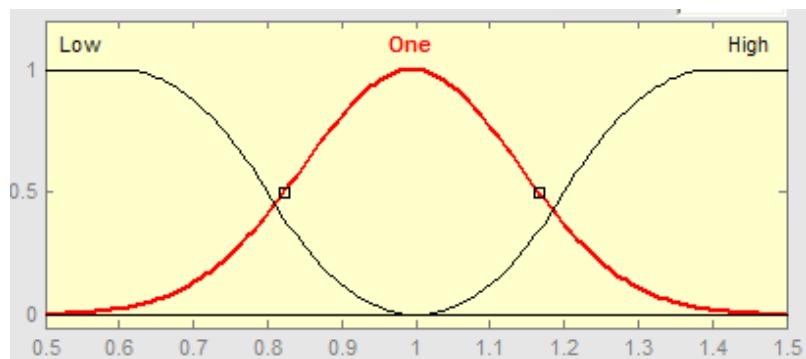


Figure 5.25: The T Membership Functions.

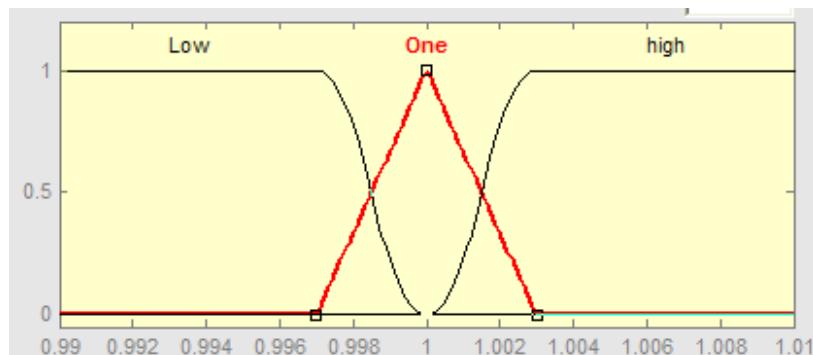


Figure 5.26: The E Membership Functions.

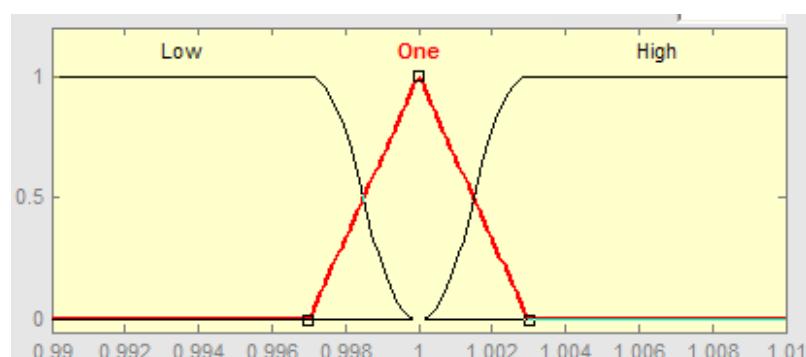


Figure 5.27: The O Membership Functions.

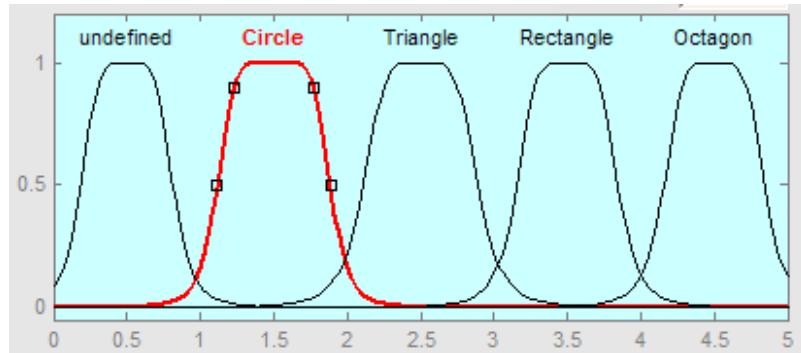


Figure 5.28: Output Membership Functions.

Table 5.5: Shape Measures of Stop Sign.

R1	R2	T	E	O
0.802083	0.609371	1.4553	0.995308	1.0004
0.819853	0.724007	1.4579	0.997082	0.9986
0.827206	0.827206	1.4557	0.995595	1.0002
0.820683	0.758180	1.4559	0.995694	1.0001
0.822917	0.559804	1.4568	0.996312	0.9994

Table 5.6: Shape Measures of Yield Sign.

R1	R2	T	E	O
0.627880	0.593256	1.1195	0.765624	1.3006
0.633641	0.588887	1.1381	0.778361	1.2793
0.629464	0.584195	1.1077	0.757563	1.3144
0.597701	0.595498	1.1424	0.781334	1.2744
0.671053	0.664599	1.2104	0.827840	1.2028

Table 5.7: Shape Measures of No-Entry Sign

R1	R2	T	E	O
0.802521	0.802521	1.4605	0.998834	0.9969
0.789719	0.772936	1.4620	0.999885	0.9959
0.776591	0.776618	1.4618	0.999765	0.9960
0.790977	0.760449	1.4617	0.999701	0.9960
0.782466	0.775507	1.4621	0.999936	0.9958

Table 5.8: Shape Measures of Rectangular Signs.

R1	R2	T	E	O
0.916325	0.938060	1.3318	0.910852	1.0932
0.923166	0.959186	1.3350	0.913026	1.0906
0.944465	0.535576	1.3368	0.914245	1.0891
0.956602	0.839174	1.3391	0.915847	1.0872
0.926373	0.802776	1.3378	0.914982	1.0883

5.6 The Algorithm

The algorithm starts by converting the RGB image into the HSV colour space. A shadow-highlight invariant colour segmentation algorithm is invoked to segment this image according to the desired colour. This colour segmentation algorithm is applied twice; for the red colour which corresponds to the rim of the sign and for the yellow which represents the interior of the sign.

The algorithm checks every red segmented image for any objects. If any objects are found, it calculates the number of these objects by applying the connected components labelling algorithm. The colour segmentation algorithm is designed to reject noise and all small objects which cannot be recognised by using the region growing algorithm, and hence no median filter is needed.

Using the red segmented image, a list of objects is then generated and objects are sent sequentially to the next stage. At this stage, the convex hull of the object under consideration is computed using Graham's algorithm. Using the convex hull of the red segmented image, the corresponding area of the yellow segmented image is checked for the presence of pixels. If yellow pixels are found with a reasonable ratio, this area is considered as a probable sign and sent to the next stage, otherwise the object is rejected and the next object in the list is treated in exactly the same manner. Once an object with these specifications is found, the fuzzy shape recogniser starts to put the object into one of the four categories described earlier. Once the object fits in any category, a rectangle is drawn around it; otherwise the object is removed from the list of objects. Figure 5.29 shows a number of images recognised using this algorithm. It is tested under bad light conditions, blurred images, faded signs, under highlight condition, occlusion, rain and snow, and it shows very good robustness. Further details of these tests are presented in the next chapter.





Figure 5.29: Examples of sign recognition

5.7 Training Database of the SVM Classifier

Warning and Prohibitory signs are characterised by a red rim and a pictogram which has a different colour. For Swedish traffic signs, the pictograms have a yellow background with black shape or text. Figure 5.30 illustrates this model of a traffic sign.

Classification is carried out in two stages; in the first stage the shape of the sign's rim is classified and if the desired shape is found then the next stage is carried out by initiating the pictogram classification. This concept is illustrated in Figure 5.31. For this reason, two databases are created: one for the sign's rims and the other for the pictograms.



Figure 5.30: A traffic sign is a rim and a pictogram.

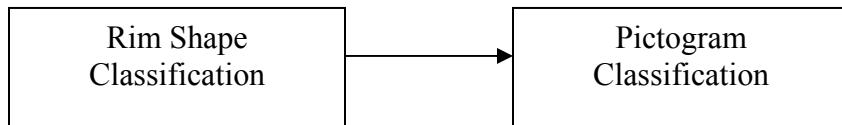


Figure 5.31: Classification is carried out by rim classifier and pictogram classifier.

Every image in this database is 36x36 pixels and is invariant to in-plane transformations, which include scale, translation and rotation. The binary image output of

colour segmentation is normalised to 36x36 pixels. This is done to standardise the size of the ROI (Region of Interest) irrespective of its scale in the original RGB image. This will also improve the reliability of the shape and pictogram classifier at a later stage. Normalised images are created as follows:

1. Apply the connected component labelling algorithm [114] to label each object in the binary image and select the desired sign for the database, then compute the convex hull of the object under consideration using Graham's algorithm.
2. To carry out the translation invariance, the object's area a and centroid (x_{cen}, y_{cen}) are computed from the following equations:

$$a = \sum_x \sum_y f(x, y) \quad (5.35)$$

$$x_{cen} = \frac{1}{a} \sum_x x f(x, y) \quad (5.36)$$

$$y_{cen} = \frac{1}{a} \sum_y y f(x, y) \quad (5.37)$$

3. Find the radius of the minimum circle containing the object by calculating the furthest object's pixel from the centroid (x_{cen}, y_{cen}) , denoted r_{min} , using Euclidean distance.

$$r_{min} = \sqrt{(x - x_{cen})^2 + (y - y_{cen})^2} \quad (5.38)$$

4. Use r_{min} to calculate the coordinates of the four corners x_{min} , x_{max} , y_{min} and y_{max} of the rectangle containing the object.
5. Calculate the new coordinates of all pixels inside the convex hull of the object to the normalised size; say NxN pixels by using the following formulas:

$$x' = N \frac{x - x_{min}}{x_{max} - x_{min}} \quad (5.39)$$

$$y' = N \frac{y - y_{min}}{y_{max} - y_{min}} \quad (5.40)$$

where (x', y') are the coordinates of a generic point in the new 36x36 matrix corresponding to the (x, y) coordinates of the pixel of the original matrix.

A convex hull is invoked in order to preserve all the details of the object under consideration. Figure 5.32 depicts the importance of using convex hull in the creation of normalised images. The data set which is built using this method contains 350 binary images classified in seven categories of traffic sign shapes (Figure 5.35) and 250 binary images classified in five categories of Speed-Limit signs (Figure 5.36). Each road sign category has 50 data samples.

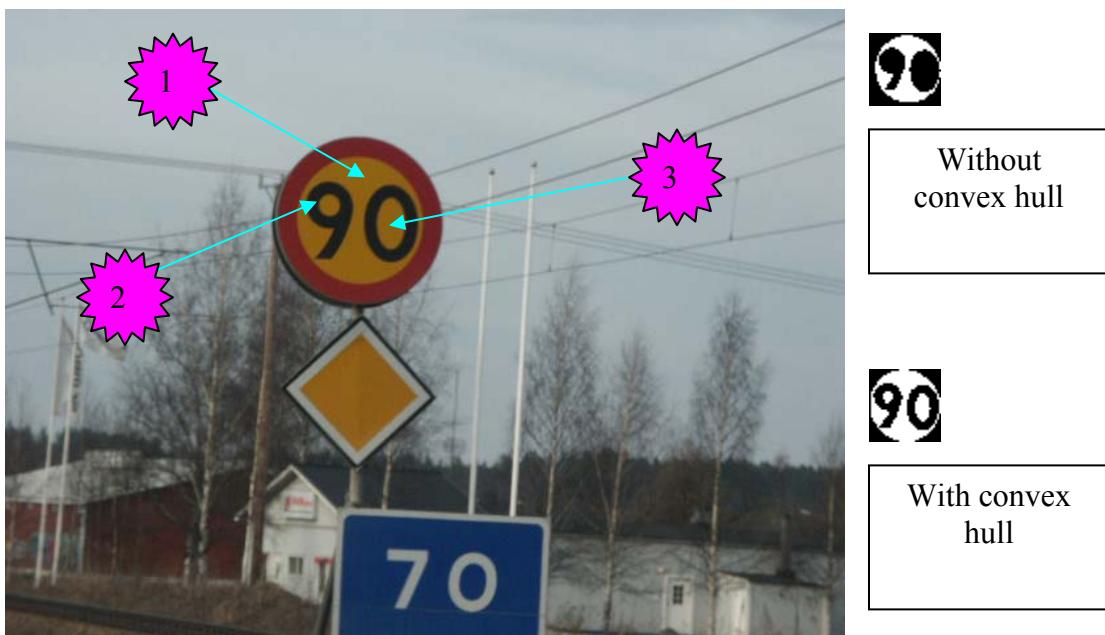
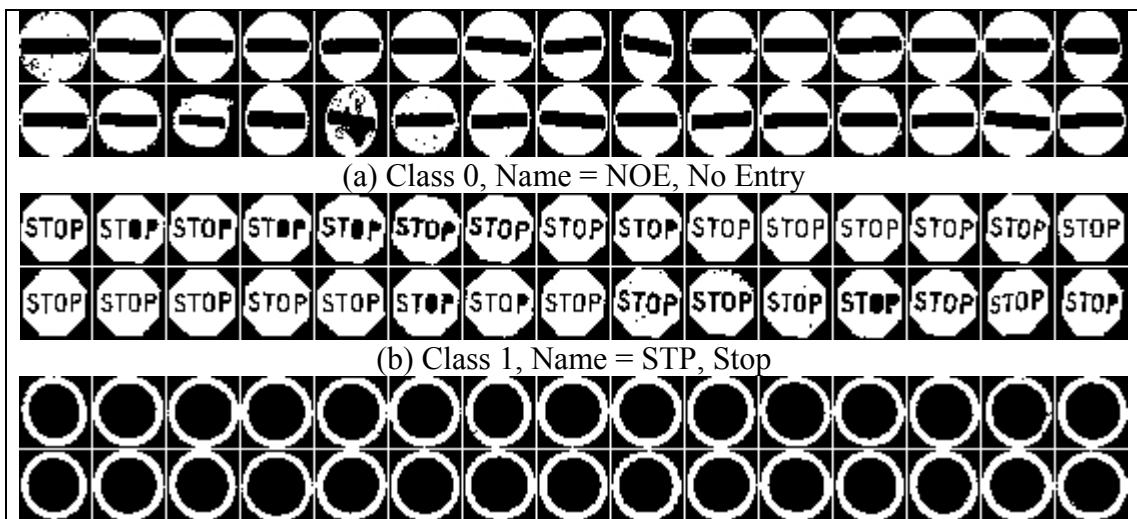


Figure 5.32: Convex hull is used to preserve objects details.



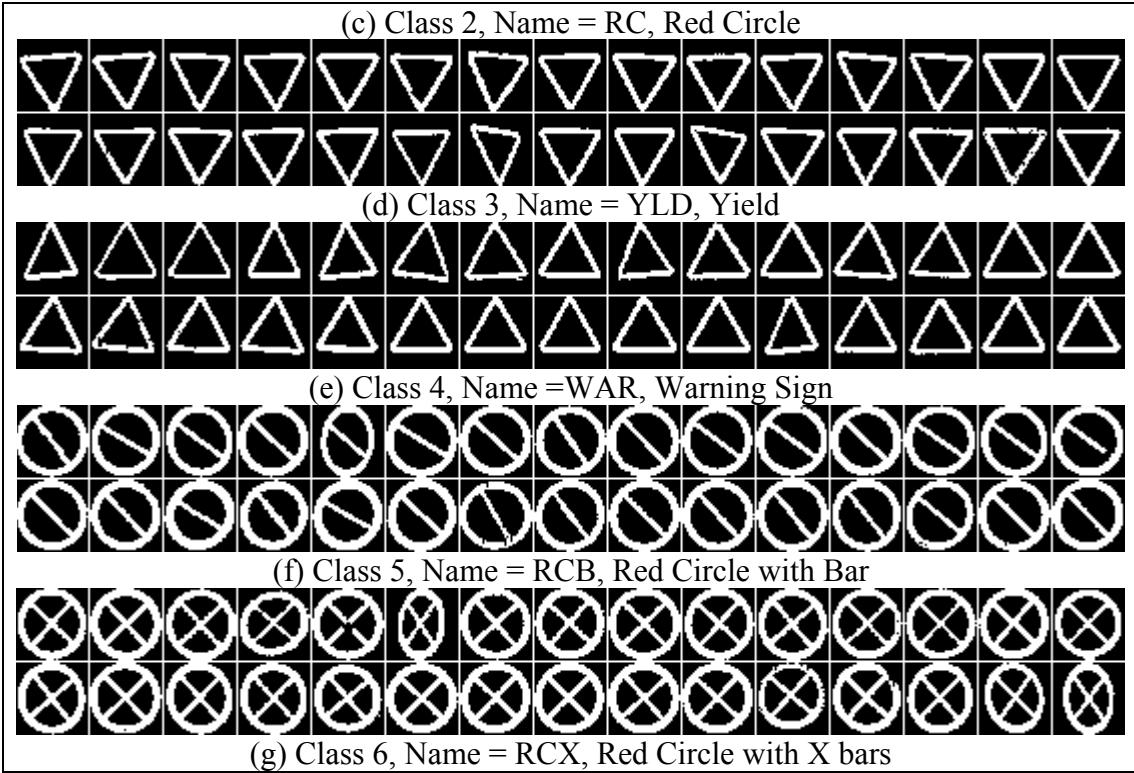
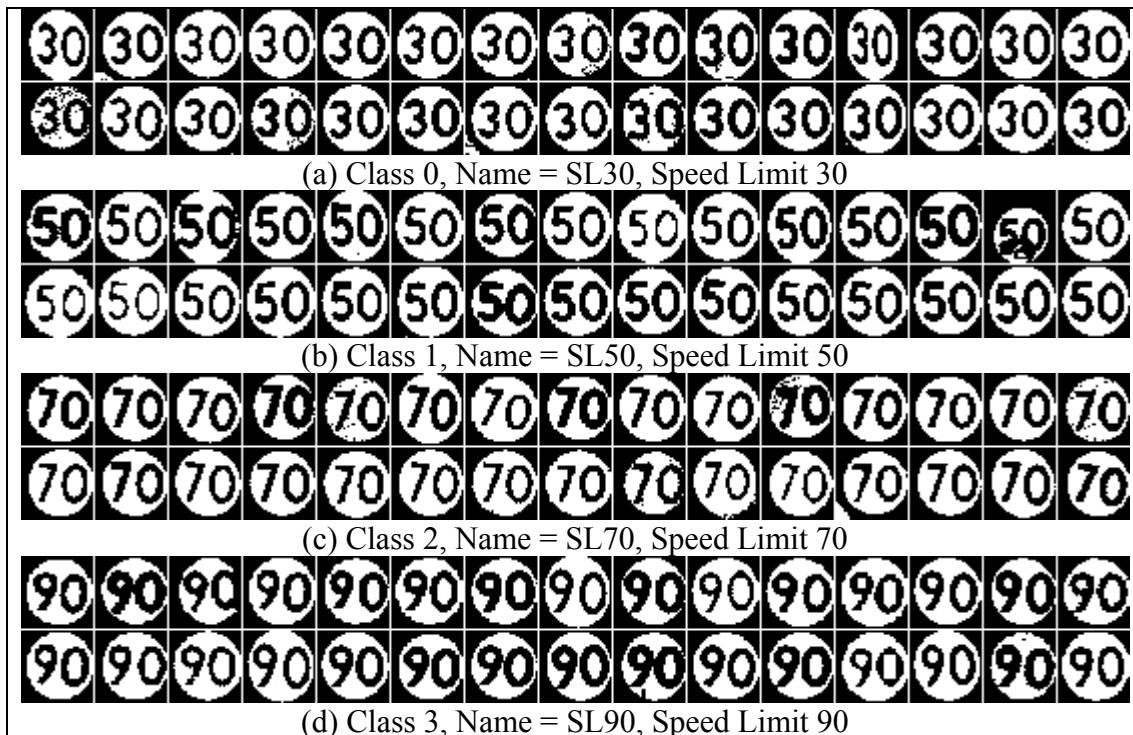


Figure 5.33: Part of the training database and their corresponding categories of traffic sign rims.



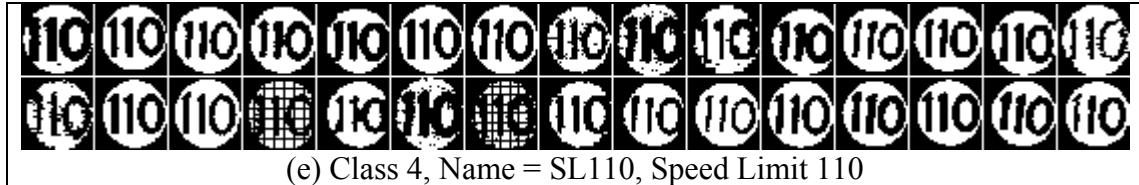


Figure 5.34: Part of the training database and their corresponding categories of Speed-Limit signs.

5.8 Classification with SVM

Images from the training database were invoked either directly or by calculating the following features; Geometric moments, Zernike moments, Legendre moments, Orthogonal Fourier-Mellin moments and Binary Haar Features. Ten pairs of training/test datasets were created either from the binary images or from these features. Each pair of datasets was selected randomly without repetition from the database for binary images or from feature files for features. There were 35 training instances and 15 test instances for every category of traffic sign rims or Speed-Limit signs. These ten pairs of datasets were invoked, separately, for training and testing of the SVM model which were carried out in this experiment and those which follow. The conditions under which this experiment was carried out are given:

- ◆ Kernel Type: Linear kernel
- ◆ SVM Type: C-SVM
- ◆ $C = 1$

The classification results of these ten experiments are listed in Appendix D. In the following subsection, the classification details are presented.

5.8.1 Classification using Binary Images

Binary images comprising seven categories of traffic sign rims and five categories of Speed-Limit signs were used for training and testing of the SVM classifier. The SVM classifier was trained and tested using the aforementioned conditions. Each category and its desired output are shown in Figures 5.34 and 5.35 respectively.

Classification of Traffic Sign Rims

For each road sign rim there are 35 training instances and 15 test instances. In this way every dataset pair contains 245 instances in the training set and 105 instances in the

test set. Tables 5.9 and 5.10 show the confusion matrices of the training and testing sets. From the confusion matrices, it is clear that using images to train the SVM gives very good results. The classifier can classify all the test images. There are two reasons for this: first, the categories are not similar to each other and second the pre-processing step has achieved good training sets. However, the main drawback of using binary images is the high number of attributes which causes the SVM to take a longer time in the training phase.

Table 5.9: Confusion matrix of training set of Rims with binary images.

Desired Output	Classified As							Total
	NOE	STP	RC	YLD	WAR	RCB	RCX	
NOE	35							35
STP		35						35
RC			35					35
YLD				35				35
WAR					35			35
RCB						35		35
RCX							35	35
Total	35	35	35	35	35	35	35	

Table 5.10: Confusion matrix of test set of Rims with binary images.

Desired Output	Classified As							Total
	NOE	STP	RC	YLD	WAR	RCB	RCX	
NOE	15							15
STP		15						15
RC			15					15
YLD				15				15
WAR					15			15
RCB						15		15
RCX							15	15
Total	15	15	15	15	15	15	15	

Classification of Speed Limit Sign

In the same manner ten pairs of training/test datasets from the database of speed limit sign images are created. Each dataset pair contains 175 instances in the training set and 75 instances in the test set. The SVM is trained and tested with these dataset pairs. The worst pair of results that are performed is 100% accuracy for training set and 96%

for the test set. The confusion matrices of the worst result are shown in Tables 5.11 and 5.12, respectively.

Table 5.11: Confusion matrix of training set of Speed-Limit with binary images.

Desired Output	Classified As					Total
	SL30	SL50	SL70	SL90	SL110	
SL30	35					35
SL50		35				35
SL70			35			35
SL90				35		35
SL110					35	35
Total	35	35	35	35	35	

Table 5.12: Confusion matrix of testing set of Speed-Limit with binary images.

Desired Output	Classified As					Total
	SL30	SL50	SL70	SL90	SL110	
SL30	15					15
SL50	1	14				15
SL70	1		14			15
SL90	1			14		15
SL110					15	15
Total	18	14	14	14	15	

Due to the poor quality of some images, they are incorrectly classified. Table 5.13 shows the images which are misclassified by this method. From the confusion matrix depicted in Table 5.12, all of the three misclassified instances are classified as SL30. The first image is damaged badly, while the second and third images are distorted and the right hand side of the sign is missing.

Table 5.13: Speed-Limit images which are incorrectly classified.

Road Sign Images	Classified Incorrectly As
	Speed Limit Sign 30
	Speed Limit Sign 30
	Speed Limit Sign 30

5.8.2 Classification with Moments and Features

Some moments such as Zernike and Legendre moments have the property of rotation invariance, which means that the values of these moments are invariant to any rotation in the image. This is a useful property as far as traffic signs are concerned since there is no guarantee that all the signs are located perpendicularly to the ground level. However, it is problematic when upward and downward pointed triangles are involved since these moments cannot distinguish between these two types of signs. Therefore, there are only six categories for traffic sign rims presented by moments. Table 5.14 shows the new category set after merging the two triangle categories into one.

Table 5.14: Traffic signs rims and their categories.

Binary Image						
Name	NOE	STP	RC	TRI	RCB	RCX
Desired Output	0	1	2	3	4	5

Different moments and features are calculated for the traffic sign rims and Speed-Limit signs. Each moment set is divided randomly and without repetition into a training set and a test set. Ten pairs of training/test datasets are created. A total of 210 instances are included in the training set per experiment compared with 90 instances for the testing set per experiment.

The SVM recognition model is trained with different moments and features using the ten training/testing sets. Confusion matrices of one experiment for each moment per sign rim type are presented in Tables 5.15-5.19, and per Speed-Limit signs in Tables 5.20-5.24.

From the confusion matrices of the rim classification, it is noticed that the major source of error is the two categories, RCB and RCX, which are both circles. The degree of similarity between the two categories makes the distinction between them using moments and features difficult. The other source of error is the similarity between the NOE sign and the RCB sign. If the RCB is inverted in colour and rotated by a certain angle, it will be very similar to the NOE sign. Since some of the moments are rotation invariant, it is a good reason to understand why this kind of misclassification occurs. On

the other hand, using moments to classify sign rims gives good results compared with the binary images.

For the Speed-Limit signs and under the specific conditions of the experiment, it is clear that Legendre moments are the best among the others. Binary Haar features come next, then Zernike moments, Geometric moments and finally OFMM. This judgement is restricted merely to this experiment. The figures shown in the confusion matrices can definitely be improved when a new set of conditions are applied. This will be demonstrated in the following chapter. The results of the Legendre moments confusion matrix shown in Table 5.22 demonstrate good consistencies with results of binary images which are shown in Table 5.12. The two tables are similar to each other and the classifier fails to recognise the same categories. Binary Haar Features show a high classification rate but the errors in the classification show a different direction. With the exception of the results of Zernike moments which are fair, the results of the OFMM and Geometric moments are very scattered.

Table 5.15: Confusion matrix of test set of Rims with Geometric moments.

Desired Output	Classified As						Total
	NOE	STP	RC	TRI	RCB	RCX	
NOE	15						15
STP	1	14					15
RC			15				15
TRI				15			15
RCB					12	3	15
RCX			1		2	12	15
Total	16	14	16	15	14	15	

Table 5.16: Confusion matrix of test set of Rims with Zernike moments.

Desired Output	Classified As						Total
	NOE	STP	RC	TRI	RCB	RCX	
NOE	15						15
STP		15					15
RC			14		1		15
TRI				15			15
RCB	1				14		15
RCX						15	15
Total	16	15	14	15	15	15	

Table 5.17: Confusion matrix of test set of Rims with Legendre moments.

Desired Output	Classified As						Total
	NOE	STP	RC	TRI	RCB	RCX	
NOE	15						15
STP		15					15
RC			15				15
TRI				15			15
RCB			1		14		15
RCX						15	15
Total	15	15	16	15	14	15	

Table 5.18: Confusion matrix of test set of Rims with OFMM moments.

Desired Output	Classified As						Total
	NOE	STP	RC	TRI	RCB	RCX	
NOE	14				1		15
STP		15					15
RC			14		1		15
TRI				15			15
RCB		1	1		13		15
RCX			1		2	12	15
Total	14	16	16	15	17	12	

Table 5.19: Confusion matrix of test set of Rims with Haar features.

Desired Output	Classified As						Total
	NOE	STP	RC	TRI	RCB	RCX	
NOE	15						15
STP		15					15
RC			15				15
TRI				15			15
RCB			1		14		15
RCX			1			14	15
Total	15	15	17	15	14	14	

Table 5.20: Confusion matrix of test set of Speed-Limit with Geometric moments.

Desired Output	Classified As					Total
	SL30	SL50	SL70	SL90	SL110	
SL30	11	3			1	15
SL50		13		2		15
SL70	2		12	1		15
SL90	1	7		7		15
SL110	3	5	1	2	4	15
Total	17	28	13	12	5	

Table 5.21: Confusion matrix of test set of Speed-Limit with Zernike moments.

Desired Output	Classified As					Total
	SL30	SL50	SL70	SL90	SL110	
SL30	11		3	1		15
SL50	1	13	1			15
SL70	1		13	1		15
SL90	1	1		13		15
SL110				1	14	15
Total	14	14	16	16	14	

Table 5.22: Confusion matrix of test set of Speed-Limit with Legendre moments.

Desired Output	Classified As					Total
	SL30	SL50	SL70	SL90	SL110	
SL30	15					15
SL50		15				15
SL70	1		14			15
SL90				15		15
SL110	1				14	15
Total	17	15	14	15	14	

Table 5.23: Confusion matrix of test set of Speed-Limit with OFMM moments.

Desired Output	Classified As					Total
	SL30	SL50	SL70	SL90	SL110	
SL30	2		11	2		15
SL50		1	13	1		15
SL70			15			15
SL90			5	9	1	15
SL110			4		11	15
Total	2	1	48	12	12	

Table 5.24: Confusion matrix of test set of Speed-Limit with Haar Features.

Desired Output	Classified As					Total
	SL30	SL50	SL70	SL90	SL110	
SL30	14		1			15
SL50	1	13			1	15
SL70			15			15
SL90				15		15
SL110					15	15
Total	15	13	15	15	16	

5.9 Summary

In this chapter, the building block of the proposed system of traffic sign recognition has been presented. To develop a traffic sign recognition system, a database of images which were collected using a camera was built to test and validate the developed algorithms. This database is managed using an Access database application.

The problem of traffic recognition was divided into three steps: colour segmentation, recognition and classification. In the first step, four colour segmentation algorithms which show a wide range of methods and ideas that can be used for colour segmentation were presented. The result of colour segmentation was a binary image containing the road sign as well as a number of other objects with similar colours.

In the second step, a description of the sign recognition algorithm using colour-shape combination was discussed. Recognition is achieved by a fuzzy shape recogniser. It uses a number of fuzzy rules derived from a set of shape measures which are immune to in-plane transformations.

The last step of traffic sign recognition was classification. The Support Vector Machines classifier was used in this stage. This classifier was trained and tested by either a set of normalised images or a set of features which involve geometrical moments, Zernike moments, Legendre moments, Orthogonal Fourier-Mellin moments, and Binary Haar features.

The performance of the whole system in general and every individual step in particular together with failure analyses and reasons for failure are presented in the next chapter.

6. Results Analysis

This chapter is concerned with the evaluation of the performance of the Road Sign Recognition System described in the previous chapter. A series of experiments are carried out for each stage of the RSRS, colour segmentation algorithms, the fuzzy shape recogniser, and the SVM classifier.

The performance of each of the colour segmentation algorithms is evaluated by four experiments. In the first experiment, the overall performance of the colour segmentation algorithms is evaluated by using a number of images which are selected randomly. In the second experiment, each algorithm is evaluated with respect to the environmental and lighting conditions. The third and fourth experiments are concerned with performance time and quality of segmentation, respectively.

Similar experiments are carried out to evaluate the performance of the shape recogniser. This analysis included a comparison among the ability of the shape recogniser to recognise different groups of signs and analysis of the reasons for failure.

The last part of this chapter is an analysis of the classifier performance and the parameters which could affect the classification rate. Five different experiments are carried out to show the performance of different combinations of the SVM models, different SVM types, different kernels and different parameters. These experiments included a study of the effect of different moments and different orders of these moments on the performance of the classification rate.

6.1 Colour Segmentation Algorithms

6.1.1 Performance Evaluation

The performance of colour segmentation algorithms which were described in section 5.4 is evaluated by using four different experiments. They are as follows:

Experiment 1:

In this experiment, the global performance of each of the colour segmentation algorithms is evaluated via 560 signs distributed in 500 images (set 1). These images are

selected randomly and without any repetition using a random number generator. Each algorithm is tested using these signs for complete or incomplete segmentation. Complete segmentation means that the algorithm generates a complete binary object but it may contain a certain amount of noise. Incomplete segmentation means that the object is not fully segmented because of different surface illumination or noise in the image or any other similar reason. It may also be because the algorithm fails completely to produce a segmented image because of colour loss or weather conditions. Table 6.1 depicts the segmentation results of different colours and different segmentation algorithms. In this table, the proposed algorithms are tested to segment the red, yellow and blue colours. Furthermore, the performances of these algorithms are compared with that of de la Escalera [16]. This algorithm is chosen for comparison because of its high robustness and the fact that it is described in a clear way which helps to re-implement it. The Shadow and Highlight invariant algorithm achieved the best performance in the red and blue colour segmentation being 97.6% and 96%, respectively, compared with 91.3% and 89.9% for the de la Escalera algorithm. It achieved the next best result in the yellow colour after the modified version of de la Escalera. This explains why the shadow and highlight invariant algorithm is selected as the main segmentation algorithm in this research.

Table 6.1: Segmentation success rate (%) of different colour segmentation algorithms.

Segmented Colour	Shadow and Highlight invariant	Dynamic Thresholding	Modified de la Escalera	Fuzzy	de la Escalera
Red	97.6	86.3	92.3	91.5	91.3
Yellow	90.0	76.6	90.1	89.4	88.3
Blue	96.0	90.1	91.0	90.0	89.9

Experiment 2:

In the second experiment, the performance of each algorithm is evaluated by using images taken under different light conditions or different effects. The set of images used in this experiment is different from the set of images used in experiment 1 and is called set 2. However, the images are selected using the same criteria of randomness by which the images in experiment 1 are selected. This experiment is carried out for red rim-yellow

interior set of signs as this category includes Swedish speed limit signs, which are of course of importance to highway authorities in Sweden. Images are selected by invoking the Access database application which is used to make a list of images with certain requirements such as weather conditions and different sign properties. These images are used to test the different colour segmentation algorithms. Table 6.2 presents the results of the segmentation of different algorithms tested under different conditions. From this table, the most effective reduction of performance of all algorithms takes place when faded signs are segmented, followed by the effect of fog. The algorithms performed better in the case of snowfall and bad lighting geometry. The best performance is achieved for the case of sunny, blurred or noisy images followed by highlighted signs and images taken in dusk or dawn, and rainfall conditions. When a sign becomes faded, the hue of the colour changes and when the sign is severely faded, the red rim of the sign becomes yellow. This change in hue value makes the segmentation of these signs very difficult. While in the case of fog, a white component of light diffuses in the image, which affects the tint of the colour of the sign.

Table 6.2: Segmentation success rate (%) of different algorithms tested under different effects.

Effect	No. of Signs	Shadow and Highlight invariant	Dynamic Thresholding	Modified de la Escalera	Fuzzy	de la Escalera
Bad Lighting Geometry	48	87.5	56.2	75.0	75.0	75.0
Blurred images	40	97.5	90.0	87.5	85.0	87.5
Dusk/Dawn	66	93.9	78.9	87.8	86.3	87.8
Faded Signs	45	53.3	33.3	33.3	33.3	33.3
Fog	27	74.0	77.7	62.9	62.9	62.9
Highlights	40	97.5	77.5	77.5	75.0	77.5
Noisy images	46	95.6	95.6	93.4	91.3	93.4
Rainfall	44	95.4	75.0	95.4	90.9	95.4
Snowfall	44	88.6	65.9	65.9	65.9	65.9
Sunny	112	98.2	98.2	97.3	96.4	97.3

The Shadow and Highlight colour segmentation algorithm performs the best among all algorithms under all test conditions. The algorithm shows high robustness of segmentation in all conditions except for the case of fog where a certain drop in

performance is noticed. The algorithm shows better performance even for faded signs. It could successfully segment 53% of images selected for this test compared with 33% for the rest of algorithms.

Experiment 3:

This test is to measure the processing time of each segmentation algorithm. Segmentation times from previous experiments are computed for each algorithm and the minimum, maximum, and average times together with standard deviation are calculated. These values which are depicted in Table 6.3 are measured using DELL Latitude D620 computer with 2.0 GHz processor speed. The Shadow and Highlight invariant algorithm shows the best time performance and the best standard deviation which denotes small time variations around the average, followed by the modified version of the de la Escalera. The dynamic thresholding algorithm comes last because of the heavy computations carried out by this algorithm, a burden avoided by the aforementioned algorithms. The Fuzzy segmentation algorithm is not included in this test because it is implemented in MATLAB while the other algorithms are implemented using C++.

Table 6.3: A comparison of the processing Time of different colour segmentation algorithms.

Timing	Shadow and Highlight invariant	Dynamic Thresholding	Modified de la Escalera	de la Escalera
Minimum (sec.)	0.1940	0.2664	0.1935	0.2018
Maximum (sec.)	0.2458	0.3262	0.2637	0.2675
Mean (sec.)	0.2070	0.2782	0.2138	0.2185
Standard deviation	0.0099	0.0106	0.0126	0.0156

Experiment 4:

The last test is concerned with the quality of segmentation. Recognition time is related to the number of objects produced by the segmentation algorithm in a certain image. Therefore, the fewer the objects produced by the segmentation algorithm the better the speed of recognition. In this experiment, the number of objects produced by each segmentation algorithm is computed for 100 images which are randomly selected and the average number of objects is computed. This result is shown in Table 6.4. Dynamic thresholding and shadow and highlight invariant algorithms produce fewer objects compared with other algorithms which produce 40 times the number of objects.

This high number of objects must definitely be filtered as most of them are few pixels in size or they will slow down the recognition speed.

Table 6.4: Average number of objects generated by different segmentation algorithms.

	Shadow and Highlight invariant	Dynamic Thresholding	Modified de la Escalera	de la Escalera
Average No. of objects	14	11	463	439

Figure 6.1 depicts a comparison of the output of the shadow and highlight invariant algorithm and the de la Escalera algorithm. It is clear that the segmented image generated by the latter is very noisy. By comparing the time needed for recognition, the shadow and highlight invariant algorithm takes 0.215 seconds while recognition based on the de la Escalera algorithm takes 10.99 seconds and it fails to recognise the sign in the image.

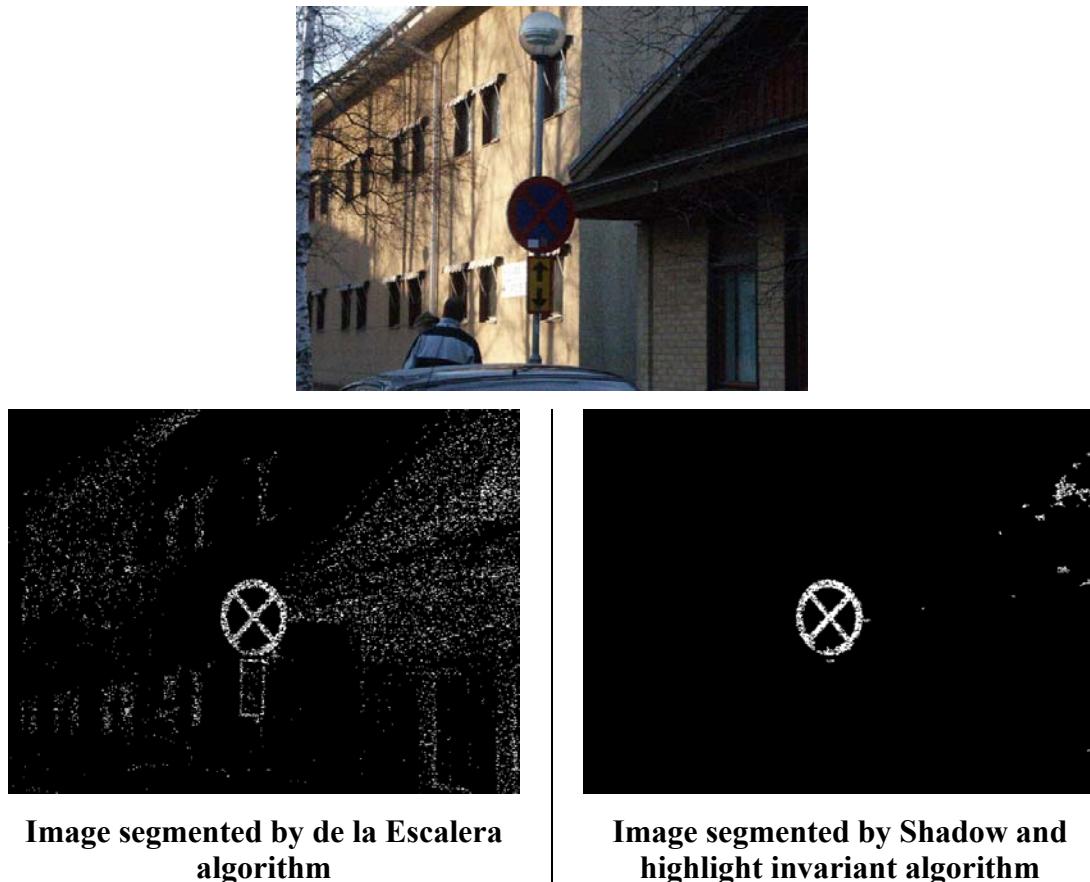


Figure 6.1: Comparison of quality of segmentation of two different algorithms.

6.1.2 Failure Analysis

Effect of Fog:

The segmentation of images collected in foggy conditions shows the worst results compared to other weather conditions. The reason is that a white light component is diffused between the camera and the object which affects the tint of the colour. Figure 6.2 depicts an image taken in foggy conditions. The segmentation of this image fails to give any results. The colour constancy algorithm described earlier in chapter 5 is an effective algorithm to recover colours in such cases and helps to give better segmentation. In Figure 6.2, a comparison of the results before and after applying the colour constancy algorithm is depicted. In the upper row, the image is segmented directly using the shadow and highlight invariant algorithm which gives very poor segmentation results, and in the lower row, it is enhanced with the colour constancy algorithm and then segmented by the same algorithm. This improvement in segmentation helps correct recognition of the sign.

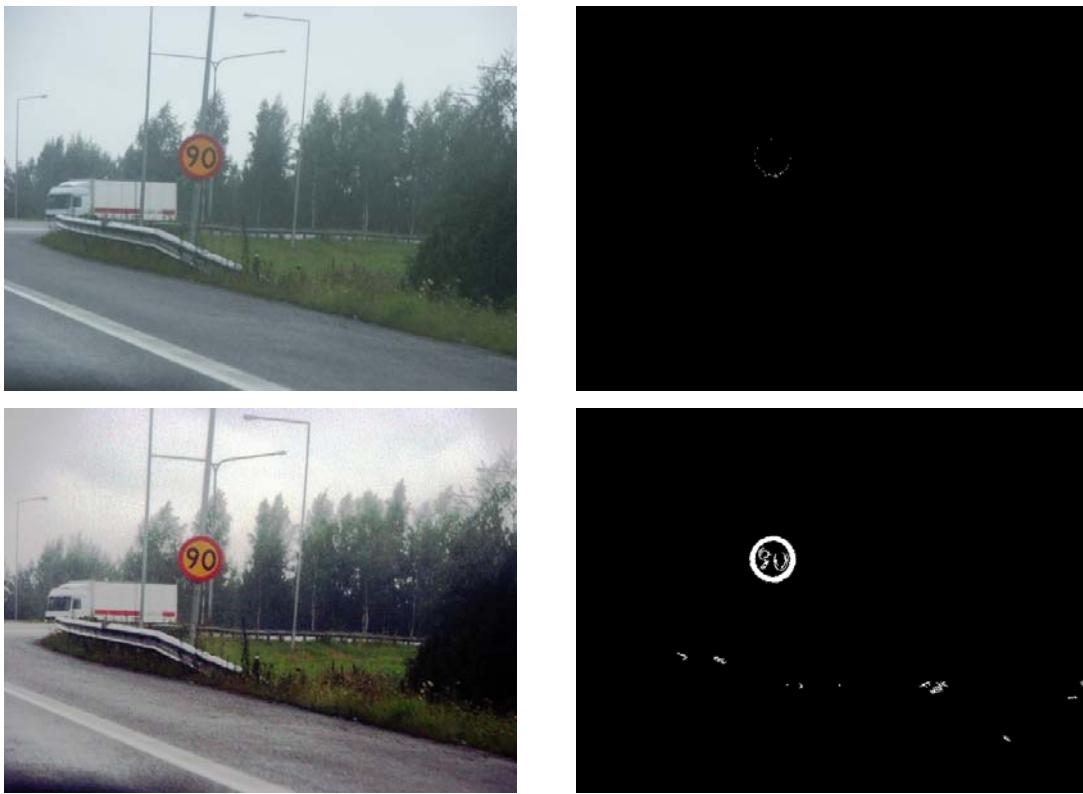


Figure 6.2: The effect of fog and how it can be treated.

Instability of yellow colour segmentation:

One reason for the drop of the segmentation success rate is the failure of algorithms in segmenting yellow. Regardless of the quality of the sign and its age, this failure happens when the image is taken from a certain angle in which a different yellow (hue) can be seen. Figure 6.3 shows one example of such a case. When the segmentation of yellow fails, recognition of the sign also fails, as it based on combining red-yellow colour for sign recognition. This failure in colour segmentation is responsible for 30% of all failures in the recognition stage which is very significant.



Figure 6.3: Instability of segmentation with yellow.



Figure 6.4: Yellow Instability because of low illumination.

Loss of yellow in dark images:

Dark images can represent another reason for segmentation failure where yellow colour is concerned. When an image is dark due to a wrong camera setting, for instance, the colour hue will either be in the unstable or in the achromatic area of the HSV colour space. Segmentation of hue in these two regions does not give any significant result, and

hence poor quality segmentation is produced. Figure 6.4 depicts such case. About 15% of the whole loss of performance of recognition is caused by this problem.

6.2 The Recognition Stage

6.2.1 Performance Evaluation

The performance of the recognition stage which was described in sections 5.5 and 5.6 is evaluated. Images segmented by the Shadow and Highlight invariant algorithm are then recognised by the fuzzy recogniser. Hence, the performance shown here is the resultant of the performance of colour segmentation algorithm and the performance of the recogniser.

Experiment 1:

The main objective of this experiment is to evaluate the global performance of the recogniser. The set of images selected in this experiment is the one used in experiment 1 in the preceding section and called set 1. The recogniser achieves a successful recognition of 88.4 %. This result is depicted in the first row of Table 6.5.

Table 6.5: Recognition rates of traffic signs in different test conditions.

Test Condition	Image Set	No. of Signs	Recognition Rate %
Overall Performance	1	560	88.4
Bad Lighting Geometry	2	48	81.2
Blurred	2	40	92.5
Dusk/Dawn	2	66	90.9
Faded Signs	2	45	53.3
Fog	2	27	81.4
Highlights	2	40	95.0
Noisy images	2	46	73.9
Occluded Signs	2	32	56.2
Rainfall	2	44	95.4
Snowfall	2	44	90.9
Sunny	2	112	94.6

Experiment 2:

In this experiment, images in set 2 are used. The performance of the recogniser is tested by recognising red rim-yellow interior signs in different test conditions such as

different sign conditions, weather conditions, lighting geometry, and image conditions. The best performance achieved by the recogniser is 95% for images collected in sunny conditions, during rainfall, and signs with highlights. The worst performance is for faded and occluded signs which achieved 53.3% and 56.2%, respectively. Table 6.5 illustrates the results of these tests.

Experiment 3:

The third test is carried out to measure the processing time. Red rim-yellow interior signs in 100 images are recognised and time of recognition is calculated for every image. Table 6.6 presents the processing time achieved by the recogniser. The time needed to recognise a sign varies depending on the distance between the camera and the sign. The minimum time required to recognise a sign located at a long distance is 0.2 seconds, while the maximum time required to recognise a close sign is 1.5 seconds. The reason for this time difference is the big area of the sign which is located close to the camera, and hence more time is spent to calculate connected components labelling and shape measures. Further discussion about this issue will be presented in the next subsection.

Table 6.6: Time of recognition.

Timing	Value
Minimum (sec.)	0.2040
Maximum (sec.)	1.5380
Average (sec.)	0.5824
Standard Deviation	0.3493

6.2.2 Failure Analysis

Occluded signs: While the recognition algorithm shows a high rate of success in most test conditions, the next worst recognition result is for occluded signs. The main reason for this significant drop in performance is that the algorithm is based on shape measures which indicate the similarity of the object under consideration with the basic shapes, i.e. rectangle, ellipse, triangle, and octagon. When the sign is occluded, its shape departs from these basic shapes and the fuzzy system rejects it as it is not close to any of these shapes. This problem is of great interest and is discussed in more detail in the future work section.

False Alarms: Experiment 1 and 2 show that the system fails to recognise some of the traffic signs for different reasons such as weather conditions, sign condition or occlusion. This is called false negatives. The figures of false negative generated by the system are listed in Table 6.5. In addition to the false negatives, there is a number of false positives generated by the system due to the presence of objects similar to traffic signs in the image. Figure 6.5 shows two examples of these false positives. In the example on the left, the interior of the sign is detected as a standalone sign because of the presence of a yellow blob in the convex hull of a red object. In addition to this the convex hull of the external object is an oval shape which makes the system think that it is a sign. In the example on the right, the traffic light is detected as a sign due to the reasons mentioned in the previous example. The number of false positives generated by the recogniser is around 1% of the images tested in the aforementioned experiments. The reason for this low number of false positives is the design of the recogniser which is based on combining colour and shape information to recognise the sign.



Figure 6.5: False Positives.

Speed of Recognition: The speed of recognition is related to the size of object in the scene. When a big object is encountered, the system takes more time to recognise that object compared to a small one. In order to come up with a concrete result, a sign was photographed at 40, 60, and 80 meters respectively, and the time of recognition was computed for every image. This experiment was repeated 10 times and the minimum, maximum and the average values were computed. Table 6.7 shows the results. The sign at 80m took 0.356 seconds to be recognised compared with 0.841 seconds for the 40m sign. When an image without a sign is introduced to the recogniser or when the sign is

faded, the recogniser takes about 0.06 seconds to decide that. Two parts of the system are computationally expensive: the connected components labelling algorithm and the shape measures. The last one is much more computationally expensive than the first parameter. This part of the work will be improved in the future either by improving the algorithm itself, or by moving to another kind of shape measures.

Table 6.7: Speed of recognition is affected by size of the sign.

Image	Distance (m)	Minimum Recognition Time (sec.)	Maximum Recognition Time (sec.)	Average Recognition Time (sec.)
	80	0.355	0.357	0.356
	60	0.578	0.580	0.579
	40	0.839	0.844	0.841

Sensitivity to connected signs: The algorithm fails to detect traffic signs when two signs are attached together. The overall shape of the two signs departs from the basic shapes which the algorithm is designed to recognise. Morphological opening is used to separate some of these attached signs when the attachment is not so severe. It is successful in many cases but it fails when the two signs are very close to each other such as the one

shown in Figure 6.6. This failure in the recognition stage reduces the number of objects correctly recognised by the system and the entire performance is affected negatively.



Figure 6.6: Sensitivity to connected signs.

6.3 The Classification Stage

6.3.1 Classification with Different Features

Tables 6.8 and 6.9 show the average classification rates of the ten experiments described in section 5.8. The performances of road sign shape classification with binary representation are identical and appear to be extremely high, achieving 100% accuracy, on all dataset pairs. None of these instances are classified incorrectly for either the training or testing datasets. However, it is a little bit lower than that when features are used.

Table 6.8: Average classification rate of traffic sign rims with different features.

Feature	Training%	Testing%
Binary Images	100	100
Geometric Moments ($p = 7, q = 7$)	95.8	96.2
Zernike Moments ($p_{\min} = 5, p_{\max} = 12$)	100	98.9
Legendre Moments (min=0, max=12)	100	99.7
Orthogonal Fourier-Mellin Moments ($p_{\min} = 0, p_{\max} = 7, q = 8$)	96.6	94.7
Binary Haar Features ($\Delta x = 7, \Delta y = 7$)	99.6	99.1

The feature which showed the best results is the Legendre moments followed by Zernike moments and the other features come after that. Whilst this is not quite as high as

the use of binary images, using features is much faster than binary images in the training and testing phases. This is because of the number of attributes of binary images which are much higher compared to the number of attributes for features, as shown in Table 6.10.

Table 6.9: Average classification rate of speed-limit signs with different features.

Features	Training%	Testing%
Binary Images	100	98.1
Geometric Moments ($p = 7, q = 7$)	78.5	68.6
Zernike Moments ($p_{\min} = 5, p_{\max} = 12$)	98.8	88.1
Legendre Moments(min=0, max=12)	100	98.3
Orthogonal Fourier-Mellin Moments ($p_{\min} = 0, p_{\max} = 7, q = 8$)	53.4	50.0
Binary Haar Features ($\Delta x = 7, \Delta y = 7$)	98.1	94.7

For the classification of Speed-Limit signs with binary images, the average performances are 100% for training sets and 98.1% accuracy for test sets. Those road sign images which are classified incorrectly had very poor image qualities.

From this discussion, it is concluded that using features has the advantage of reducing the amount of computation needed by the system without reducing the classification abilities.

The results in Tables 6.8 and 6.9 reflect the classification under certain circumstances in which the results vary if different conditions are tested. Orthogonal Fourier-Mellin moments, for instance, can give much better classification results if the SVM is trained under different parameter values. This will be shown in the coming subsections.

Table 6.10: Default values of γ computed for different features.

Feature	No. of Attributes	γ
Binary Images	1296	0.00077
Geometric Moments	64	0.01562
Zernike Moments	40	0.02500
Legendre Moments	169	0.00590
OFMM	72	0.01388
Binary Haar Features	63	0.01587

6.3.2 Classification with Different Kernels and SVM Types

This part focuses on the analysis of the performance of the SVM using different kernels and different SVM types. The SVM recognition model is trained using four basic kernels: Linear, Polynomial, Radial Basis Function (RBF) and Sigmoid. It is tested using two kinds of SVM types, the C-SVM and the ν -SVM. Four groups of experiments are carried out. Each group of experiments is carried out using the same pair of training/test dataset. Other parameters of the SVM model are given as:

$$C = 1, \nu = 0.5, \gamma = 1/n, r = 0, d = 3.$$

where n is the number of attributes for an input vector.

Tables 6.11 - 6.16 show the experimental results of training and testing the SVM recognition model with binary images and for different features for both traffic sign rims and Speed-Limit signs.

Table 6.11: Classification rates of sign rims and Speed-Limit signs using different kernels and SVM types when binary images are used.

SVM Type	Kernel	Sign Rims		Speed-Limits	
		Training %	Testing %	Training %	Testing %
C-SVM	Linear	100	100	100	100
	Polynomial	100	100	100	100
	RBF	100	100	100	100
	Sigmoid	100	100	100	98.7
ν -SVM	Linear	100	98.9	100	98.9
	Polynomial	100	97.8	100	98.7
	RBF	100	98.9	100	98.7
	Sigmoid	100	98.9	100	98.7

Table 6.12: Classification rates of sign rims and Speed-Limit signs using different kernels and SVM types when Geometric moments are used.

SVM Type	Kernel	Sign Rims		Speed-Limits	
		Training %	Testing %	Training %	Testing %
C-SVM	Linear	95.7	95.5	81.1	70.6
	Polynomial	68.6	66.6	49.7	46.6
	RBF	70.0	66.6	47.4	37.3
	Sigmoid	64.3	60.0	49.7	44.0
ν -SVM	Linear	95.7	97.8	84.6	73.3
	Polynomial	96.2	97.8	86.3	74.7
	RBF	96.2	97.8	84.0	69.3
	Sigmoid	95.7	97.8	82.3	70.7

Table 6.13: Classification rates of rim shapes and Speed-Limit signs using different kernels and SVM types when Zernike moments are used.

SVM Type	Kernel	Sign Rims		Speed-Limits	
		Training %	Testing %	Training %	Testing %
C-SVM	Linear	100	100	97.7	89.3
	Polynomial	93.3	90.0	75.4	64.0
	RBF	99.0	98.9	95.4	86.7
	Sigmoid	98.1	96.7	90.8	82.7
ν -SVM	Linear	99.0	98.9	96.0	89.3
	Polynomial	96.7	93.3	99.4	89.3
	RBF	99.0	98.9	96.6	89.3
	Sigmoid	99.0	98.9	96.0	85.3

Many of the experiments showed that the linear kernel combined with C-SVM had a better performance than others. One of the reasons is that the linear kernel normally has a good performance when the number of attributes is large. The leading cause, however, is that the accuracy of a SVM model is highly dependent on the selection of the model parameters.

Table 6.14: Classification rates of rim shapes and Speed-Limit signs using different kernels and SVM types when Legendre moments are used.

SVM Type	Kernel	Sign Rims		Speed-Limits	
		Training %	Testing %	Training %	Testing %
C-SVM	Linear	100	98.9	100	98.7
	Polynomial	94.7	94.4	97.7	97.3
	RBF	100	98.9	98.8	97.3
	Sigmoid	100	98.9	99.4	97.3
ν -SVM	Linear	100	98.9	98.8	97.3
	Polynomial	100	98.9	100	97.3
	RBF	100	98.9	98.8	97.3
	Sigmoid	100	98.9	98.8	97.3

Classification of traffic sign rims and Speed-Limit signs perform better results when OFMM is combined with ν -SVM rather than C-SVM. This is regardless of the type of kernel used as shown in Table 6.15.

The last point to be mentioned here is that there is a drop in the performance of the classification system when the polynomial kernel is used.

Table 6.15: Classification rates of rim shapes and Speed-Limit signs using different kernels and SVM types when OFMM are used.

SVM Type	Kernel	Sign Rims		Speed-Limits	
		Training %	Testing %	Training %	Testing %
C-SVM	Linear	97.6	94.4	61.7	57.3
	Polynomial	83.3	82.2	45.7	38.7
	RBF	83.3	82.2	58.3	54.7
	Sigmoid	83.3	82.2	60.6	60.0
ν -SVM	Linear	94.8	94.4	93.7	90.7
	Polynomial	94.8	94.4	91.4	89.3
	RBF	95.2	94.4	94.3	90.7
	Sigmoid	95.2	94.4	85.7	84.0

Table 6.16: Classification rates of rim shapes and Speed-Limit signs using different kernels and SVM types when Binary Haar Features are used.

SVM Type	Kernel	Sign Rims		Speed-Limits	
		Training %	Testing %	Training %	Testing %
C-SVM	Linear	99.5	98.8	97.1	93.3
	Polynomial	66.2	65.6	35.4	30.6
	RBF	97.6	98.9	86.8	84.0
	Sigmoid	97.6	97.8	77.7	77.3
ν -SVM	Linear	96.6	97.8	94.3	90.7
	Polynomial	94.2	90.0	86.3	81.3
	RBF	97.1	97.7	93.7	90.7
	Sigmoid	96.7	97.8	94.3	90.7

6.3.3 Performance of SVM with Different Parameters

To analyse how the performance of SVM is affected by changing the values of parameters, the worst pair of training/test dataset of Speed-Limit signs in each moment type, which is shown in Appendix D, is chosen to train and test the SVM classification model in a set of experiments which is illustrated in this section.

1. The Parameter C

The parameter C defines the upper bound of α in the C-SVM model; it is a trade-off between maximum margin and classification error. Figure 6.7 shows the performance of the SVM model using linear kernel and C-SVM type with different values of parameter C . There is no kernel parameter in the linear kernel; therefore the parameter C is the only variable in this model. A higher value of the parameter C allows α to have a large value. The accuracy of classification could be improved by increasing α . However, it is not useful to define an excessive upper bound of α .

A good example to illustrate this case is depicted in Table 6.9. According to this table, the performance of the Orthogonal Fourier-Mellin Moments is not as good as the others. This can be improved by using higher C values. As shown in Figure 6.7, the classification rate can be improved from about 40% up to 88% only by changing the value of C .

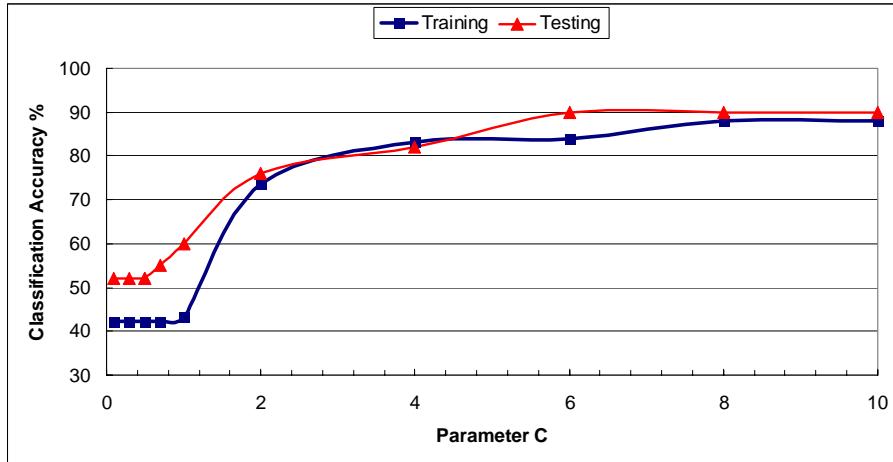


Figure 6.7: The effect of parameter C on the classification accuracy of the SVM classifier when C -SVM and Linear kernel are used.

2. The Parameter ν

The ν -SVM uses another parameter ν which was defined in chapter 4, to control the number of margin errors and the number of support vectors. Figures 6.8 and 6.9 show the performance of the SVM model using the linear kernel and ν -SVM type with different values of ν . The number of support vectors increases by increasing the value of ν . However, since the number of margin errors also increases, the accuracy of the training classification decreases and the overall accuracy of the classification of the test set decreases.

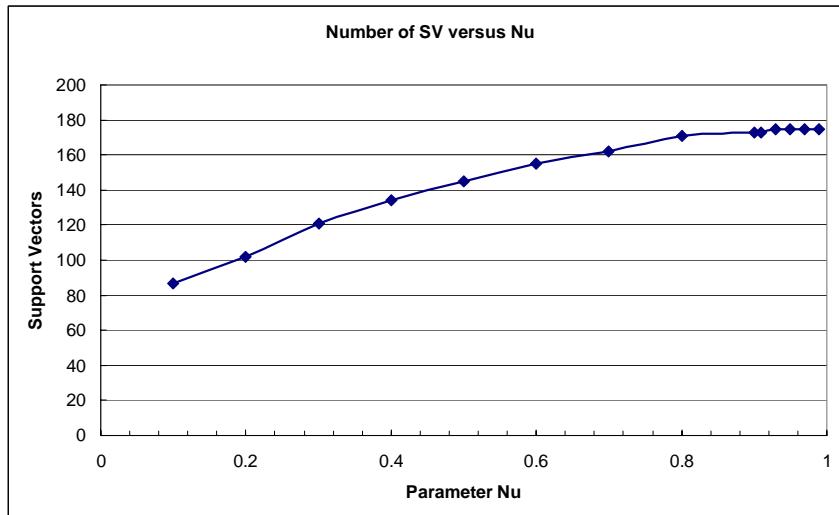


Figure 6.8: Number of support vectors versus parameter ν of SVM model using ν -SVM model and linear kernel.

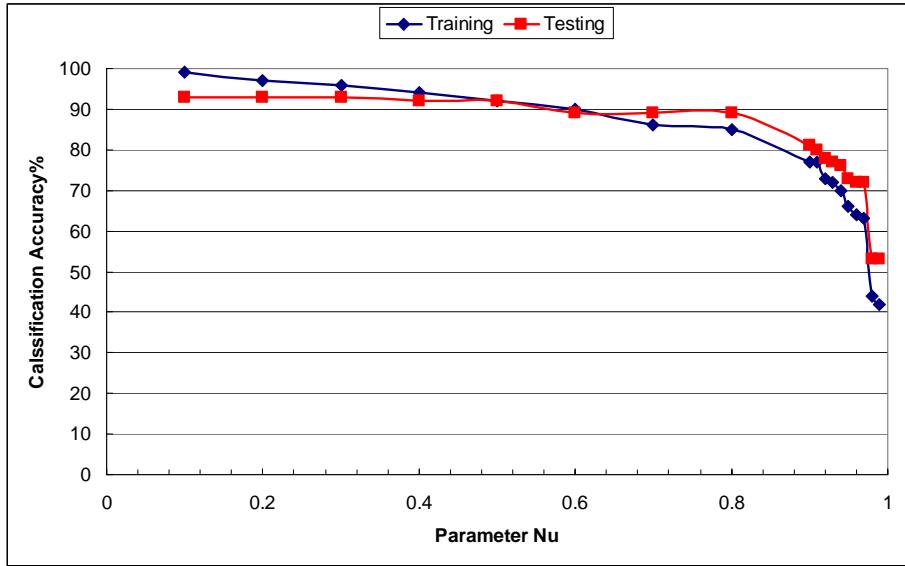


Figure 6.9: Performance of the SVM model using Linear kernel and different values of parameter ν .

3. The Parameter γ

The RBF kernel contains one parameter which is called γ . The parameter γ is normally given a very small value. In all of the above experiments, it is initialised as $\gamma = 1/n$, where n is the number of attributes for an input vector. Figure 6.10 illustrates the performance of SVM with different values of γ . The training classification improves by increasing the value of γ .

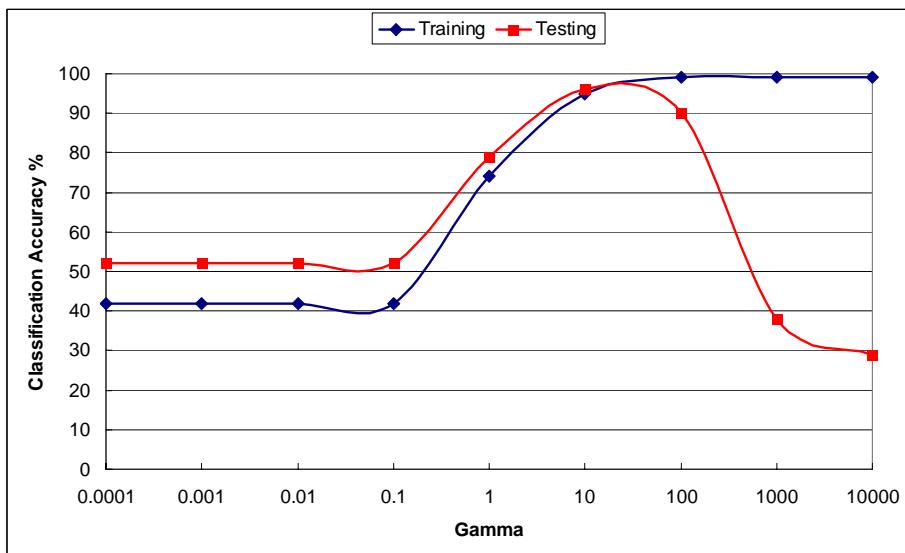


Figure 6.10: Performance of the SVM model using C-SVM, RBF kernel, $C = 1$ and different values of parameter γ

The generalisation of the SVM model, however, degrades when the value of γ exceeds a certain limit. In the aforementioned experiments, the number of attributes differs according to the type of moments under consideration. Therefore different values of γ are used. However, it is always good to start with $\gamma = 1/n$, but it is not always the right choice and some experiments are needed before the optimum value is reached.

4. The Parameter r

In addition to the parameter γ , the sigmoid kernel has another parameter which is known as r . This parameter is also used in the polynomial kernel. It is not a very important one and normally it is initialised to zero. Figure 6.11 shows the performance of the SVM with different values of r . The accuracy of classification degrades by increasing the value of r . In this Figure a logarithmic scale is used to show the effect on a wide range of this variable.

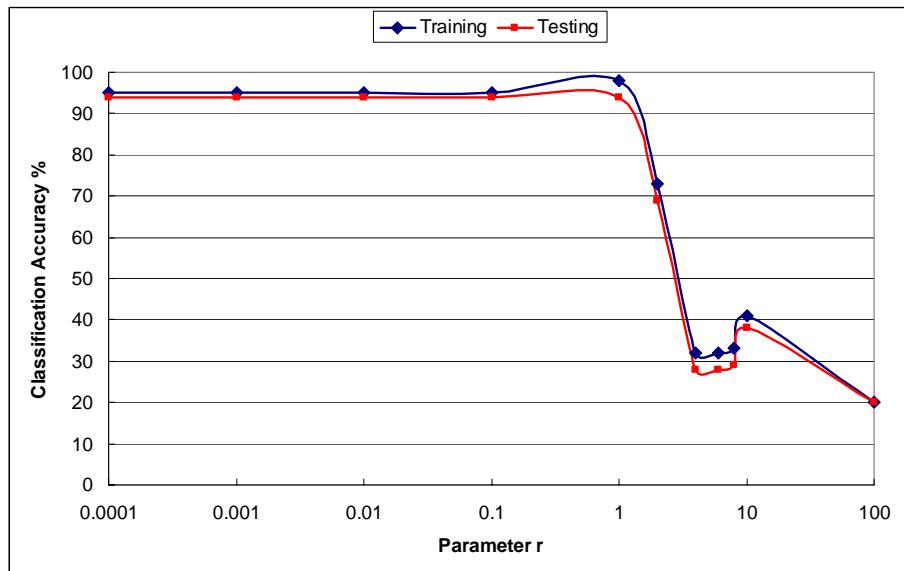


Figure 6.11: Performance of the SVM model using C-SVM, Sigmoid kernel, $C = 1, \gamma = 0.1$ and different values of parameter r .

5. The Parameter d

There are three kernel parameters in the polynomial kernel, γ , r and d . Figure 6.12 shows the performance of classification corresponding to different values of d . Normally, small values of d are good choices. On the whole, the generalisation of the SVM model drops by increasing the value of d .

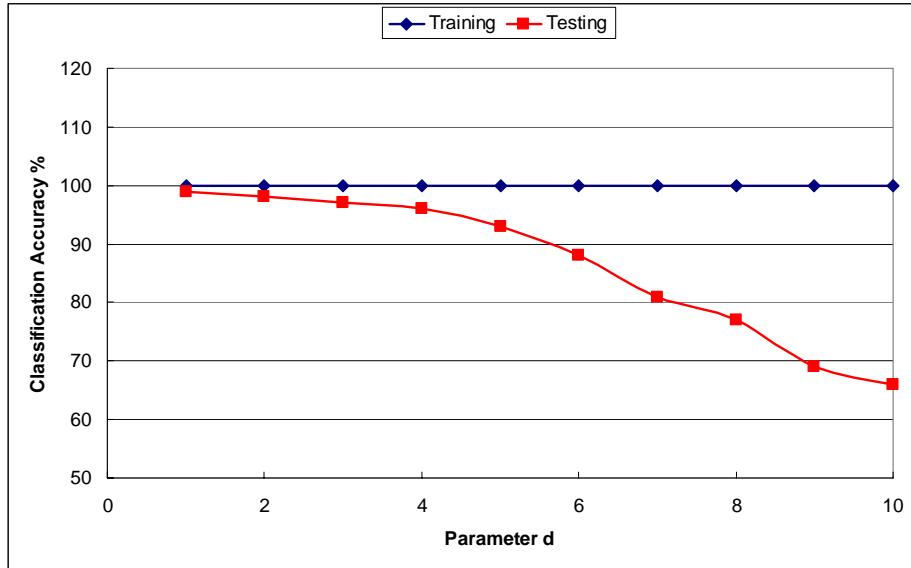


Figure 6.12: Performance of the SVM model using C-SVM, polynomial kernel, $C = 1$, $\gamma = 1$, $r = 0$ and different values of parameter d .

6.3.4 Classification with Different Moment Orders

The next set of experiments illustrates the effect of changing the order of features on the classification rate. Without any exception, increasing the order of moments improves the classification rate. However, the exact increase of order differs from one type of moments to another. Generally speaking, the lower orders of the moments describe the general shape of the object under classification, while the higher orders describe the fine details. However, high order moments can easily be affected by noise. Therefore, a compromise between the two issues should always be taken into consideration.

This experiment is applied on one set of train/test datasets of Speed-Limit signs listed in Appendix D. It is important to illustrate the effect of the order of moments on the classification rate. All experiments are carried out using C-SVM, Linear Kernel, and $C=1$. Values of moments are changed from lower values up to a certain value and for every step, the classification rate is computed. The results of this test are depicted in Figures 6.13-6.17. Geometric moments show that they reach the maximum classification by order 6, while Zernike moments reach the maximum by order 10. Legendre moments and Binary Haar features show similar curves and they reach the maximum classification rate by orders 6 and 8 respectively.

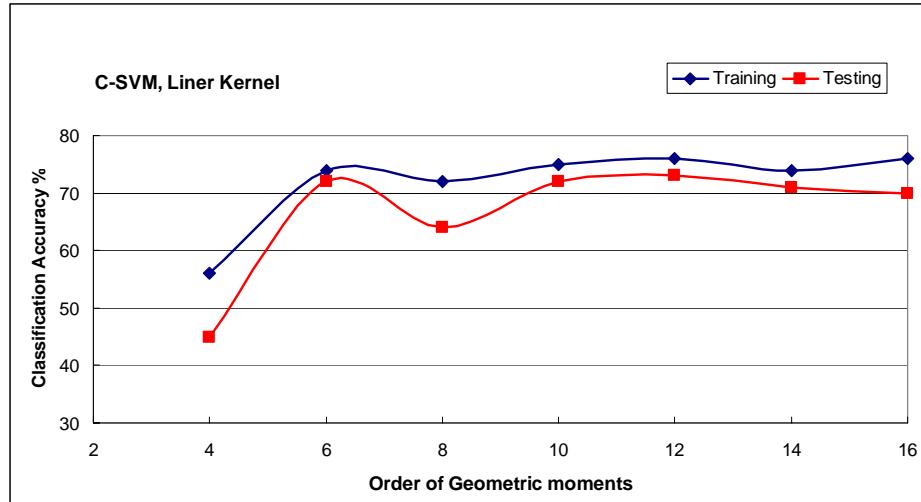


Figure 6.13: Effect of the order of Geometric moments order on classification rate when C is constant.

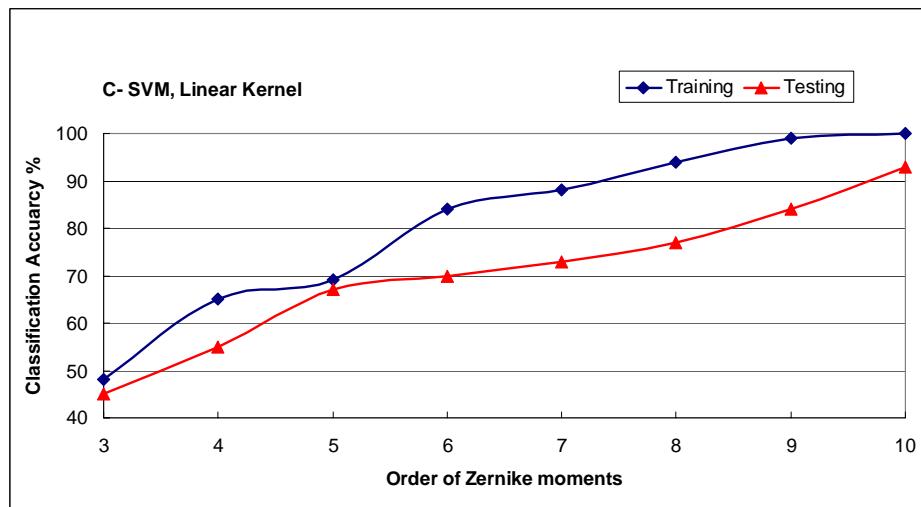


Figure 6.14: Effect of the order of Zernike moments on classification rate when C is constant.

Some curves such as OFMM show some kind of fluctuation, but the general trend of the classification rate curve improves as the order is increased. The reason behind these fluctuations is the effect of noise on the training and testing datasets as the order of moment increases.

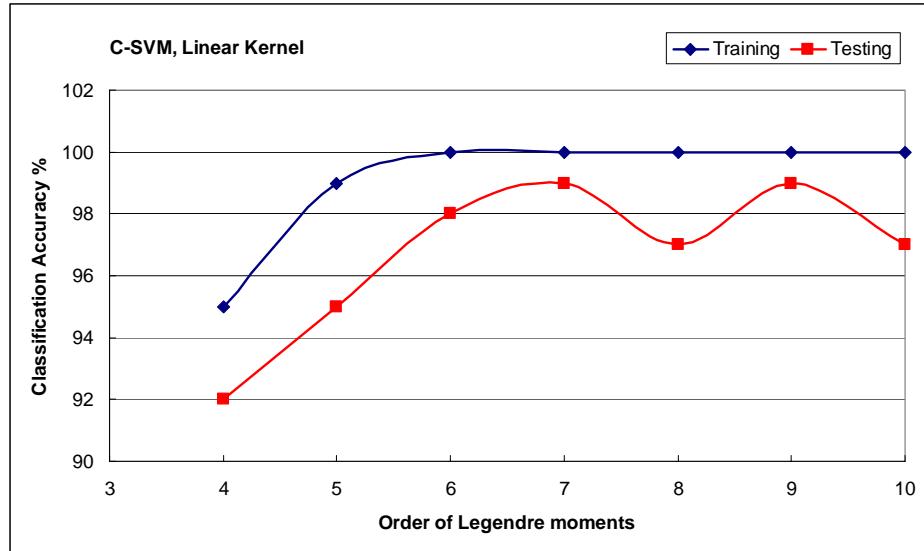


Figure 6.15: Effect of the order of Legendre moments on classification rate when C is constant.

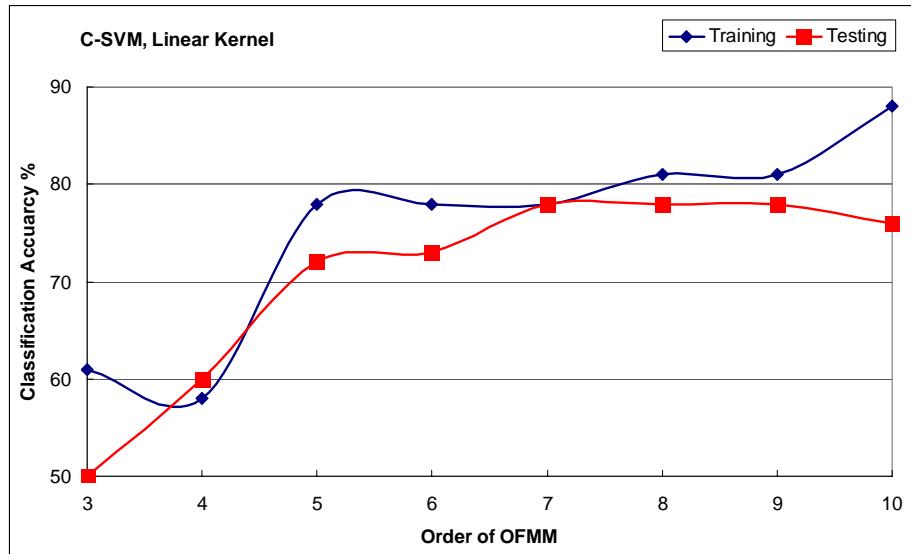


Figure 6.16: Effect of the order of OFMM moments on classification rate when C is constant.

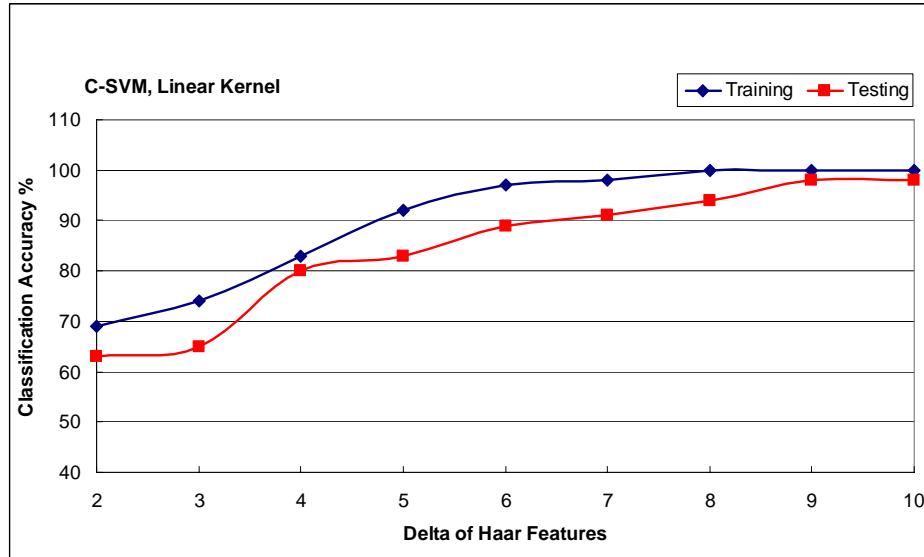


Figure 6.17: Effect of $(\Delta x + \Delta y)$ of Haar Features on classification rate when C is constant.

6.3.5 Classification Time

The timing performance of the SVM classifier is evaluated via computing the training and testing times of different features together with that of the 36x36 pixel normalised images.

Table 6.17: A comparison of training and testing times using different features and normalised images.

Feature	Training Time (sec.)	Testing Time (sec.)
Geometric Moments - shape	0.187	0.047
Geometric Moments - speed limit	0.141	0.046
Zernike - shape	0.110	0.047
Zernike – speed limit	0.140	0.047
Legendre - shape	0.140	0.063
Legendre – speed limit	0.109	0.047
OFMM - shape	0.109	0.063
OFMM – speed limit	0.109	0.079
Haar - shape	0.094	0.047
Haar – speed limit	0.109	0.047
Images - shape	1.109	0.532
Images – speed limit	1.234	0.531

Table 6.17 depicts the results of this test. From the table, training and testing the SVM with features is much faster than that with images. However, the computation time

of some of these features such as Zernike moments and Legendre moments is relatively high. This will not cause any problems for the classification as these computations can be carried out at any time outside the experiment time and can be saved in a file which can be retrieved when the experiment is carried out.

6.3.6 Search for Optimal Parameters

Two types of search mechanisms are implemented to find optimal parameters: Grid search and Simulated Annealing (SA). Grid search could be feasible to find the optimal parameters of an SVM model when the number of model parameters is small, normally less than three. The time complexity of the search is increased exponentially by increasing the number of parameters, making optimality an impossible goal. Heuristic search is a good choice to find near optimal parameters in an efficient manner. SA search is one heuristic search methods, which works efficiently when the number of model parameters is more than one.

The real search space is infinite. This means that it is impossible to search all the space with the grid search. The upper bound (UB), lower bound (LB) and search step for each parameter are defined, so that the search space is partitioned geometrically. The search regions of kernel parameters are defined as:

$$C = \{2^i \mid LB\ i = -5, UB\ i = 15, step\ i = 1\}$$

$$\gamma = \{2^j \mid LB\ j = -15, UB\ j = 3, step\ j = 1\}$$

$$\nu = \{k \mid LB\ k = 0.1, UB\ k = 1, step\ k = 0.1\}$$

Grid Search:

Grid search works as an exhaustive search, where the size of the search step controls the complexity of the search. As the model must be evaluated at every grid region and the time complexity of the search is increased exponentially with the increase of parameters, the grid search of the SVM model is tested with the RBF kernel only.

The same pairs of training/test datasets used in the former experiments are used here. The performances with and without grid search for the SVM model are illustrated in

Tables 6.18 and 6.20. Tables 6.19 and 6.21 show the corresponding values of C, γ , and ν .

Table 6.18: Classification rates of grid search for sign rims when C-SVM and ν -SVM are used.

Features	C-SVM		ν -SVM	
	Training% Bfr (Aftr)	Testing% Bfr (Aftr)	Training% Bfr (Aftr)	Testing% Bfr (Aftr)
Binary Images	100 (100)	100 (100)	100 (100)	100 (100)
Geometric Moments	100 (100)	92.2 (97.7)	100 (100)	92.2 (97.7)
Zernike Moments	100 (100)	96.6 (96.6)	100 (100)	96.6 (96.6)
Legendre Moments	100 (100)	98.9 (98.9)	100 (100)	98.9 (98.9)
OFFM	97.1 (99.5)	90.0 (92.0)	97.1 (99.5)	90.0 (93.3)
Binary Haar Features	100 (100)	97.7 (97.7)	100 (100)	97.7 (97.7)

Table 6.19: SVM parameters of grid search for traffic signs rims when C-SVM and ν -SVM are used.

Features	C-SVM		ν -SVM	
	C	γ	ν	γ
Binary Images	4	0.000122	0.1	0.000122
Geometric Moments	64	0.5	0.1	0.5
Zernike Moments	4	0.03125	0.2	0.03125
Legendre Moments	1	0.012562	0.1	0.015625
OFMM	1	4	0.1	4
Binary Haar Features	8	0.0625	0.1	0.25

After grid search the performances of the SVM model with RBF kernel are remarkably improved, sometimes even better than that of the linear kernel. Grid search,

however, only finds a region near the global optimal point. The global optimal point could be found after combining the pattern search with the grid search.

Table 6.20: Classification rates of grid search for Speed-Limit signs when C-SVM and ν -SVM are used.

Features	C-SVM		ν -SVM	
	Training% Bfr (Aftr)	Testing% Bfr (Aftr)	Training% Bfr (Aftr)	Testing% Bfr (Aftr)
Binary Images	100(100)	96.0(96.0)	100(100)	96.0(96.0)
Geometric Moments	81.7 (99.4)	60.0 (72.0)	81.7 (96.5)	60.0 (72.0)
Zernike Moments	100 (98.2)	82.7 (85.3)	100 (96.0)	82.7 (86.6)
Legendre Moments	100 (100)	97.3 (96.0)	100 (100)	97.3 (96.0)
OFMM	49.1 (98.8)	42.7 (94.7)	49.1 (99.4)	42.7 (99.4)
Binary Haar Features	98.8 (100)	90.7 (90.7)	98.8 (99.4)	90.7 (92.0)

Table 6.21: SVM parameters of grid search for Speed-Limit signs when C-SVM and ν -SVM are used.

Features	C-SVM		ν -SVM	
	C	γ	ν	γ
Binary Images	2	0.0009765	0.1	0.0009765
Geometric Moments	4096	0.0078125	0.2	0.015625
Zernike Moments	64	0.0039106	0.3	0.000122
Legendre Moments	2	0.015625	0.1	0.015625
OFMM	4	8	0.2	8
Binary Haar Features	20480	0.000976	0.1	0.001953

Simulated Annealing:

A Simulated Annealing (SA) search is also called a Monte Carlo Annealing derived from the process of physical crystal formation. It starts with some solution which is totally random and changes it to another solution which is similar to the previous one. It makes slight changes to the result until it reaches a result close to the optimum. The algorithm uses random numbers in its execution. Therefore, it comes with a different result every time the program is executed. An important part of the simulated annealing process is how the inputs are randomised. This randomisation process takes the previous values of the inputs and the current temperature as inputs. The input values are then randomised according to the temperature. A higher temperature will result in more randomisation while a lower temperature will result in less randomisation. The new solution is always accepted if it gives a better evaluation than the current one; otherwise it is accepted with some probability p . The probability of acceptance is defined as:

$$p = \frac{1}{1 + e^{\frac{eval(V_c) - eval(V_n)}{T}}} \quad (6.1)$$

where V_c is the current point, V_n is the new neighbour and T is an additional parameter looked at such as temperature.

There are two major processes which occur during the simulated annealing algorithm. First, for each temperature the simulated annealing algorithm runs through a predetermined number of cycles. As the cycle runs the inputs are randomised. Once the specified number of training cycles has been completed, the temperature can be lowered and checked whether it has reached the lowest permissible temperature. If the temperature is not lower than the lowest allowed permissible one, then it is lowered and another cycle of randomisation will take place. If the temperature is lower than the minimum temperature allowed, the simulated annealing algorithm is complete. The cooling ratio, the rate at which the temperature is reduced, is another important definition of the SA search. Similar to the process of crystal formation, the solution is not good if the cooling is done too quickly or too slowly.

The performances with and without SA search for the SVM model are depicted in Tables 6.22 and 6.24. Tables 6.23 and 6.25 show the corresponding values of C , γ , and ν . The algorithm gives better results in the majority of the cases. However, there is a drop of the performance in some cases such as the case of using OFFM which is indicated in Table 6.22.

Table 6.22: Classification rates of SA search for sign rims when C-SVM and ν -SVM are used.

Features	C-SVM		ν -SVM	
	Training% Bfr (Aftr)	Testing% Bfr (Aftr)	Training% Bfr (Aftr)	Testing% Bfr (Aftr)
Binary Images	100 (100)	100 (100)	100 (100)	100 (99.0)
Geometric Moments	100 (100)	92.2 (97.7)	100 (100)	92.2 (97.7)
Zernike Moments	100 (100)	96.6 (96.6)	100 (100)	96.6 (96.6)
Legendre Moments	100 (100)	98.9 (98.9)	100 (100)	98.9 (98.9)
OFFM	97.1 (99.5)	90.0 (94.4)	97.1 (98.6)	90.0 (94.4)
Binary Haar Features	100 (100)	97.7 (97.7)	100 (100)	97.7 (97.7)

Table 6.23: SVM parameters of SA search for traffic signs rims when C-SVM and ν -SVM are used.

Features	C-SVM		ν -SVM	
	C	γ	ν	γ
Binary Images	30.47	0.0002548	0.39	3.05e-005
Geometric Moments	32768	0.0003776	0.02	0.369828
Zernike Moments	3.699	0.113393	0.10	0.005606
Legendre Moments	5.806	0.011629	0.04	0.015008
OFMM	0.7972	8	0.08	0.007355
Binary Haar Features	54.53	0.05022	0.01	0.016551

Table 6.24: Classification rates of SA search for Speed-Limit signs when C-SVM and ν -SVM are used.

Features	C-SVM		ν -SVM	
	Training% Bfr (Aftr)	Testing% Bfr (Aftr)	Training% Bfr (Aftr)	Testing% Bfr (Aftr)
Binary Images	100 (100)	100 (96.0)	100 (100)	100 (96.0)
Geometric Moments	81.7 (96.0)	60.0 (72.0)	81.7 (91.4)	60.0 (72.0)
Zernike Moments	100 (100)	82.7 (92.0)	100 (98.3)	82.7 (85.3)
Legendre Moments	100 (100)	97.3 (97.3)	100 (100)	97.3 (96.0)
OFMM	49.1 (99.4)	42.7 (94.6)	49.1 (98.8)	42.7 (94.6)
Binary Haar Features	98.8 (100)	90.7 (93.3)	98.8 (99.4)	90.7 (93.3)

Table 6.25: SVM parameters of SA search for Speed-Limit signs when C-SVM and ν -SVM are used.

Features	C-SVM		ν -SVM	
	C	γ	ν	γ
Binary Images	207.243	3.05e-005	0.09726	0.000258
Geometric Moments	1669	0.005202	0.33735	0.015625
Zernike Moments	1547	0.029662	0.37626	0.897143
Legendre Moments	1489	0.000246	0.15903	3.91e-005
OFMM	1133	5.36484	0.27698	8
Binary Haar Features	32768	0.013252	0.22334	0.061115

To demonstrate the effectiveness of simulated annealing algorithm, an experiment is carried out to measure the time of execution of both grid search and simulated annealing algorithms. Times of search for the parameters of the SVM using different features and normalised images are computed for both road sign rims and Speed-Limit signs. Tables 6.26 and 6.27 show a comparison of search times between that obtained by

grid search and simulated annealing. By the latter algorithm the semi-optimum values could be obtained in almost 20% of the time needed for grid search. Both types of SVM were tested in this experiment. The grid search and the simulated annealing are used to search for the best classification and then find the corresponding parameters. Once these parameters are computed, they can be saved in a file for the fast training of the SVM.

Table 6.26: A comparison of times of search of grid and SA for traffic sign rims.

Features	Search time for C-SVM		Search time for ν-SVM	
	Grid (sec.)	SA (sec.)	Grid (sec.)	SA (sec.)
Binary Images	438.0	62.4	295.0	55.6
Geometric Moments	37.6	8.0	26.1	5.4
Zernike Moments	37.6	8.4	26.1	5.4
Legendre Moments	36.3	5.9	22.5	4.9
OFFM	32.3	5.5	23.0	4.5
Binary Haar Features	26.8	4.6	19.9	5.1

Table 6.27: A comparison of times of search of grid and SA search for Speed-Limit.

Features	Search time for C-SVM		Search time for ν-SVM	
	Grid (sec.)	SA (sec.)	Grid (sec.)	SA (sec.)
Binary Images	329.0	48.5	211.0	40.3
Geometric Moments	28.5	4.6	15.2	3.4
Zernike Moments	22.7	3.5	13.2	3.1
Legendre Moments	26.0	3.8	15.6	3.4
OFFM	30.6	4.6	15.5	3.7
Binary Haar Features	24.9	3.5	14.5	3.3

6.4 Summary

In this chapter, the techniques used for traffic sign recognition are evaluated. The problem of traffic sign recognition was divided into three steps: colour segmentation, shape recognition and classification.

Different colour segmentation algorithms were tested by four different experiments which include the global performance of each algorithm, the specific tests of different image or sign conditions, the quality of segmentation, and processing times. The Shadow and Highlight invariant algorithm performed better than others. The algorithm was faster than the others and showed high robustness in different weather and lighting conditions.

Similar tests were used to evaluate the performance of the recognition algorithm. It is evaluated by a large set of images which reflect the overall performance of the recognition algorithm. It was also evaluated by another set of images taken under specific weather conditions. The algorithm was robust to a substantial variety of weather conditions, but there was a drop in performance in the case of occluded signs and fog conditions.

The last stage of this analysis was the evaluation of the performance of the SVM classifier. It is important to illustrate the conditions under which the classification takes place and which parameters affect the classification. In this analysis five types of tests are involved including classification with different features, classification with different SVM types and different kernels, classification with different moment orders, and searching for optimum parameters. Finally, a comparison between exhaustive search and simulated annealing search was presented which showed that using Simulated Annealing reduces the computational time down to 20% of the grid search without reducing the classification rate.

7. Conclusions

In this thesis a road and traffic sign recognition system which can help in creating a road sign inventory was developed, implemented and evaluated. This system, which involves a mixture of computer vision and pattern recognition problems, was able to extract road signs from still images of complex scenes subject to uncontrollable illumination. In the computer vision part, algorithms were developed to segment the image by using colours and to recognise the sign by colour-shape combinations as a priori knowledge. In the pattern recognition part, two SVM classifiers were invoked to put the unknown sign in one of the traffic sign categories depending on the sign rim and interior. This goal has now been reached and the system shows high robustness according to the experiments illustrated in sections 6.1, 6.2, and 6.3.

The following sections summarise the main findings and the contribution made by this research, which could be the beginning of a new approach to traffic sign recognition pointing to new directions for further research.

7.1 Collection of Traffic Sign Images

A library of images is created for the purpose of developing and testing the different algorithms of the traffic sign recognition system. The number of images collected in Sweden is 3415 images. Another 330 images are collected from different parts of Europe and some other countries such as Canada, the USA and Japan. Thousands of kilometres are driven in Sweden for this purpose. The majority of the images are collected in Dalarna, Stockholm, and southern parts of Sweden. All images are collected using one camera and they are all taken using one size: 640x480 pixels. The library is one of the biggest in Europe and is processed by Microsoft Access as shown in appendix E.

7.2 Colour Segmentation Algorithms

In this research, four colour segmentation algorithms are developed. They are as follows:

- Shadow and Highlight invariant colour segmentation algorithm,
- The dynamic threshold algorithm,
- A modification of de la Escalera's algorithm,
- The Fuzzy colour segmentation algorithm.

The experiments and tests carried out show that the Shadow and Highlight invariant algorithm is the best performer, compared with the other algorithms. The reasons behind this can be attributed to:

1. The high segmentation performance achieved by this algorithm which is 97.6% (Table 6.1),
2. The ability of the algorithm to achieve colour segmentation under a wide range of weather, sign, and image conditions (Table 6.2),
3. The highest segmentation speed among the other algorithms (Table 6.3) and
4. Its ability to enhance the recognition speed by suppressing noise and producing fewer objects in the segmented image (Table 6.4).

The performance of the developed algorithms, however, may be affected by several parameters, including weather conditions under which the image is taken, poor environmental lighting and the age of the sign under consideration.

7.3 Colour Segmentation in Poor Light Conditions

Colour segmentation in poor light conditions is a process which is specially developed by this research to eliminate the effect of fog and snowfall which often occur in many parts of Europe and more frequently in Scandinavia. These light conditions affect the general brightness of the image and hence affect the stability of the colours. In order to eliminate these effects, the colour segmentation algorithm is enhanced by invoking a special treatment of colour. The histogram equalisation technique is separately applied to the RGB channels of the captured image and the resultant RGB image is further treated by the colour constancy algorithm. Figure 6.2 shows the superiority of this algorithm in enhancing the image for colour segmentation. In this research, the user selects this algorithm manually when snowfall is noticed. In the future, however, an automated method can be adopted to switch to this algorithm when snow fall is detected.

This could be part of the future work in which metrological sensors can be attached to the computer to detect the snow fall.

7.4 Octagonality as a New Shape Measure

In order to specify the shape of any object in the segmented image, a specific criterion should be considered. As a traffic sign's shape can be a triangle, a circle, a rectangle or an octagon, four shape measures representing these shapes are invoked. Otagonality is introduced in section 5.6.1 and developed as a new measure in this research. Other shape measures include ellipticity, triangularity, and rectangularity. These shape measures give an idea of the shape of the object under consideration, and their range is [0,1] in which the value 1 means a perfect shape. They are easy to calculate since they use the Affine Moment Invariants. They are invariant to in-plane transformation, i.e. translation, rotation and scaling which suits the work with traffic sign recognition because images taken from different distances show the signs in different sizes. In addition, the sign could exist anywhere in the image and it could be in any orientation.

7.5 The Fuzzy Shape Recogniser

A fuzzy shape recogniser which relies on the four shape measures is developed during this research. A set of fuzzy rules are invoked to decide the shape of the sign. To recognise the right sign, simple IF-THEN rules are developed. These rules check the presence of a certain combination of information which includes the shape of the object under consideration, its rim colour, and its interior colour. Once a sign with these specifications is found it is sent to the classification stage, otherwise the object is deleted from the object list. This method prevents any unknown object with a specification different from traffic signs being sent to the classifier, and hence reduces the number of false alarms and the amount of calculations. The recogniser is tested in different test environments and it shows high robustness. Table 6.5 shows that the overall performance of the recogniser is 88.4%. It performs even better in some difficult environmental conditions such as snowfall, rainfall, dusk and dawn. The performance evaluation of this recogniser is illustrated in Table 6.5.

7.6 SVM for Traffic Sign Recognition

The SVM is a classifier which belongs to what is called linear classifiers. It separates data into two sets by using a hyperplane and makes the distance between them as large as possible. This classifier is invoked in this research to classify an unknown traffic sign in one of the traffic sign categories. Two stages of this classifier are invoked. The first stage classifies the sign rim and the second stage classifies the pictogram of the sign. In addition to normalised images, the classifier is trained and tested using other features. Among these features Legendre moments and Binary Haar invariants perform the best, bearing in mind that the road sign inventory can be handled during daytime and the system does not have to work during other times. The performance achieved by this classifier, which is about 97%, suites this kind of application.

Based on the preceding discussions, the road and traffic sign recognition system that can be used for automatic traffic sign inventory is presented. This system can assist local and national authorities with the task of maintaining and updating their road and traffic signs.

7.7 Future Directions

There are several avenues for further research which could follow from work begun here. They are given as follows:

7.7.1 Occluded Signs and Object Recognition

Signs are often occluded by obstacles and they are usually surrounded by many other objects. Figure 7.1 illustrates such a situation. This kind of situation often happens when images are taken from different angles. The main problem with these occlusions is that these signs are unpredictable and the shapes produced by them are also unpredictable. A number of researchers have already started to tackle this problem seriously [53], but the amount of work is still below what is necessary.

One approach to solve this problem is to use the type of features which are unaffected by clutter or partial occlusion. These features should also be invariant to in-plane transformations, 3D projective transformations, and common object variations.



Figure 7.1: Occlusions produced by the existence of obstacles.

Scale Invariant Feature Transform (SIFT) suggested by Lowe [115] is one kind of features which may fulfil these requirements. The image is transformed into a large collection of local feature vectors which are invariant to translation, scaling, and rotation, and partially invariant to illumination changes and affine or 3D projection. These features share a number of properties with the responses of neurons in Inferior Temporal (IT) cortex in primate vision. The scale-invariant features are identified by using a staged filtering approach. The first stage identifies key locations in scale space by looking for locations that are maxima or minima of a difference-of-Gaussian function. Each point is used to generate a feature vector that describes the local image region sampled relative to its scale-space coordinate frame. The features achieve partial invariance to local variations, such as affine or 3D projections, by blurring image gradient locations. The resulting feature vectors are called SIFT keys. The SIFT keys derived from an image are used in a nearest-neighbor approach to identify candidate object models. When at least 3 keys agree on the model parameters with low residual, there is strong evidence for the presence of the object. Since there may be dozens of SIFT keys in the image of a typical object, it is possible to have substantial levels of occlusion in the image and yet retain high levels of reliability.

7.7.2 Detachment of Signs

Another problem emerges when groups of signs share the same pole and are often attached to one another. When the image is colour segmented, the signs become attached to each other and hence connected component labelling creates a single object. This new

object does not belong to any of the expected sign shapes which the algorithm deals with. Of course, this problem affects the whole performance of the system. Figure 7.2 depicts samples of these images in which two signs are attached as one object. These signs should be detached from each other.



Figure 7.2: Traffic signs may be seen as one object after segmentation.

7.7.3 Similarity Measures for Sign Detection

Traffic sign detection is the most important stage in the whole process. It decides whether the sign can be classified or not. In this research a fuzzy shape recogniser is used to detect the sign. However, there are other methods which can be used at this stage. One is image similarity or Hausdorff distance which measures the similarity between two images. These methods can be used as part of the future work. Another factor, the aspect ratio of the object under consideration, can be used as an additional measure to detect the traffic signs. This is because all traffic signs have uniform shapes and are designed to have the same width and height. A combination of image similarity measures and shape measures can be a good solution to achieve better detection.

7.7.4 Real Time Applications

Another direction for further research is to develop a real time traffic sign recognition system which captures a video by a camera mounted on the vehicle, detects and recognises the traffic signs in real time and gives the result to the driver within a sufficient time frame in order to take the right action. The crucial issue in real time applications is the time spent to recognise the traffic sign. This should be reduced to the

minimum by choosing the proper techniques for real time applications and by optimising the code.

The methods presented in this thesis can be modified to fit the real time requirements. After detecting the border of the traffic sign and its interior, it can be tracked by a Kalman filter or by a suitable blob tracking algorithm which can be developed for this purpose. The main objective of this blob tracking algorithm is to minimize the search region from the whole image to an area which fits the traffic sign. Taking into consideration that the size of the traffic sign increases as the vehicle approaches the sign, the blob tracking algorithm should be able to match the traffic sign in the current frame with that in the next frame. The algorithm should be immune to the in-plane transformations. Tracking the traffic sign has an advantage that if the traffic sign is occluded in some frames or disappeared, it is still possible to follow that sign in the frames that follow.

If such a system is integrated with a GPS, it can be used to provide the driver with useful information about the actual speed limit on a certain road. By comparing the signed limit with the GPS speed reading, the driver can be warned when the speed limit is exceeded or when the driver does not stop before a STOP sign.

7.8 Final Remarks

The problem of traffic sign recognition for the purpose of road sign inventory has been approached by using colour and shape information of the traffic signs. A new set of algorithms, which has been developed and evaluated in a wide range of conditions, is exhibiting a good and robust performance.

The success of the proposed system opens new frontiers for further research in the future. Automation of road sign inventory is becoming a necessity for road authorities and such a system will be in use in the very near future.

References

- [1] US Department of Transportation, "Intelligent Transportation Systems, URL: http://www.its.dot.gov/its_overview.htm," 2006.
- [2] Swedish-Road-Administration, "URL: http://www.vv.se/templates/page3_15600.aspx," 2006.
- [3] C. Fang, C. Fuh, S. Chen, and P. Yen, "A road sign recognition system based on dynamic visual model," presented at The 2003 IEEE Computer Society Conf. Computer Vision and Pattern Recognition, Madison, Wisconsin, 2003.
- [4] C. Fang, S. Chen, and C. Fuh, "Road-sign detection and tracking," *IEEE Trans. on Vehicular Technology*, vol. 52, pp. 1329-1341, 2003.
- [5] N. Hoose, *Computer Image Processing in Traffic Engineering*. New York: John Wiley & sons Inc., 1991.
- [6] S. Vitabile and F. Sorbello, "Pictogram road signs detection and understanding in outdoor scenes," presented at Conf. Enhanced and Synthetic Vision, Orlando, Florida, 1998.
- [7] P. Parodi and G. Piccioli, "A feature-based recognition scheme for traffic scenes," presented at Intelligent Vehicles '95 Symposium, Detroit, USA, 1995.
- [8] J. Plane, *Traffic Engineering Handbook*: Prentice-Hall, 1992.
- [9] G. Jiang and T. Choi, "Robust detection of landmarks in color image based on fuzzy set theory," presented at Fourth Inter. Conf. on Signal Processing, Beijing, China, 1998.
- [10] M. Lalonde and Y. Li, "Road sign recognition. Technical report, Center de recherche informatique de Montréal, Survey of the state of Art for sub-Project 2.4, CRIM/IIT," 1995.
- [11] S. Vitabile, A. Gentile, and F. Sorbello, "A neural network based automatic road sign recognizer," presented at The 2002 Inter. Joint Conf. on Neural Networks, Honolulu, HI, USA, 2002.
- [12] P. Paclik, J. Novovicova, P. Pudil, and P. Somol, "Road sign classification using Laplace kernel classifier," *Pattern Recognition Letters*, vol. 21, pp. 1165-1173, 2000.

- [13] " http://www.vv.se/vag_traf/vagmarken/farglikare.htm."
- [14] P. Paclik, "ITS, Intelligent Transport System, <http://euler.fd.cvut.cz/research/rs2/articles/itsp.html>."
- [15] N. Kehtarnavaz and D. Kang, "Stop-sign recognition based on color/shape processing," *Machine Vision and Applications*, vol. 6, pp. 206-208, 1993.
- [16] A. de la Escalera, J. Armingol, and M. Mata, "Traffic sign recognition and analysis for intelligent vehicles," *Image and Vision Comput.*, vol. 21, pp. 247-258, 2003.
- [17] J. Miura, T. Kanda, and Y. Shirai, "An active vision system for real-time traffic sign recognition," presented at 2000 IEEE Intelligent Transportation Systems, Dearborn, MI, USA, 2000.
- [18] M. Blancard, "Road Sign Recognition: A study of Vision-based Decision Making for Road Environment Recognition," in *Vision-based Vehicle Guidance*, I. Masaki, Ed. Berlin, Germany: Springer-Verlag, 1992, pp. 162-172.
- [19] S. Vitabile, A. Gentile, G. Damnone, and F. Sorbello, "Multi-layer perceptron mapping on a SIMD architecture," presented at The 2002 IEEE Signal Processing Society Workshop, 2002.
- [20] S. Vitabile, G. Pollaccia, G. Pilato, and F. Sorbello, "Road sign Recognition using a dynamic pixel aggregation technique in the HSV color space," presented at 11th Inter. Conf. Image Analysis and Processing, Palermo, Italy, 2001.
- [21] S. Buluswar and B. Draper, "Color recognition in outdoor images," presented at Inter. Conf. Computer vision, Bombay, India, 1998.
- [22] R. Luo, H. Potlapalli, and D. Hislop, "Outdoor landmark recognition using fractal based vision and neural networks," presented at 1999 IEEE/RSJ Inter. Conf. Intelligent Robots and Systems, Yokohama, Japan, 1993.
- [23] P. Paclik and J. Novovicova, "Road sign classification without color information," presented at Sixth Annual Conf. of the Advanced School for Computing and Imaging, Lommel, Belgium, 2000.
- [24] E. Perez and B. Javidi, "Composite filter bank for road sign recognition," presented at 13th Annual Meeting IEEE Lasers and Electro-Optics Society, Rio Grande, Puerto Rico, 2000.

- [25] D. Kang, N. Griswold, and N. Kehtarnavaz, "An invariant traffic sign recognition system based on sequential color processing and geometrical transformation," presented at IEEE Southwest Symposium on Image Analysis and Interpretation, Dallas, Texas, USA, 1994.
- [26] D. Ghica, S. Lu, and X. Yuan, "Recognition of traffic signs by artificial neural network," presented at IEEE Inter. Conf. Neural Networks, Perth, W.A., 1995.
- [27] L. Estevez and N. Kehtarnavaz, "A real-time histogramic approach to road sign recognition," presented at IEEE Southwest Symposium on Image Analysis and Interpretation, San Antonio, Texas, 1996.
- [28] A. Yuille, D. Snow, and M. Nitzberg, "Signfinder, Using color to detect, localize and indentify informational," presented at Sixth Inter. Conf. on Computer Vision, Bombay, India, 1998.
- [29] N. Yabuki, Y. Matsuda, Y. Fukui, and S. Miki, "Region detection using color similarity," presented at 1999 IEEE Inter. Symposium on Circuits and Systems, Orlando, Florida, USA, 1999.
- [30] W. Shadeed, D. Abu-Al-Nadi, and M. Mismar, "Road traffic sign detection in color images," presented at 10th IEEE Inter. Conf. on Electronics, Circuits and Systems (ICECS 2003), Sharjah, United Arab Emirates, 2003.
- [31] M. Bénallal and J. Meunier, "Real-time color segmentation of road signs," presented at Canadian Conf. on Electrical and Computer Engineering (IEEE CCECE), Montréal, Canada, 2003.
- [32] G. Piccioli, E. De Micheli, P. Parodi, and M. Campani, "Robust road sign detection and recognition from image sequences," presented at Intelligent Vehicles Symposium, paris, France, 1994.
- [33] L. Priese, R. Lakmann, and V. Rehrmann, "Ideogram identification in a realtime traffic sign recognition system," presented at Intelligent Vehicle '95 Symposium, Detroit, USA, 1995.
- [34] Y. Aoyagi and T. Asakura, "A study on traffic sign recognition in scene image using genetic algorithms and neural networks," presented at The1996 IEEE IECON 22nd Inter. Conf. on Industrial Electronics, Control and Instrumentation, Taipei, Taiwan, 1996.

- [35] Y. Aoyagi and T. Asakura, "Detection and recognition of traffic sign in scene image using genetic algorithms and neural network," presented at The 35th SICE Annual Conference (SICE '96), Tottori, Japan, 1996.
- [36] G. Adorni, V. D'Andrea, G. Destri, and M. Mordonini, "Shape searching in real world images: a CNN-based approach," presented at Fourth IEEE Inter. Workshop on Cellular Neural Networks and Their Applications, Seville, Spain, 1996.
- [37] D. Gavrila, "Multi-feature Hierarchical template matching using distance transforms," presented at Fourteenth Inter. Conf. on Pattern Recognition, Brisbane, Qld Australia, 1998.
- [38] C. Schiekel, "A fast traffic sign recognition algorithm for gray value images," presented at 8th Inter. Conf. Computer Analysis of Images and Patterns, Ljubljana, Slovenia, 1999.
- [39] C. Huang and S. Hsu, "Road sign interpretation using matching pursuit method," presented at 4th IEEE Southwest Image Analysis and Interpretation Symposium, Austin, TX USA, 2000.
- [40] C. Huang and S. Hsu, "Road sign interpretation using matching pursuit method," presented at 15th Inter. Conf. on Pattern Recognition, Barcelona, Spain, 2000.
- [41] S. Hsu and C. Huang, "Road sign detection and recognition using matching pursuit method," *Image and Vision Comput.*, vol. 19, pp. 119-129, 2001.
- [42] E. Perez and B. Javidi, "Scale and Illumination-invariant road sign detection," presented at 13th Annual meeting IEEE Lasers and Electro-Optics Society, Rio Grande, Puerto Rico, 2000.
- [43] H. Sandoval, T. Hattor, S. Kitagawa, and Y. Chigusa, "Angle-dependent edge detection for traffic signs recognition," presented at IEEE Intelligent Vehicles Symposium 2000, Dearborn, MI, USA, 2000.
- [44] S. Puntavungkour, X. Chen, and M. Kusanagi, "Automatic Recognition and location of road signs from terrestrial color imagery," presented at Geoinformatics and DMGIS'2001, Bangkok, 2001.

- [45] K. Hirose, T. Asakura, and Y. Aoyagi, "Real-time recognition of road traffic sign in moving scene image using new image filter," presented at 26th Annual Conf. IEEE Industrial Electronics Society, Nagoya, Aichi, Japan, 2000.
- [46] H. Liu, D. Liu, and J. Xin, "Real-time recognition of road sign in motion image based on genetic algorithm," presented at 2002 Inter. Conf. Machine Learning and Cybernetics, Beijing, China, 2002.
- [47] H. Liu, D. Liu, and Q. Li, "Real-time recognition of road traffic sign in moving scene image using genetic algorithm," presented at 4th World Congress on Intelligent Control and Automation, Shanghai, China, 2002.
- [48] G. Loy and N. Barnes, "Fast shape-based sign detection for a drives assisatnce system," presented at IEEE/RSJ International Conference on Intelligent Robots and Systems, Sendai, Japan, 2004.
- [49] T. Hibi, "Vision based extraction and recognition of road sign region from natural color image, by using HSL and coordinates transformation," presented at 29th Inter. Symposium on Automotive Technology and Automation, Robotics, Motion and Machine Vision in the Automotive Industries, ISATA, 1996.
- [50] G. Piccioli, E. De Micheli, P. Parodi, and M. Campani, "Robust method for sign detection and recognition," *J. Image and Vision Computing*, vol. 14, pp. 209-223, 1996.
- [51] S. Azami, S. Katahara, and M. Aoki, "Route guidance sign identification using 2-D sturctural description," presented at The 1996 IEEE Intelligent Vehicle Symposium, Tokyo, Japan, 1996.
- [52] Y. Lauzière, D. Gingras, and F. Ferrie, "A model-based road sign identification system," presented at IEEE Computer Society Conf. Computer Vision and Pattern Recognition., Kauai, Hawaii, 2001.
- [53] A. de la Escalera, J. Armingol, and J. Pastor, "Visual sign information extraction and identification by deformable models for intelligent vehicles," *IEEE Trans. on Intelligent Transportation Systems*, vol. 5, pp. 57-68, 2004.
- [54] H. Ohara, I. Nishikawa, S. Miki, and N. Yabuki, "Detection and recognition of road signs using simple layered neural network," presented at The 9th Inter. Conf. Neural Information Processing, Singapore, 2002.

- [55] M. Shirvaikar, "Automatic detection and interpretation of road sign," presented at Thirty-sixth Southeastern Symposium on System Theory, Atlanta, USA, 2004.
- [56] M. Nakamura, S. Kodama, T. Jimbo, and M. Umeno, "Searching and recognition of road signpost using ring detection network," presented at 1999 IEEE Inter. Conf. on Systems, man, and Cybemetics (IEEE SMC '99), Tokyo, Japan, 1999.
- [57] D. Kellmeyer and H. Zwahlen, "Detection of highway warning signs in natural video images using color image processing and neural networks," presented at 1994 IEEE World Congress on Computational Intelligence, Orlando, Florida, USA, 1994.
- [58] R. Dahyot and P. Charbonnier, "Robust visual recognition of color images," presented at IEEE Conf. Computer Vision and Pattern Recognition, Hilton Head Island, SC, USA, 2000.
- [59] P. Paclik, J. Novovicova, and R. Duin, "Building Road-sign classification using a trainable similarity measure," *IEEE Trans. Intelligent Transportation*, vol. 7, pp. 309-321, 2006.
- [60] S. Lafuente-Arroyo, P. Gil-Jiménez, R. Maldonado-Bascón, F. López-Ferreras, and S. Maldonado-Bascón, "Traffic sign shape classification evaluation I: SVM using distance to borders," presented at IEEE Intelligent Vehicles Symposium, Las Vegas, USA, 2005.
- [61] G. Jiang, T. Choi, and Y. Zheng, "Morphological traffic sign recognition," presented at 3rd Inter. Conf. on Signal Processing, Beijing, China, 1996.
- [62] G. Jiang, Y. Zheng, and Y. Choi, "Morphological skeleton analysis of traffic signs on road," presented at IEEE Conf. Systems, Man, and Cybernetics, Beijing, China, 1996.
- [63] A. de la Escalera, L. Moreno, M. Salichs, and J. Armingol, "Road traffic sign detection and classification," *IEEE Trans. Industrial Electronics*, vol. 44, 1997.
- [64] R. Gonzalez and R. Woods, *Digital Image Processing*, Second ed. New Jersey: Prentice Hall, 2002.
- [65] W. Ritter, F. Stein, and R. Janssen, "Traffic sign recognition using color information," *Mathl. Comput. Modelling*, vol. 22, pp. 149-161, 1995.

- [66] H. Fleyeh, "Color detection and segmentation for road and traffic signs," presented at 2004 IEEE Conf. on Cybernetics and Intelligent Systems, Singapore, 2004.
- [67] R. Luo, H. Potlapalli, and D. Hislop, "Traffic sign recognition in outdoor environment using reconfigurable neural networks," presented at 1993 Inter. Joint Conf. on Neural Networks, Nagoya, Japan, 1993.
- [68] M. Betke and N. Makris, "Fast Object recognition in noisy images using simulated annealing," presented at Fifth Inter. Conf. on Computer vision, Cambridge, MA, USA, 1995.
- [69] S. Estable, J. Schick, F. Stein, R. Janssen, Y. Ott, W. Ritter, and J. Zheng, "A real-time traffic sign Recognition system," presented at Intelligent Vehicle '94 Symposium, Paris, France, 1994.
- [70] N. Kehtarnavaz and A. Ahmed, "Traffic sign recognition in noisy outdoor scene," presented at Intelligent Vehicle '95 Symposium, Detroit, USA, 1995.
- [71] M. Ebner, "A parallel algorithm for color constancy," *J. Parallel Distrib. Comput.*, vol. 64, pp. 79-88, 2004.
- [72] S. Buluswar and B. Draper, "Non-parametric classification of pixels under varying outdoor illumination," presented at ARPA Image Understanding Workshop, 1994.
- [73] CIE, *International Lighting Vocabulary*, 4th ed: CIE Publications 17.5, Commission International de L'Eclairage, 1989.
- [74] M. Sridharan and P. Stone, "Towards on-board color constancy on mobile robots," presented at First Canadian Conf. Computer and Robot Vision (CRV'04), Ontario, Canada, 2004.
- [75] B. Funt, K. Barnard, and L. Martin, "Is machine colour constancy good enough?," presented at Fifth European Conf. on Computer Vision (ECCV'98), Freiburg, Germany, 1998.
- [76] G. Finlayson, "Computational colour constancy," presented at 15th Inter. Conf. on Pattern Recognition, Barcelona, Spain, 2000.

- [77] Y. Tsin, R. Collins, V. Ramesh, and T. Kanade, "Bayesian color constancy for outdoor object recognition," presented at The 2001 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR 2001), Kauai, Hawaii, 2001.
- [78] C. Rosenberg, M. Hebert, and S. Thrun, "Color constancy using KL-Divergence," presented at Eight IEEE Inter. Conf. Computer Vision (ICCV 2001), Vancouver, BC Canada, 2001.
- [79] L. Török and Á. Zarányi, "CNN based color constancy algorithm," presented at 7th IEEE Inter. Workshop on Cellular Neural Networks and Their Applications (CNNA 2002), Frankfurt, Germany, 2002.
- [80] M. Ebner, "Color constancy using local color shifts," presented at The 8th European Conf. on Computer Vision, Prague, Czech Republic, 2004.
- [81] R. Stanikunas, H. Vaitkevicius, and J. Kulikowski, "Investigation of color constancy with a neural network," *Neural Networks*, vol. 17, pp. 327-337, 2004.
- [82] M. Teague, "Image Analysis via the general theory of moments," *J. Opt. Soc. Am.*, vol. 70, pp. 920-930, 1980.
- [83] R. Mukundan and K. Ramakrishnan, *Moment functions in image analysis theory and applications*. Singapore: World Scientific, 1998.
- [84] C. Chong, P. Raveendran, and R. Mukundan, "Translation invariants of Zernike moments," *Pattern Recog.*, vol. 36, pp. 1765-1773, 2003.
- [85] N. Kamila, S. mahapatra, and S. Nanda, "Invariance image analysis using modified Zernike moments," *Pattern Recognition Letters*, vol. 26, pp. 747-753, 2005.
- [86] A. Khotanzad and Y. Hong, "Rotation invariant pattern recognition using Zernike moments," presented at 9th International Conf. on Pattern Recognition, Rome, Italy, 1988.
- [87] S. Belkasim, M. Ahmadi, and M. Shridhar, "An efficient algorithm for fast computation or Zernike moments," presented at IEEE 39th Midwest Symposium on Circuits and Systems, 1996.
- [88] C. Chong, P. Raveendran, and R. Mukundan, "A comparative analysis of algorithms for fast computation of Zernike moments," *Pattern Recog.*, vol. 36, pp. 731-742, 2003.

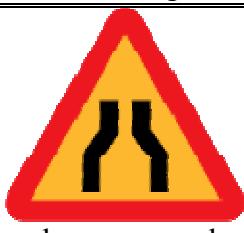
- [89] C. Kan and M. Srinath, "Invariant character recognition with Zernike and orthogonal Fourier-Mellin moments," *Pattern Recog.*, vol. 35, pp. 143-154, 2002.
- [90] P. Yap and R. Paramesran, "An efficient method for the computation of Legendre moments," *IEEE Trans. Pattern Anal. Mach Intell.*, vol. 27, pp. 1996-2002, 2005.
- [91] C. Chong, P. Raveendran, and R. Mukundan, "Translation and scale invariants of Legendre moments," *Pattern Recog.*, vol. 37, pp. 119-129, 2004.
- [92] Y. Sheng and H. Arsenault, "Experiments on pattern recognition using invariant Fourier-Mellin descriptors," *J. Opt. Soc. Am.*, vol. 3, pp. 771-776, 1986.
- [93] Y. Sheng and L. Shen, "Orthogonal Fourier-Mellin moments for invariant pattern recognition," *J. Opt. Soc. Am.*, vol. 11, pp. 1748-1757, 1994.
- [94] H. Schulz-Mirbach, "Invariant features for gray scale images," presented at 17 DAGM - Symposium "Mustererkennung", Bielefeld, 1995.
- [95] C. Cortes and V. Vapnik, "Support vector networks," *Machine Learning*, vol. 20, pp. 273-297, 1995.
- [96] N. Cristianini and J. Shawe-Taylor, *An Introduction to support vector machine*: Cambridge University Press, 2000.
- [97] M. Sewell, "Support Vector Machines (SVM), Karush-Kuhn-Tucker conditions. <http://www.svms.org/kkt/>"
- [98] J. Platt, "Sequential minimal optimization: A fast algorithm for training support vector machines. Technical Report MSR-TR-98-14," 1998.
- [99] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*: Cambridge University press, 2004.
- [100] V. Vapnik, *Statistical learning theory*: Wiley, 1998.
- [101] B. Schölkopf, A. Smola, R. Williamson, and P. Bartlett, "New support vector algorithms," *Neural Computation*, vol. 12, pp. 1083-1121, 2000.
- [102] C. Chang and C. Lin, "Training nu-Support vector classifiers: Theory and algorithms," *Neural Computation*, vol. 13, pp. 2119-2147, 2001.
- [103] P. Chen, C. Lin, and B. Schölkopf, "A tutorial on nu-support vector machines. Technical report, National Taiwan University, <http://www.csie.ntu.edu.tw/~cjlin/papers/nusvmtutorial.pdf>."
- [104] <http://www.cs.cmu.edu/afs/cs/project/cil/ftp/html/v-images.html>.

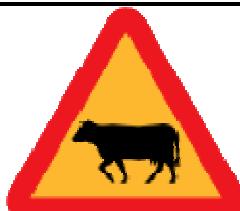
- [105] Swedish-Road-Administration, "http://www.vv.se/vag_traf/vagmarken/farglikare.htm," 2004.
- [106] H. Fleyeh, "Road and Traffic Sign Color Detection and Segmentation - A Fuzzy Approach," presented at Machine Vision Applications (MVA2005 IAPR), Tsukuba-Japan, 2005.
- [107] T. Gevers and A. Smeulders, "Color-based object recognition," *Pattern Recog.*, vol. 32, pp. 453-464, 1999.
- [108] H. Fleyeh, "Shadow And Highlight Invariant Colour Segmentation Algorithm For Traffic Signs," presented at 2006 IEEE Conf. on Cybernetics and Intelligent Systems, Bangkok, Thailand, 2006.
- [109] H. Fleyeh, "Traffic Signs Color Detection and Segmentation in Poor Light Conditions," presented at Machine Vision Applications (MVA2005 IAPR), Tsukuba-Japan, 2005.
- [110] J. Flusser and T. Suk, "Pattern recognition by affine moment invariants," *Pattern Recognition*, vol. 26, pp. 167-174, 1993.
- [111] P. Rosin, "Measuring shape: ellipticity, rectangularity, and triangularity," *Machine Vision and Applications*, vol. 14, pp. 172-184, 2003.
- [112] R. Graham and F. Yao, "Finding the convex hull of a simple polygon," *Journal of Algorithms*, vol. 4, pp. 324-331, 1983.
- [113] H. Fleyeh, "A novel fuzzy approach for shape determination of traffic signs," presented at 2nd Indian international conference on Artificial Intelligence (IICAI-05), Pune, India, 2005.
- [114] K. Suzuki, I. Horiba, and N. Sugie, "Linear-time connected component labelling based on sequential local operations," *Computer Vision and Image Understanding*, vol. 89, pp. 1-23, 2003.
- [115] D. Lowe, "Object Recognition from Local Scale-Invariant Features," presented at The International Conference on Computer Vision, Corfu, Greece, 1999.
- [116] http://www.ncsu.edu/scivis/lessons/colormodels/color_models2.html.
- [117] R. Hunt, *The reproduction of colour*, Sixth ed. West Sussex: John Wiley & Sons Ltd, 2004.

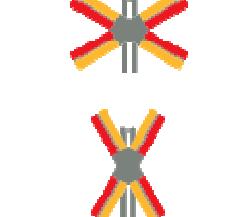
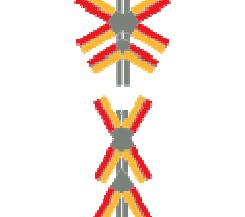
- [118] P. Colantoni and AI, "Color Space Transformations, URL: <http://colantoni.nerim.net/download/colorspacetransform-1.0.pdf>," 2004.
- [119] C. Poynton, "Frequently asked questions about color, URL: <http://www.poynton.com/PDFs/ColorFAQ.pdf>," 1999.
- [120] D. Hearn and M. Baker, *Computer Graphics with OpenGL*, Third ed. New Jersey: Prentice Hall, 2004.
- [121] http://www.cs.rit.edu/~ncs/color/t_convert.html.
- [122] <http://semmix.pl/color/index.html>.
- [123] J. Foley, A. v. Dam, S. Feiner, and J. Hughes, *Computer Graphics Principles and Practice*, Second ed. New York: Addison-Wesley Publishing company, Inc., 1996.
- [124] R. Castleman, *Digital Image Processing*: Prentice Hall, 1996.
- [125] <http://www.nationmaster.com/encyclopedia/HLS-color-space>.
- [126] A. Hanbury and J. Serra, "A 3D-polar coordinate colour representation suitable for image analysis," *Computer Vision and Image Understanding*, 2002.
- [127] J. Angulo and J. Serra, "Color segmentation by ordered mergings," presented at Int. Conf. on Image Processing, Barcelona, Spain, 2003.
- [128] <http://astronomy.swin.edu.au/~pbourke/colour/ycc/>.

Appendix A - The Swedish Road and Traffic Signs

Warning Signs

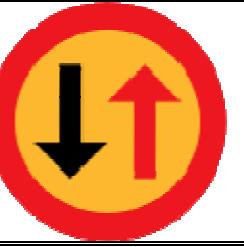
			
Dangerous bend, bend to right	Dangerous bend, bend to left	Dangerous bends, first bend to right	Dangerous bends, first bend to left
			
Steep hill downwards	Steep hill upwards	Road narrows on both sides	Road narrows from right side
			
Road narrows from left side	Opening or swing bridge	Quayside or ferry berth	Uneven road
			
Road works	Slippery road	Falling rocks from right	Falling rocks from left
			
Loose chippings	Junction	Roundabout	Junction with a road

			the users of which must give way, 90° left, 90° right
			
Junction with a road the users of which must give way, 45° left, 45° right	Junction with a road the users of which must give way, 90° left, 45° right	Junction with a road the users of which must give way, 45° left, 90° right	Junction with a road the users of which must give way, 90° left
			Traffic signals
Level crossing with gates	Level crossing without gates	Intersection with tramway line	Pedestrian crossing
			
Children	Animals (elk)	Animals (deer)	Animals (cow)
			Cyclists and moped riders on carriageway
Animals (horse)	Animals (reindeer)	Animals (sheep)	Cyclists and moped riders on carriageway

			
			
	 Distance to level crossing	 Single track level crossing	 Multitrack level crossing

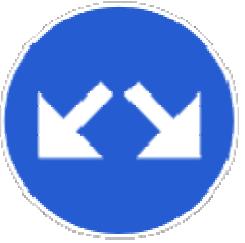
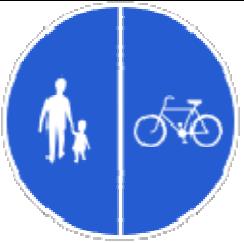
Prohibitory Signs

			
Closed to all vehicles in both directions	No entry	No power-driven vehicles	No power-driven vehicles except motorcycles without side cars
			
No motorcycles	No power driven vehicles drawing a trailer other than a semi trailer or a single axle trailer	No lorries	No tractors, construction vehicles etc.
			
No cycles or mopeds	No mopeds	No animal-drawn vehicles	No riding
			
No pedestrians	No off-road vehicles	No vehicles having an overall width exceeding ... meters	No vehicles having an overall height exceeding ... meters

			
No vehicles exceeding ... tons laden weight	No vehicles having a weight exceeding ... tons on one axle	Minimum distance between power driven vehicles	No vehicles or combination of vehicles exceeding ... meters
			
No vehicles or combination of vehicles exceeding ... tons laden weight or bearing capacity class	No vehicles having a weight exceeding ... tons on a tandem axle	passing without stopping prohibited at customs	Priority for oncoming vehicles
			
No right turn	No left turn	No U-turns	No overtaking
			
No overtaking by lorries	Maximum speed (30 km/h)	Maximum speed (50 km/h)	Maximum speed (70 km/h)
			

Maximum speed (90 km/h)	Maximum speed (110 km/h)	No vehicles carrying dangerous goods	Priority for oncoming vehicles

Mandatory Signs

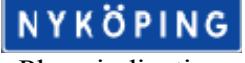
			
Pass this side. Right	Pass this side. Left	Pass this side. Either side	Roundabout
			
Foot path	Track for cycles and mopeds	Compulsory track for pedestrians, cyclists and moped drivers.	Compulsory track for pedestrians, cyclists and moped drivers. Dual track
			
Compulsory track for pedestrians, cyclists and moped drivers. Dual track	Track for rider on a horseback	Track for off-road vehicles	Beginning of lane reserved for scheduled buses etc.
			
End lane reserved for scheduled buses etc.			

Informative Signs

			
Post office	Telephone	Radio station for road and traffic information	Information
			
Workshop	Petrol station	Refreshments	Restaurant
			
Hotel	Picnic site	Toilet	First aid
			
Industrial zone	Youth hostel	Holiday chalets	Caravan site
			
Camping site	Bathing	National heritage	Open-air recreation
			
Hiking trail	Chair lift	Tow lift	Fishing licences on sale here

			
Golf course	Lorry	Car	Bus
			
Airfield	Airfield straight ahead	Ferry	Taxi rink
			
Low-speed road	End of low-speed road	Residential area	End of residential area
			
Pedestrian area	End of pedestrian area	Priority over oncoming vehicles	No through road
			
Lay-by or passing place	Parking	Park and ride	Pedestrian crossing
			
Motorway	End of motorway	Expressway	End of expressway

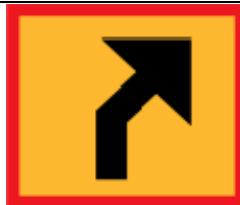
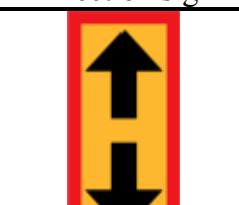
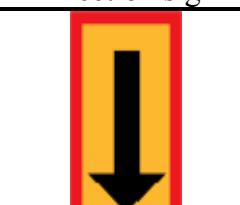
Road number sign European highway	Main highways (other than European highways) numbered 1-499	Road number sign. Direction to a numbered road	Road number sign for traffic diversion
Built-up area	End of built-up area	Maximum recommended speed	End of maximum recommended speed
Toll road	Temporary sign, left most lane ends	Temporary sign, Lane merges with oncoming traffic	Temporary sign, oncoming traffic merges with this lane
Priority road	End of priority road	One-way traffic	Advance direction sign, diagrammatic type
Advance direction sign, stack type	Lane pre-selection sign	Advance direction sign diagrammatic indicating prohibition of left turning	Direction sign flag type

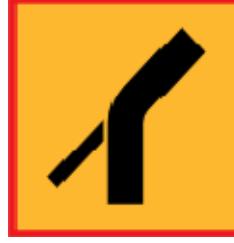
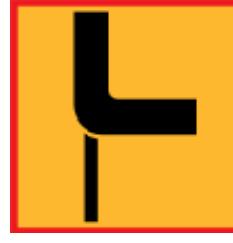
 Direction to motorway or expressway	 Sign to specific district or area	 Sign to place reached by a private road	 Sign to local amenities
 Lane assignment type	 Advance direction sign exit ahead from motorway or expressway	 Advance direction sign exit ahead from other road than motorway or expressway	 Direction sign exit sign
 Place indication sign	 Confirmatory sign	 Collection sign	 Interchange number
 Recommended route for vehicles carrying dangerous goods	 Tourist route	 Tourist attraction area	 Landmark
 Recommended route for pedal cycles and mopeds	 Recommended route for pedestrians	 Disabled persons	 Stack type design
 Flag type sign	 Place indication sign	 Confirmatory sign	 Bicycle sign

			The long turn — mandatory turning manoeuvre for pedal cycles and mopeds
			 Sign to temporary event
			

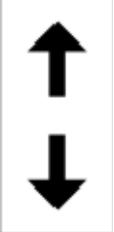
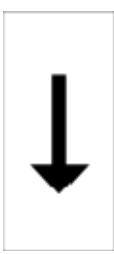
Supplementary Signs

			
Symbol plate for specified vehicle or road user category (lorry)	Symbol plate for specified vehicle or road user category (car)	Symbol plate for specified vehicle or road user category (bus)	Symbol plate for specified vehicle or road user category (caravan)
		Forestry vehicle crossing ahead	
Symbol plate for specified vehicle or road user category (car + caravan)	Soft shoulder	Forestry vehicle crossing ahead	High tension cable
			
Blind persons crossing or in the vicinity of the road	Deaf persons crossing or in the vicinity of the road	All way stop	

	vicinity of the road		
			
Direction sign	Direction sign	Direction sign	Direction sign
			Prohibition effective in the direction of arrow
Direction sign	Prohibition effective in both directions of the sign	Prohibition end at sign	
STOPP 200 m			
Stop and give way at specified distance ahead	Lateral clearance	Total weight	Times the restriction applies.
100 m			
Distance to	No parking between times indicated	No parking between times indicated (with weekday given)	
			
Direction of priority road at intersect	Direction of priority road at intersect	Direction of priority road at intersect	Direction of priority road at intersect

			
Direction of priority road at intersect	Direction of priority road at intersect	Direction of priority road at intersect	Direction of priority road at intersect
			
Direction of priority road at intersect	Direction of priority road at intersect	Direction of priority road at intersect	Direction of priority road at intersect
			
Direction of priority road at intersect	Direction of priority road at intersect	Direction of priority road at intersect	
			
Parking configuration	Parking configuration	Parking configuration	Parking configuration
			
Symbol plate for specified vehicle or road user category (lorry)	Symbol plate for specified vehicle or road user category (handicapped)	Symbol plate for specified vehicle or road user category (car)	Symbol plate for specified vehicle or road user category (bus)

			
Symbol plate for specified vehicle or road user category (motorcycle)	Symbol plate for specified vehicle or road user category (bike)	Symbol plate for specified vehicle or road user category (caravan)	Symbol plate for specified vehicle or road user category (car + caravan)
			
Parking permitted for specified period between times shown	Two-way traffic on cycle and moped tracks		
			
Direction sign	Direction sign	Direction sign	Direction sign
			
Direction sign	Direction sign	Direction sign	
			
Direction sign	Direction sign	Direction sign	
			
Direction sign	Direction sign	Direction sign	

P-hus Parking house	 Up and down arrows indicating parking is effective in both directions.	 A single downward-pointing arrow indicating parking ends at the sign.	 A single left-pointing arrow indicating parking is effective in the direction of the arrow.
	Parking, effective in both directions of the sign	Parking ends at sign	Parking, effective in direction of arrow

Appendix B - Colour Physics and Colour Spaces

B.1 Introduction

Visible light, ultraviolet light, infrared light, x-rays, TV and radio waves, etc are all forms of **electromagnetic energy** which travels in waves. These different types of waves consist of what is called the electromagnetic spectrum shown in Figure B.1.

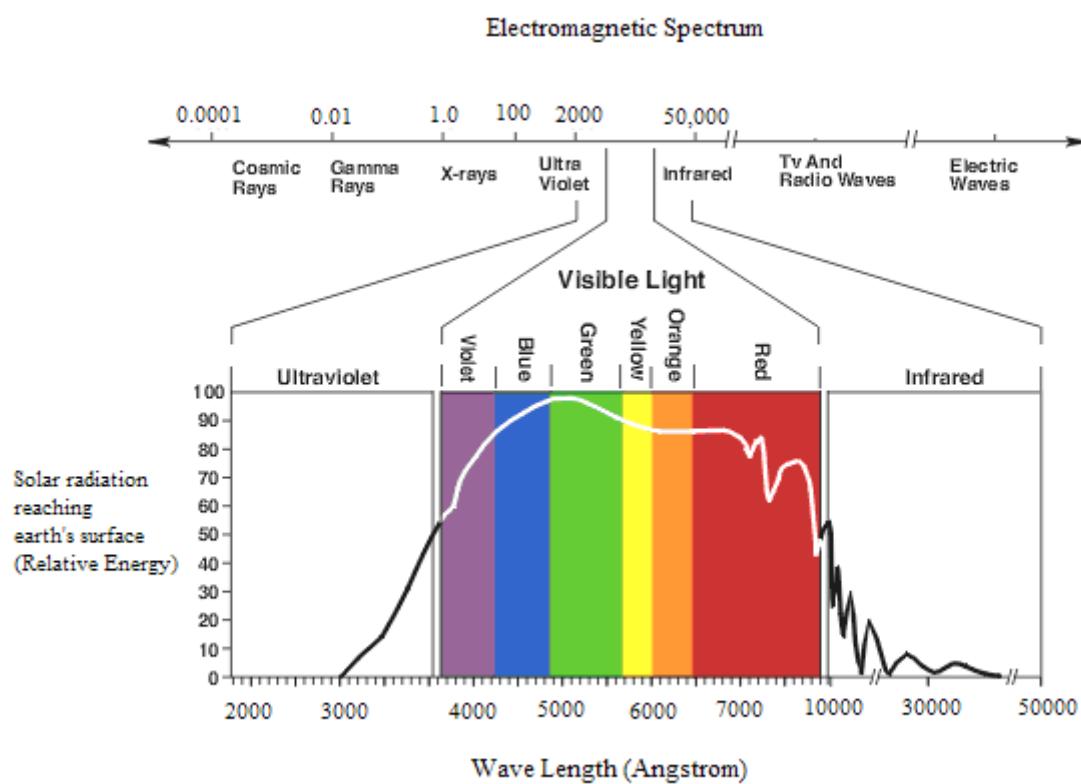


Figure B.1: The electromagnetic spectrum [116].

There is a narrow range of this electromagnetic energy from the sun and other light sources which creates wavelengths visible to humans. Each of these wavelengths, lying between 4000 - 7000 Angstroms, is associated with a particular colour response. This means that colour can be defined as this part in the visible region of the spectrum having wavelengths in the region of 400-700 nm. For example, the wavelengths near 4000

Angstroms (400 nm) are violet in colour while those near 7000 (700 nm) are red. Figure B.2 shows the visible light of this electromagnetic spectrum [116].

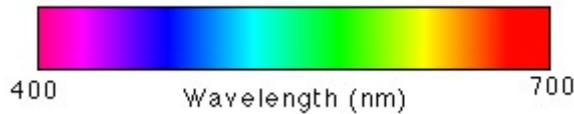


Figure B.2: The colours of the wavelengths of visible light.

B.2 Model of the Eye

Light which has a dominant frequency or set of frequencies is called *chromatic*, and that without such dominant frequencies (white light) is called *achromatic*. Our eyes have two different types of receptors: cones and rods. The cones are sensitive to chromatic light and the rods are sensitive only to achromatic light. That is, our rods can tell that there is light and they are very sensitive to it, but cannot tell what colour it is. Cones can tell what colour the light is, but it appears that they are not very sensitive to dim light. There are three types of cone cells in the human retina, each of which responds to incident radiation with different spectral response curves.

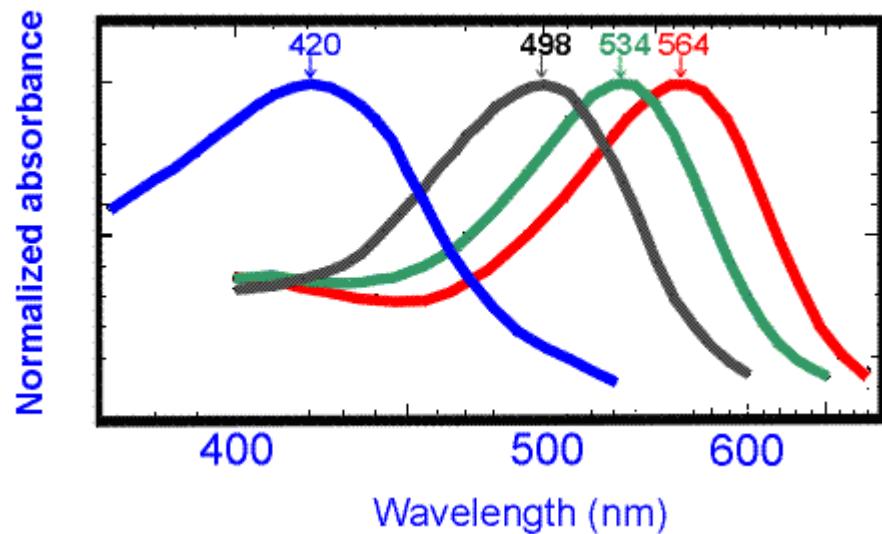


Figure B.3: Spectral Sensitivity Curves of Rods & Cones.

The peak sensitivities of light frequencies are called red (564 nm), green (534 nm) and blue (420 nm), Figure B.3. Light, at any wavelength in the visual spectrum range from 400-700 nm will excite one or more of these three types of sensors. Our perception of which colour we are seeing is determined by which combination of sensors are excited and by how much.

The sensitivity curves of the ρ (for Red), γ (for Green), and β (for Blue) cones in our eyes determine the intensity of the colours we perceive for each of the wavelengths in the visual spectrum. Figure B.4 shows an approximation of the visual spectrum illustration adjusted for the sensitivity curves of ρ , γ and β sensors.

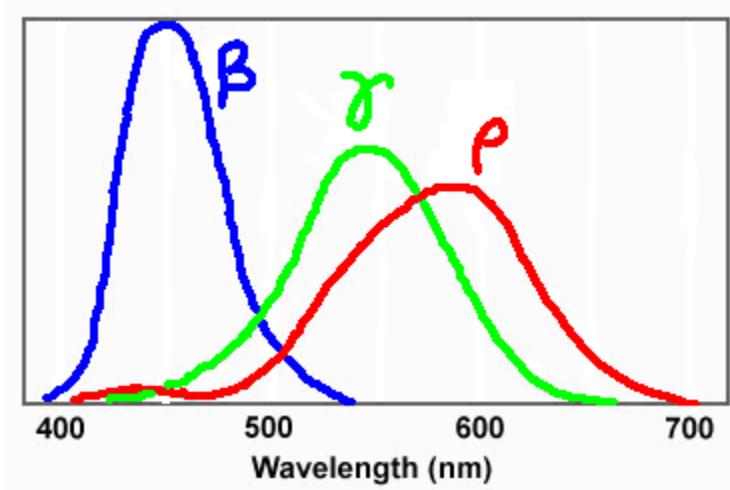


Figure B.4: Sensitivity Curves of Red, Green & Blue.

The general sensitivity of the eye is not identical in the whole thread of the visible radiation in the consideration of wavelength. Extreme sensitivity is placed in the green part, while the receptors of the blue colour are the least sensitive [117].

B.3 Colour Spaces

B.3.1 Device dependent and device independent Colour Space

A device dependent colour space is a colour space where the resultant colour depends on the equipment and the set-up used to produce it. An RGB = (250, 134, 67) will be altered as the brightness and contrast of the device are changed by the user. If the

red, green and blue phosphor of the monitor is slightly changed, the characteristics of the colours produced by this monitor will change. In this case, RGB is a colour space which depends on the device; it is *device dependent*.

A *device independent* colour space is the one where the coordinates used to produce the colour will produce the same colour wherever they are applied. The CIE L*a*b* colour space is an example of the device independent colour spaces [118].

B.3.2 Colour Gamut

A colour gamut is the area enclosed by a colour space in three dimensions. The usual way to represent colour gamut of a colour reproduction system is to show a range of colours available in a device independent colour space. Colour gamut is often represented in two dimensions [118].

B.3.3 Terminology

The International Commission on Illumination (CIE) (Commission Internationale de l'Eclairage) defined some terms which are used through this document. These definitions are as follows [119]:

Colour: is the perceptual result of light in the visible region of the spectrum, having wavelengths in the region of 400 – 700 nm, incident upon the retina.

Intensity: is a measure over some interval of the electromagnetic spectrum of the flow of power that is radiated from, or incident on, a surface. It is measured in units such as watt per square meter.

Brightness: the attribute of a visual sensation according to which an area appears to emit more or less light. This attribute is measured subjectively and has no unit of measurement.

Luminance: Luminance is the luminous intensity per unit surface area, measured in the SI units of candela per square metre (cd/m^2). Luminous intensity (unit: Candela) is radiant intensity (unit: watts/steradian) weighted by the spectral response of the human eye. The luminance measure therefore takes into account that for the three light sources which appear red, green, and blue, and have the same radiant intensity in the visible spectrum, the green one will appear the brightest, and the blue one is the dimmest.

Lumen: a unit of measure for light radiation per unit solid angle from a “standard” point light source.

Lightness: A measure which takes into account the non-linear response of the human eye to luminance. A source having a luminance of only 18% of a reference luminance appears about half as bright. The CIE uses lightness in their L*a*b and L*u*v* spaces.

Colour temperature: The temperature of the Planckian radiator whose radiation has the same chromaticity as that of a given stimulus.

Achromatic colour: Colour devoid of hue (white, black, grey, neutral).

Candela, cd: The unit of luminous intensity. The candela is the luminous intensity, in a given direction, of a source emitting a monochromatic radiation of frequency 540×10^{12} Hz, the radiant intensity of which in that direction is equal to 1/683 Watt per steradian.

Chromatic colour: Colour exhibiting hue, as opposed to achromatic colour.

Hue: according to the CIE, hue is the attribute of a visual sensation according to which an area appears to be similar to one of the perceived colours, red, yellow, green and blue, or a combination of two of them. Hue is more specifically described by the dominant wavelength in models such as the CIE system.

Saturation: refers to the dominance of hue in the colour. On the outer edge of the hue wheel are the ‘pure’ hues. Moving towards the centre of the wheel, the hues are used to describe decreasing /receding colour dominance.

B.3.4 CIE XYZ Colour Space

In 1931, the Commission Internationale de l’Eclairage (CIE) developed a method to systematically measure colours in relation to the wavelength they contain. This model is called the CIE XYZ model, in which three primaries are defined. They are called X, Y, and Z, and can be combined to generate any colour which humans can see. This means that any colour $C(\lambda)$ can be represented as:

$$C(\lambda) = (X, Y, Z) \quad (\text{B.1})$$

The X, Y, and Z are calculated by using the colour-matching functions shown in Figure B.5 as:

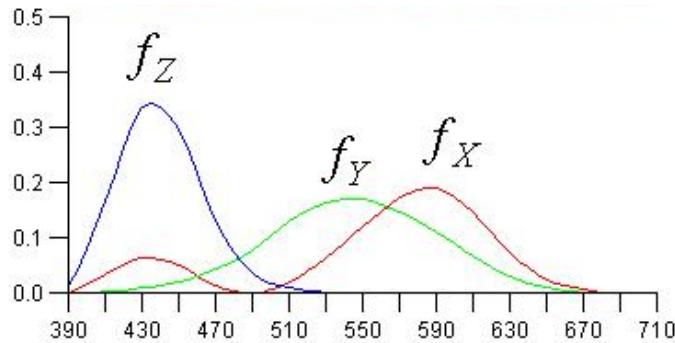


Figure B.5: The colour matching functions.

$$X = k \int_{\text{visible } \lambda} f_X(\lambda) I(\lambda) d\lambda \quad (\text{B.2})$$

$$Y = k \int_{\text{visible } \lambda} f_Y(\lambda) I(\lambda) d\lambda \quad (\text{B.3})$$

$$Z = k \int_{\text{visible } \lambda} f_Z(\lambda) I(\lambda) d\lambda \quad (\text{B.4})$$

where k is a normalisation coefficient (it has the value 683 lumens/watt), the function $I(\lambda)$ is the spectral radiance, and $f_Y(\lambda)$ is the colour matching function, which is chosen so that the parameter Y is the luminance of the colour [120]. This is different from brightness as defined in the terminology section. The original CIE colour model is a 3-D model. The Z axis is projected on the XY -plane to give the 2-D model; this is shown in Figure B.6.

The CIE chart represents hue and saturation on a two dimensional chart. Fully saturated hues lie along the outside edge with desaturated colours toward the centre of the chart. This CIE X - Y chart was later revised to better reflect human perception of colour. While the CIE chart is a very useful tool, the colour space it defines does not provide an intuitive model for our colour vision system or the devices we use to reproduce colour.

The Y primary was defined to match the luminous-efficiency function of the human eye. X and Z were obtained based on experiments involving human observers. The chromaticity values are defined as:

$$x = X / (X + Y + Z) \quad (\text{B.5})$$

$$y = Y / (X + Y + Z) \quad (\text{B.6})$$

$$z = Z / (X + Y + Z) \quad (\text{B.7})$$

Knowing x and y, z can be found as $z = 1 - x - y$.

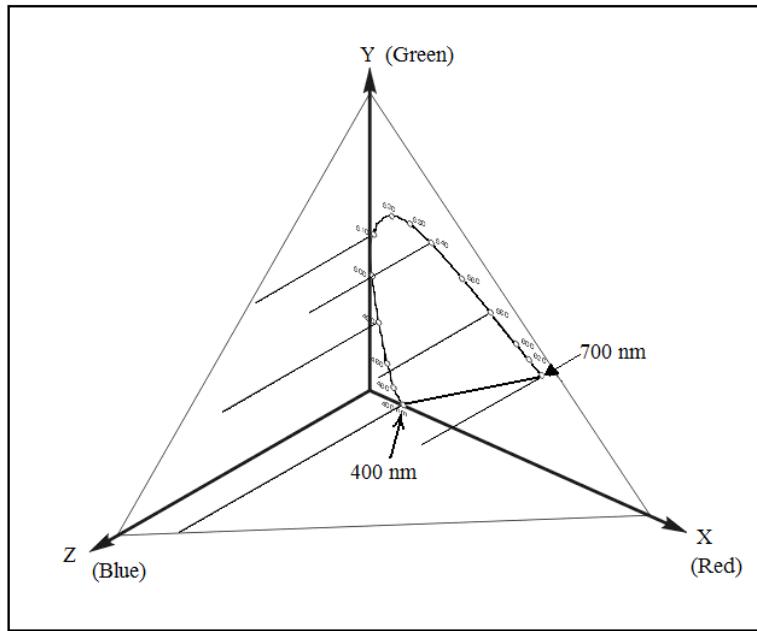


Figure B.6: The Three-Dimensional model of the CIE-XYZ colour space.

The CIE Chromaticity diagram which is shown in Figure B.7 is a plot of x versus y for all visible colours. Each point on the edge denotes a pure colour of a specific wavelength. White is at the centre where all colours combine equally ($x = y = z = 1/3$). RGB values in a particular set of primaries can be transformed from and to CIE XYZ colour space by using matrices. To transform the XYZ colour space to RGB with D65 white point, the following equation is used [121]:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 3.240479 & -1.537150 & -0.498535 \\ -0.969256 & 1.875992 & 0.041556 \\ 0.055648 & -0.204043 & 1.057311 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (\text{B.8})$$

The range of the RGB generated by equation (B.8) is [0,1]. Negative values in this matrix mean that it is not possible to generate all of the visible colours using the RGB colour space.

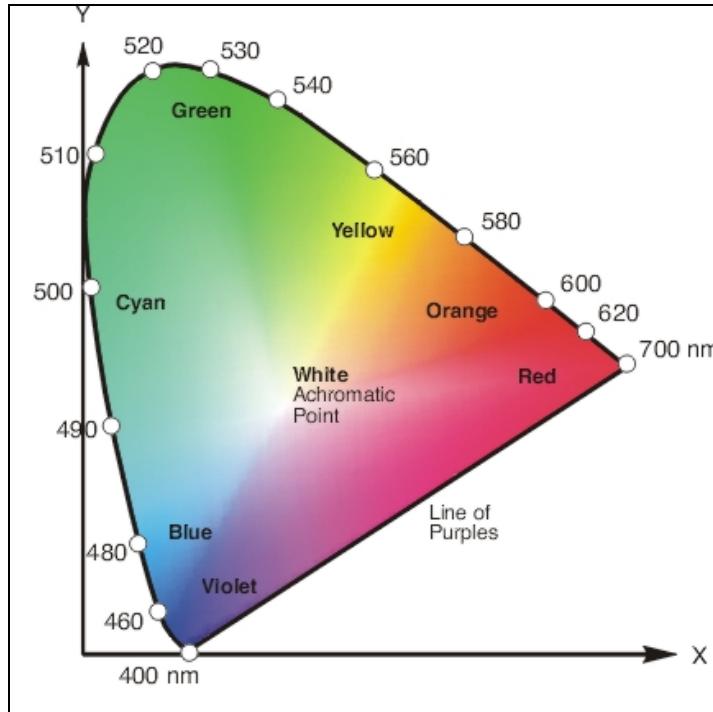


Figure B.7: The CIE diagram.

On the other hand, given the RGB values [0,1], the XYZ values can be computed by inverting equation (B.8) as follows:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (\text{B.9})$$

B.3.5 The RGB Colour Space

This colour space is used for computer graphics. It is the best-known and most widely used colour space. Each colour in this system is represented by three values referred to as R (red), G (green), and B (blue). It is built in the form of a cube in the Cartesian coordinate system in which the x, y, and z axis are represented by R, G, and B respectively, Figure B.8. The range of the RGB values are [0,1]; which leads to black, which is located at the centre of the coordinates, being given as (0,0,0) and white as (1,1,1). These two colours (the black and white) represent the opposite corners of RGB

space cube. Vertices of the cube on the axes represent the primary colours, and the remaining vertices represent the complementary colour for each of the primary colours. The other corners of the cube are red, green, blue, cyan, magenta, and yellow [120].

The grey scale is located on the diagonal of the cube between the black and the white. This space has many drawbacks such as:

- It is difficult to do any segmentation because of the 3-D nature of the system.
- The coordinates of the three colours are highly correlated, and as a result any variation in the ambient light intensity affects the RGB system by shifting the cluster of colours towards the white or the black corners. As a result, it will be hard to recognise the object under consideration under different brightness conditions [10].

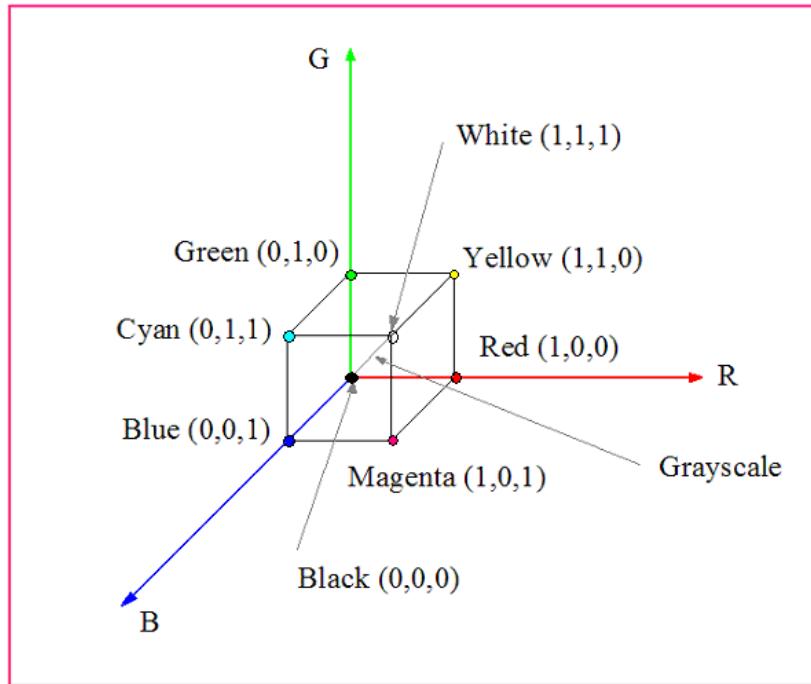


Figure B.8: The RGB colour space.

The RGB colour scheme is an additive model. Intensities of the primary colours are added to produce other colours. Each colour point within the bounds of the cube can be represented as the triple (R, G, B) , where values for R, G, and B are assigned in the range from 0 to 1. The magenta vertex is obtained by adding red and blue to produce the triple $(1, 0, 1)$, and white at $(1, 1, 1)$ is the sum of the red, green, and blue vertices. Shades of

grey are represented along the main diagonal of the cube from the origin (black) to the white vertex.

B.3.6 The CMY and CMYK Colour Space

This system is based on the secondary colours (cyan, magenta, and yellow), Figure B.9. It is used very often in colour printers and copiers which deposit pigment on paper. Each colour in this colour space is simply represented by the difference between white light and NOT reflected light. The following formula is used to convert the RGB to CMY:

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (\text{B.10})$$

The range of the C, M, and Y is assumed to be [0,1].

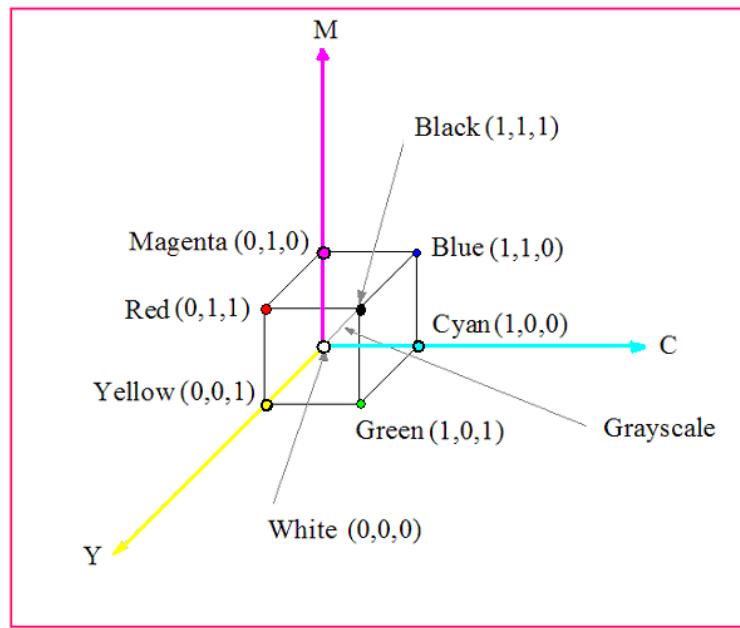


Figure B.9: The CMY colour space.

The CMYK is mostly used by printers. Since it is hard to mix the colours to get the black colour a dark grey is generated instead of black a pure black is added to this system as a separate colour with a range [0,1]. The conversion from the CMY to CMYK is achieved by the following formulas:

$$K = \min(C, M, Y) \quad (\text{B.11})$$

$$C = C - K \quad (\text{B.12})$$

$$M = M - K \quad (\text{B.13})$$

$$Y = Y - K \quad (\text{B.14})$$

Due to the same reasons described in the RGB colour space, it is impossible to describe all of the colours in the CMY colour space.

B.3.7 The Nrgb Colour Space

This is a simple colour space in which colour is represented by its normalised value with respect to the intensity. The impact of the light variations is reduced in this space, but it is not possible to recover the tint of the object when it is lit with intensity varying illumination. The Nrgb is given by:

$$N_C = \frac{C}{R+G+B}, \quad \text{for } C = (R, G, B) \quad (\text{B.15})$$

Figure B.10 shows an outdoor scene in RGB colour space converted to the normalised RGB (Nrgb) colour space. This colour space is more immune to the changes of illumination as long as that the changes takes place in white light.

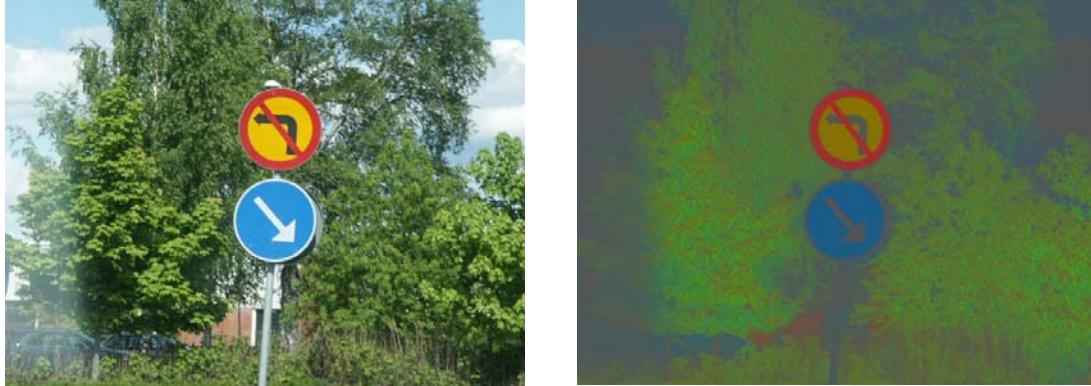


Figure B.10: The Nrgb colour space.

B.3.8 HSV (HSB) Colour Space

One of the big disadvantages of using the RGB colour space is its difficulty to separate the colour information from the brightness one. Instead of a set of colour primaries, the HSV (Hue, Saturation, and Value) colour space or as it is sometimes called HSB (Hue, Saturation, Brightness), created in 1978 by Alvey Ray Smith, uses colour descriptions which have a more intuitive appeal to users. To give a colour specification, a

user selects a spectral colour and the amounts of white or black which is to be added to obtain different shades, tints, and tones.

The three-dimensional representation of the HSV model is derived from the RGB cube. If the RGB cube is viewed along the diagonal from the white vertex to the origin (black), the outline of the cube, which has a hexagon shape, can be seen as shown in Figure B.11. The boundary of the hexagon represents the various hues and it is used as the top of the HSV hexcone; Figure B.12. In the hexcone, saturation is measured along a horizontal axis, and value along a vertical axis through the centre of the hexcone.

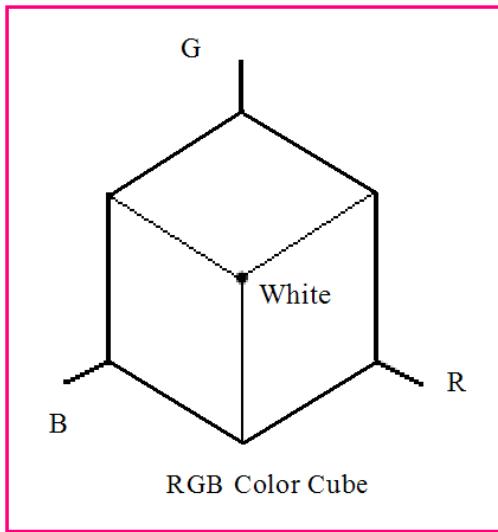


Figure B.11: RGB Colour Cube is viewed along the Diagonal from White to Black.

Hue (H) is described with the words we normally think of as describing colour: red, purple, blue, etc. Hue is more specifically described by the dominant wavelength in models such as the CIE system. Hue is also a term which describes a dimension of colour we readily experience when we look at colour. It will be the first of three dimensions we use to describe colour. It is represented as an angle about the vertical axis, ranging from 0° at red through 360° .

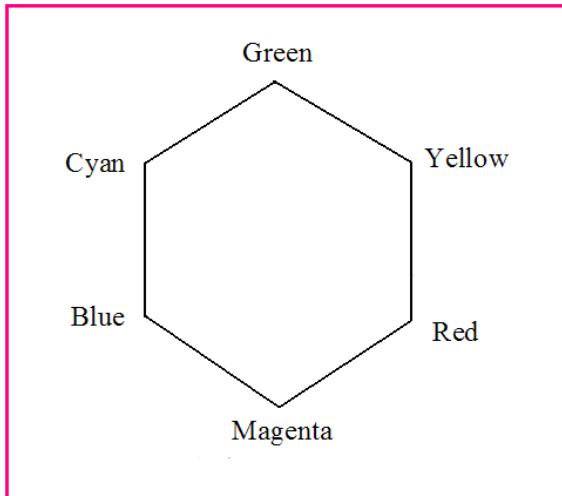


Figure B.12: The Colour-Cube Outline is a Hexagon.

The vertices of the hexagon are separated by 60° intervals. Yellow is at 60° , green at 120° , and cyan opposite red at $H = 180^\circ$, Figure B.13. Complementary colours are 180° apart. Figure B.14 shows the hue colour circle. In this figure, the outer border of the circle represents the pure hue, while the centre shows the white.

Saturation (S) refers to the dominance of hue in the colour. On the outer edge of the hue wheel are the 'pure' hues. Moving towards the centre of the wheel, the hues are used to describe decreasing/receding colour dominance. As the centre of the wheel is reached, no hue dominates. Those colours directly on the central axis are considered **desaturated**. These desaturated colours constitute the grey scale; running from white to black with all of the intermediate greys in between. Saturation, therefore, is the dimension running from the outer edge of the hue wheel (fully saturated) to the centre (fully desaturated), perpendicular to the value axis (Figure B.13). In terms of a spectral definition of colour, saturation is the ratio of the dominant wavelength to other wavelengths in the colour. White light is white because it contains an even balance of all wavelengths. It varies from 0 to 1. It is represented in this model as the ratio of the purity of a selected hue to its maximum purity at $S = 1$. A selected hue is said to be one-quarter pure at the value $S = 0.25$. At $S = 0$, we have the grey scale.

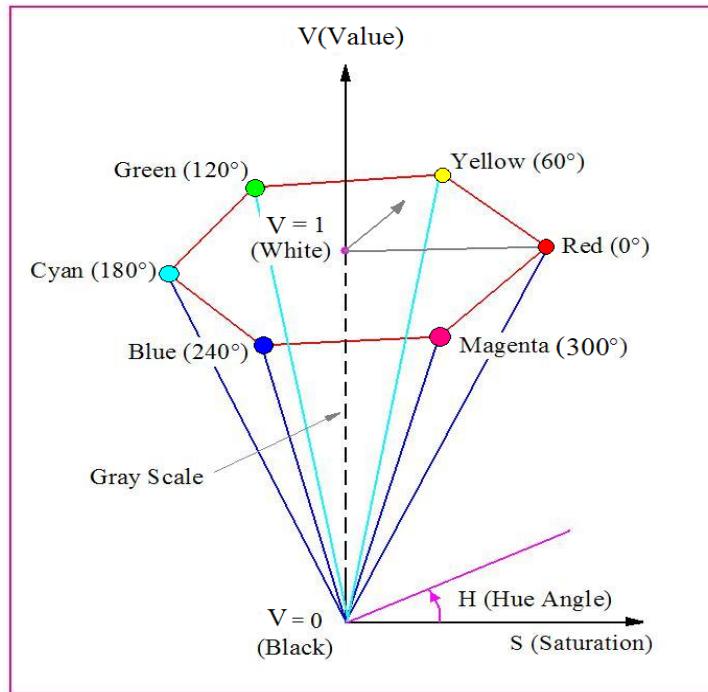


Figure B.13: HSV Colour Space.

Value V varies from 0 at the apex of the hexcone to 1 at the top. The apex represents black. At the top of the hexcone, colours have their maximum intensity. When $V = 1$ and $S = 1$, we have the “pure” hues. White is the point at $V = 1$ and $S = 0$. In terms of a spectral definition of colour, value describes the overall intensity or strength of the light. If hue can be thought of as a dimension going around a wheel, then value is a linear axis like an axis running through the middle of the wheel, Figure B.14 [120, 122].

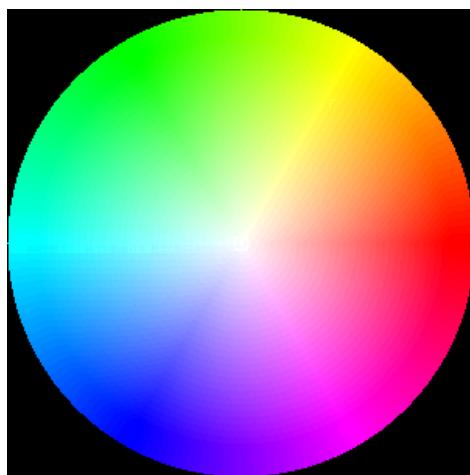


Figure B.14: The Hue Colour Circle.

Colour concepts associated with the terms shades, tints, and tones are represented in a cross-sectional plane of the HSI hexcone, Figure B.15. Adding black to a pure hue decreases V down the side of the hexcone. Thus, various shades are represented with values $S = 1$ and $0 \leq V \leq 1$. Adding white to a pure tone produces different tints across the top plane of the hexcone, where parameter values are $V = 1$ and $0 \leq S \leq 1$. Various tones are specified by adding both black and white, producing colour points within the triangular cross-sectional area of the hexcone [120].

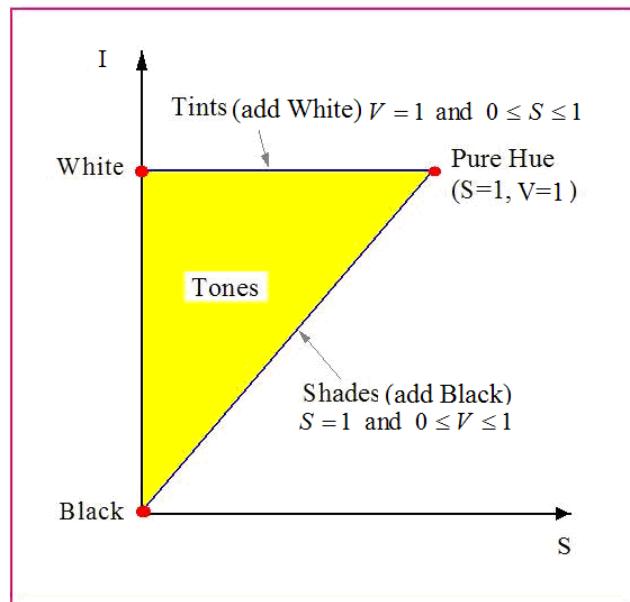


Figure B.15: Cross Section of HSV hexcone.

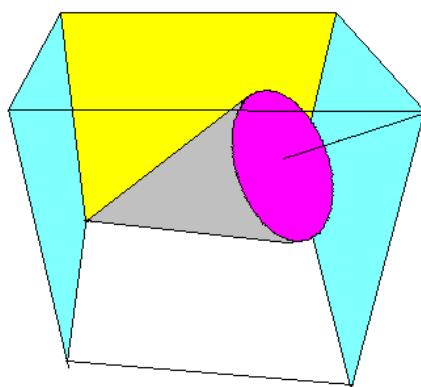


Figure B.16: The HSV inside the RGB Colour Space Box.

B.3.8.1 Converting Colours from RGB to HSV

Assuming that the RGB values are normalised to be in the range [0,1], the hue angle H is measured with respect to the red axis in the range [0,360], S and V in the range [0,1], the HSV components can be calculated from the RGB colour space as follows [123]:

The Value is given by:

$$V = \max(R, G, B) \quad (\text{B.16})$$

The Saturation component is calculated by:

$$S = \frac{\max(R, G, B) - \min(R, G, B)}{\max(R, G, B)} \quad \text{if } \max(R, G, B) \neq 0 \quad (\text{B.17})$$

$$S = 0 \quad \text{if } \max(R, G, B) = 0 \quad (\text{B.18})$$

The Hue is given by:

H is undefined if $S = 0$

$$H = \frac{G - B}{\max(R, G, B) - \min(R, G, B)} \quad \text{if } R = \max(R, G, B) \quad (\text{B.19})$$

$$H = 2.0 + \frac{B - R}{\max(R, G, B) - \min(R, G, B)} \quad \text{if } G = \max(R, G, B) \quad (\text{B.20})$$

$$H = 4.0 + \frac{R - G}{\max(R, G, B) - \min(R, G, B)} \quad \text{if } B = \max(R, G, B) \quad (\text{B.21})$$

$$H = 60 \times H$$

$$\text{if } H < 0 \text{ then } H = H + 360$$

Where H is the hue angle measured from the red.

B.3.8.2 Converting Colours from HSV to RGB

Given that the hue angle H is in the range [0,360] and S and V in the range [0,1], the RGB equivalents in the range [0,1] are calculated as follows [123]:

When $S = 0$, the colour is on the black-and-white centre line and there is no Hue. In this case, the algorithm assigns V to RGB as follows:

$$R = G = B = V$$

When $S \neq 0$, Hue exists and the RGB can be calculated as:

- Divide the Hue angle by 60 to specify which hex angle at which hue is located.

$$i = \left\lfloor \frac{H}{60} \right\rfloor \bmod 6 \quad (\text{B.22})$$

- Calculate the fractional part of H as

$$f = \frac{H}{60} - i \quad (\text{B.23})$$

- Calculate p, q , and t as:

$$p = V(1 - S) \quad (\text{B.24})$$

$$q = V(1 - Sf) \quad (\text{B.25})$$

$$t = V(1 - (1 - f)S) \quad (\text{B.26})$$

- Calculate the RGB according to the value i as follows:

○ if $i = 0$;

$$R = V$$

$$G = t$$

$$B = p$$

○ if $i = 1$;

$$R = q$$

$$G = V$$

$$B = p$$

○ if $i = 2$;

$$R = p$$

$$G = V$$

$$B = t$$

○ if $i = 3$;

$$R = p$$

$$G = q$$

$$B = V$$

○ if $i = 4$;

$$R = t$$

$$G = p$$

$$B = V$$

○ if $i = 5$;

$$R = V$$

$$G = p$$

$$B = q$$

To get an impression of what the hue, saturation, and value images look like, Figure B.17 shows an RGB image converted into HSV colour space. The Hue image is normalised from [0,360] to [0,255] so that it will be possible to show it as an image.

Saturation and Value images are normalised from [0,1] for each of them to [0,255] for the same reason.

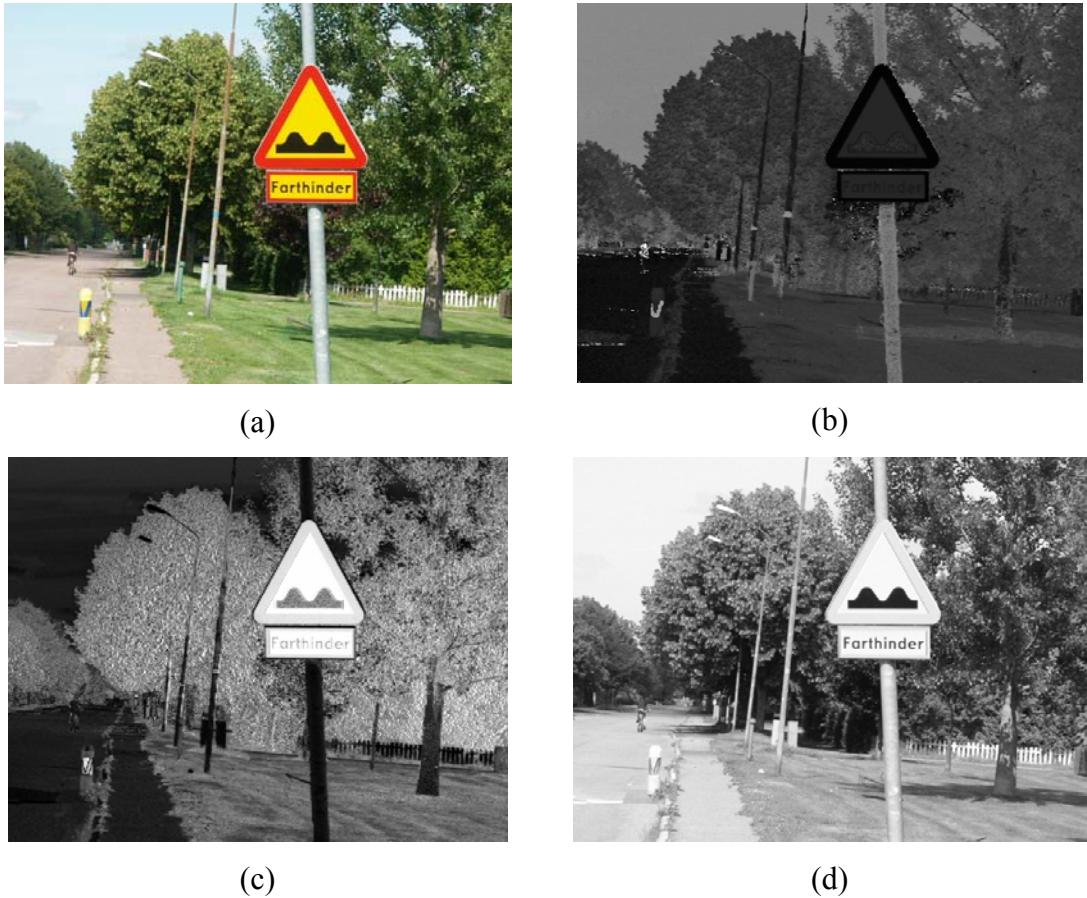


Figure B.17: (a) RGB image, (b) Hue, (c) Saturation, and (d) Value images.

B.3.9 The HSI (HLS) Colour Space

Every colour in this system is represented by three components: the hue (H): *the apparent colour of the light determined by the dominant wavelength*, the saturation (S): *the purity of colour*, and the intensity (I): *the total light across all frequencies*, or the synonymous system HLS (Hue, Lightness and Saturation). This colour space is represented by double hexagons or double cones attached to each other as shown in Figures B.18 and B.19. It is very important and attractive colour space for the applications of image processing because it represents colours in a way similar to which the human eye senses colour.

The Hue component describes the colour, and it is described by an angle between [0,360] degrees, where red is located at 0 degrees, green at 120 degrees, and blue at 240

degrees. The saturation shows how much the colour is mixed with white. It ranges between [0,1]. The intensity ranges between [0,1] where 0 represents black, and 1 is white.

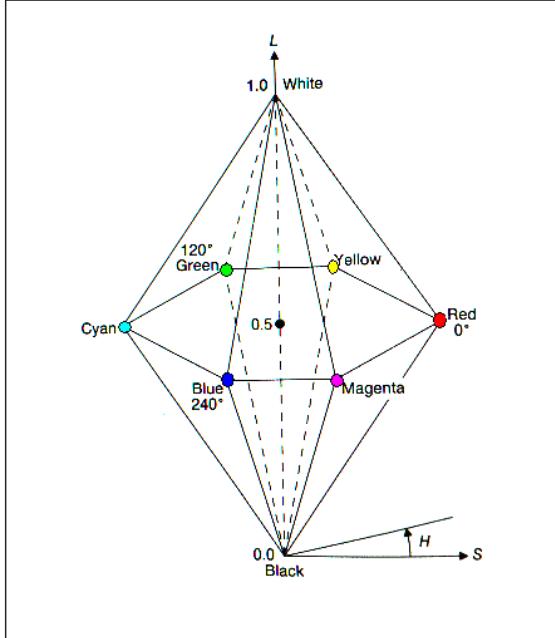


Figure B.18: The HSI (HLS) Hexagon form.

B.3.9.1 Converting Colours from RGB to HSI

Assuming that the RGB values are normalised to be in the range [0,1], and the angle θ is measured with respect to the red axis of the HSI space, the HSI components can be calculated from the RGB colour space. The HSI components can be calculated as follows [64]:

The Hue H is given by:

$$H = \theta \quad \text{if } B \leq G \quad (\text{B.27})$$

$$H = 360 - \theta \quad \text{if } B > G \quad (\text{B.28})$$

where:

$$\theta = \cos^{-1} \left\{ \frac{0.5 \times [(R-G) + (R-B)]}{\sqrt{(R-G)^2 + (G-B) \times (R-B)}} \right\} \quad (\text{B.29})$$

The saturation component is calculated by:

$$S = 1 - \frac{3}{(R+G+B)} \times [\min(R, G, B)] \quad (\text{B.30})$$

And the intensity is given by:

$$I = \frac{(R + G + B)}{3} \quad (\text{B.31})$$

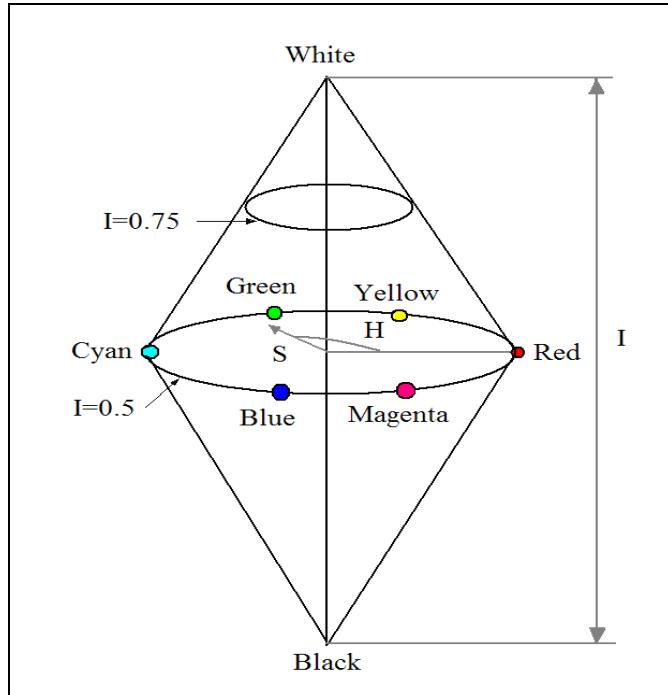


Figure B.19: The HSI (HLS) cone form.

B.3.9.2 Converting colours from HSI to RGB

Given the values of HSI in the interval $[0,1]$, the corresponding RGB values can be calculated. The set of equations used for this conversion depends mainly on the values of H. There are three sets of equations corresponding to the 120° interval which represent the separation of the primary colours. The conversion is carried out by two steps:

- Multiply H by 360° , to return the value of H to its original value.
- Calculate the RGB values according to the Hue sectors as seen below [124]:

For $0^\circ \leq H \leq 120^\circ$

$$R = \frac{I}{\sqrt{3}} \left[1 + \frac{S \cos(H)}{\cos(60^\circ - H)} \right] \quad (\text{B.32})$$

$$B = \frac{I}{\sqrt{3}} (1 - S) \quad (\text{B.33})$$

$$G = \sqrt{3} \times I - R - B \quad (\text{B.34})$$

For $120^\circ \leq H \leq 240^\circ$

$$G = \frac{I}{\sqrt{3}} \left[1 + \frac{S \cos(H - 120^\circ)}{\cos(180^\circ - H)} \right] \quad (\text{B.35})$$

$$R = \frac{I}{\sqrt{3}} (1 - S) \quad (\text{B.36})$$

$$B = \sqrt{3} \times I - R - G \quad (\text{B.37})$$

For $240^\circ \leq H \leq 360^\circ$

$$B = \frac{I}{\sqrt{3}} \left[1 + \frac{S \cos(H - 240^\circ)}{\cos(300^\circ - H)} \right] \quad (\text{B.38})$$

$$G = \frac{I}{\sqrt{3}} (1 - S) \quad (\text{B.39})$$

$$R = \sqrt{3} \times I - G - B \quad (\text{B.40})$$



(a)



(b)



(c)



(d)

Figure B.20: (a) RGB image, (b) Hue, (c) Saturation, and (d) Intensity images.

Figure B.20 shows an RGB image converted into the HSI (HLS) colour space. The values of Hue, Saturation and Intensity are normalised to [0,255] in order to show them as images.

B.3.10 Comparison of HSI (HSL) and HSV

The HSL and HSV colour spaces are similar to each other. The HSL colour space reflects the intuitive notion of saturation and lightness as two independent parameters. The consequences of this are better results for the HLS colour space. The behaviour of the HLS colour space is symmetrical to lightness and darkness which is not the case of the HSV colour space.

The saturation component of the HLS colour space gives fully saturated colour to the equivalent grey, while the HSV colour space cannot do this. The reason is that the value components of the HSV colour space are always calculated as the maximum of the RGB colours which makes the system move from saturated colour to white, which may be considered counterintuitive.

The Lightness in HSL always spans the entire range from black through the chosen hue to white (in HSV, the V component only goes half that way, from black to the chosen hue) [125].

There are some other common problems the two colour spaces share. Among them, in colour selection where lightness runs in the range [0,100], a lightness of 50 should lie in the middle of 0 and 100, but this is not the case in these two colour spaces. Simply speaking there is no reference to the lightness perception of the human vision.

The lightness or brightness in these two colour spaces takes place by $(R+G+B)/3$. This computation conflicts badly with the properties of colour vision, as it computes yellow to be six times more intense than blue for the same lightness.

Another problem may appear for both colour spaces is the discontinuity of hue at 360° . This leads to problems in the computations of colour mixtures in the polar coordinates. This same discontinuity problem appears in every 60° segment of the hue circle since the colour space involves different computations for each of these segments [119].

B.3.11 The Improved HLS colour space

Hanbury and Serra [126] introduced an improved version of HLS colour space which was later called IHLS. This colour space is very similar to the aforementioned colour spaces, but it avoids the inconveniences of the other colour spaces designed for computer graphics rather than image processing. The colour space provides independence between chromatic and achromatic components [127].

B.3.11.1 Converting Colours from RGB to IHLS

The conversion from the RGB to this colour space is done as follows [126]:

$$H = \theta \quad \text{if } B \leq G \quad (\text{B.41})$$

$$H = 360 - \theta \quad \text{if } B > G \quad (\text{B.42})$$

where:

$$\theta = \cos^{-1} \left\{ \frac{\left[R - \frac{G}{2} - \frac{B}{2} \right]}{\sqrt{R^2 + G^2 + B^2 - RG - RB - GB}} \right\} \quad (\text{B.43})$$

$$S = \max(R, G, B) - \min(R, G, B) \quad (\text{B.44})$$

$$L = 0.2126R + 0.7152G + 0.0722B \quad (\text{B.45})$$

B.3.11.2 Converting Colours from IHLS to RGB

The transformation of colours from the IHLS colour space to RGB can be obtained by calculating the chroma of values from the saturation values as follows [126]:

$$C = \frac{\sqrt{3}S}{2 \sin(120^\circ - H^*)} \quad (\text{B.46})$$

Where H^* is given by:

$$H^* = H - k \times 60^\circ \text{ where } k \in \{0, 1, 2, 3, 4, 5\} \text{ so that } 0^\circ \leq H^* \leq 60^\circ \quad (\text{B.47})$$

From the chroma, one calculates

$$C_1 = C \times \cos(H) \quad (\text{B.48})$$

$$C_2 = -C \times \sin(H) \quad (\text{B.49})$$

For the case where the hue is undefined: $C_1 = C_2 = 0$.

Finally, the RGB colours are calculated from the following equation:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.0000 & 0.7875 & 0.3714 \\ 1.0000 & -0.2125 & -0.2059 \\ 1.0000 & -0.2125 & 0.9488 \end{bmatrix} \begin{bmatrix} L \\ C_1 \\ C_2 \end{bmatrix} \quad (\text{B.50})$$

Figure B.21 shows an RGB image converted into IHLS colour space. The values of Hue, Saturation and Lightness are normalised to [0,255] in order to show them as images.

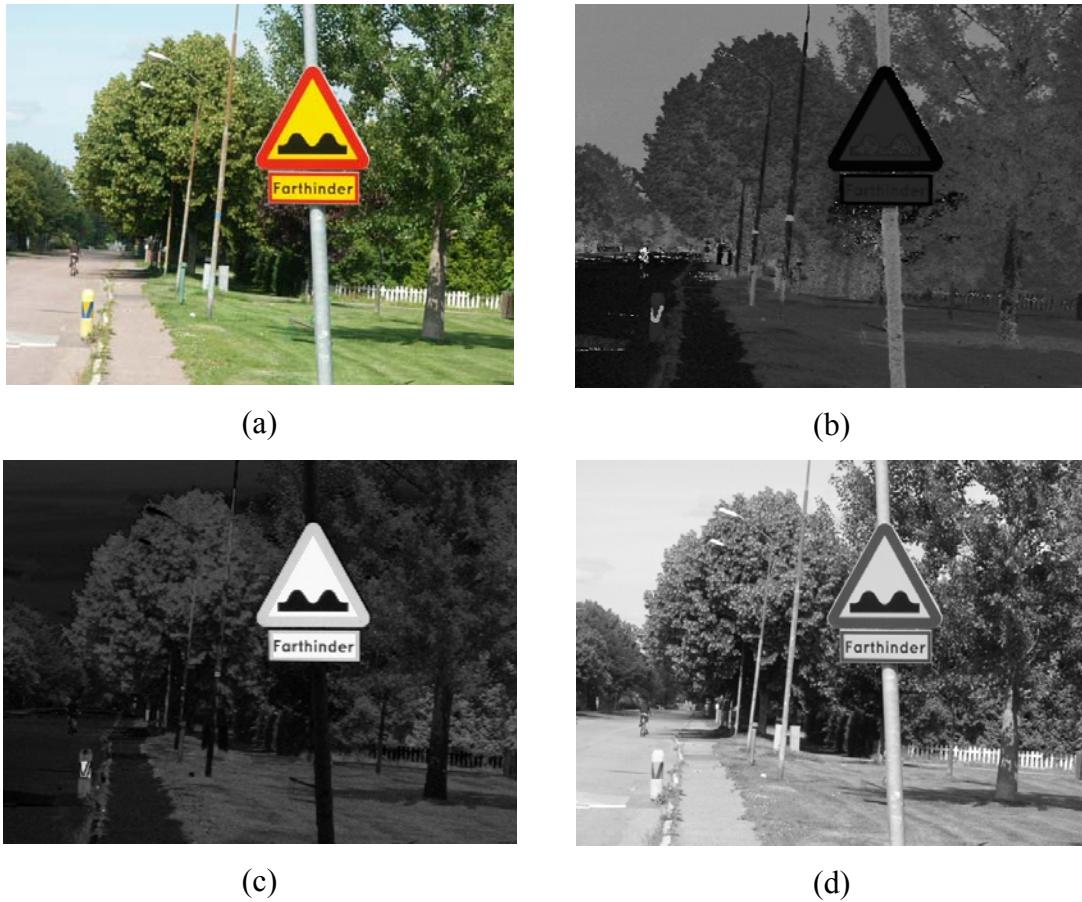


Figure B.21: (a) RGB image, (b) Hue, (c) Saturation, and (d) Lightness images.

B.3.12 The YIQ Colour Space

This colour space is used for NTSC television. The Y component represents the *luminance*; while the I image, which stands for *Intermodulation* and the Q image which stands for *quadrature* represent the chrominance information. The hue information of the colour is separated from the intensity. The system takes advantage of the fact that the human eye is more sensitive to changes in luminance than changes to hue or saturation. This separation has many advantages:

- Image processing algorithms can be applied on the intensity and colour information separately, such as applying histogram equalisation to enhance the image's contrast while keeping the colours the same.
- The system is used in TV broadcasting, because it helps to maintain compatibility with monochrome TV standards.

Given the RGB values in the range [0,1], the corresponding YIQ values in the range [0,1] can easily be computed. The conversion from the RGB colour space which ranges [0,1] to the YIQ which ranges [0,1] for Y and [-0.5,0.5] for I and Q, is linear and done using the following formula [121]:

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.275 & -0.321 \\ 0.212 & -0.523 & -0.311 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (\text{B.51})$$

An inverse matrix operation can convert the YIQ, which ranges [0,1] for Y and [-0.5,0.5] for I and Q, to RGB, which ranges [0,1], as follows [121]:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & 0.956 & 0.621 \\ 1 & -0.272 & -0.647 \\ 1 & -1.105 & 1.702 \end{bmatrix} \begin{bmatrix} Y \\ I \\ Q \end{bmatrix} \quad (\text{B.52})$$

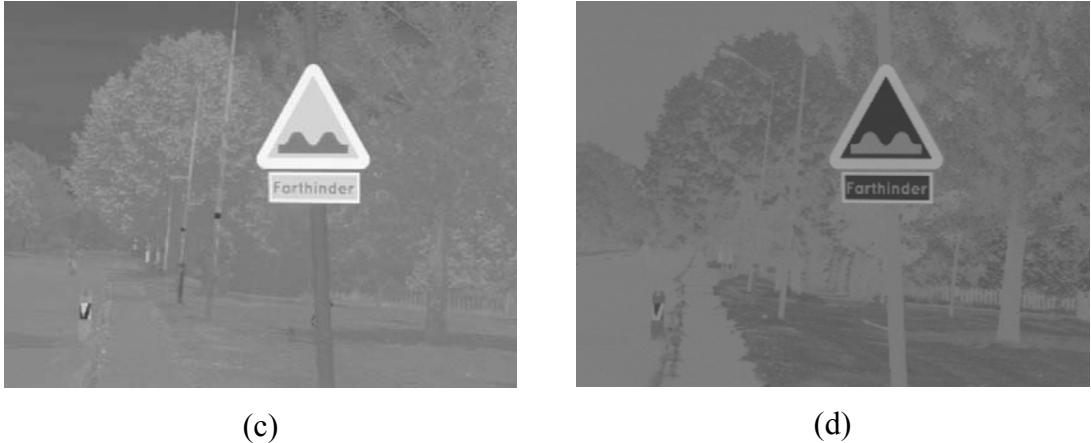
Figure B.22 shows the RGB image and its equivalent in the YIQ colour space. In this figure, the Y image is shown in Figure B.22(b); Figure B.22(c) and (d) show the I and Q images.



(a)



(b)



(c) (d)

Figure B.22: (a) RGB image, (b) Y image, (c) I image, and (d) Q images.

B.3.13 The YUV Colour Space

The YUV was originally used for PAL (European "standard") analogue video. The Y (range [0,1]) component represents the luminance and the U and V (range [-0.5,0.5], or [-128,127] in the signed digital form) are the Chrominance (colour) component. The colour space is based on the CIE Y primary, and also chrominance. Chrominance is the difference between a colour and a reference white at the same luminance.

To convert from RGB in the range [0,1] to YUV colour space (ranges are [0,1] for Y and [-0.5,0.5] for U and V), the following equation can be used:

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.147 & -0.289 & 0.436 \\ 0.615 & -0.515 & -0.100 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (\text{B.53})$$

On the other hand, the conversion from YUV colour space (ranges are [0,1] for Y and [-0.5,0.5] for U and V) to RGB in the range[0,1] is straightforward and achieved by the following equation:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.000 & 0.000 & 1.140 \\ 1.000 & -0.395 & -0.581 \\ 1.000 & 0.2032 & 0.000 \end{bmatrix} \begin{bmatrix} Y \\ U \\ V \end{bmatrix} \quad (\text{B.54})$$

Figure B.23 shows the RGB image and its equivalent in the YUV colour space. In this figure, the Y image is shown in Figure B.23(b); Figure B.23(c) and (d) show the U and V images.

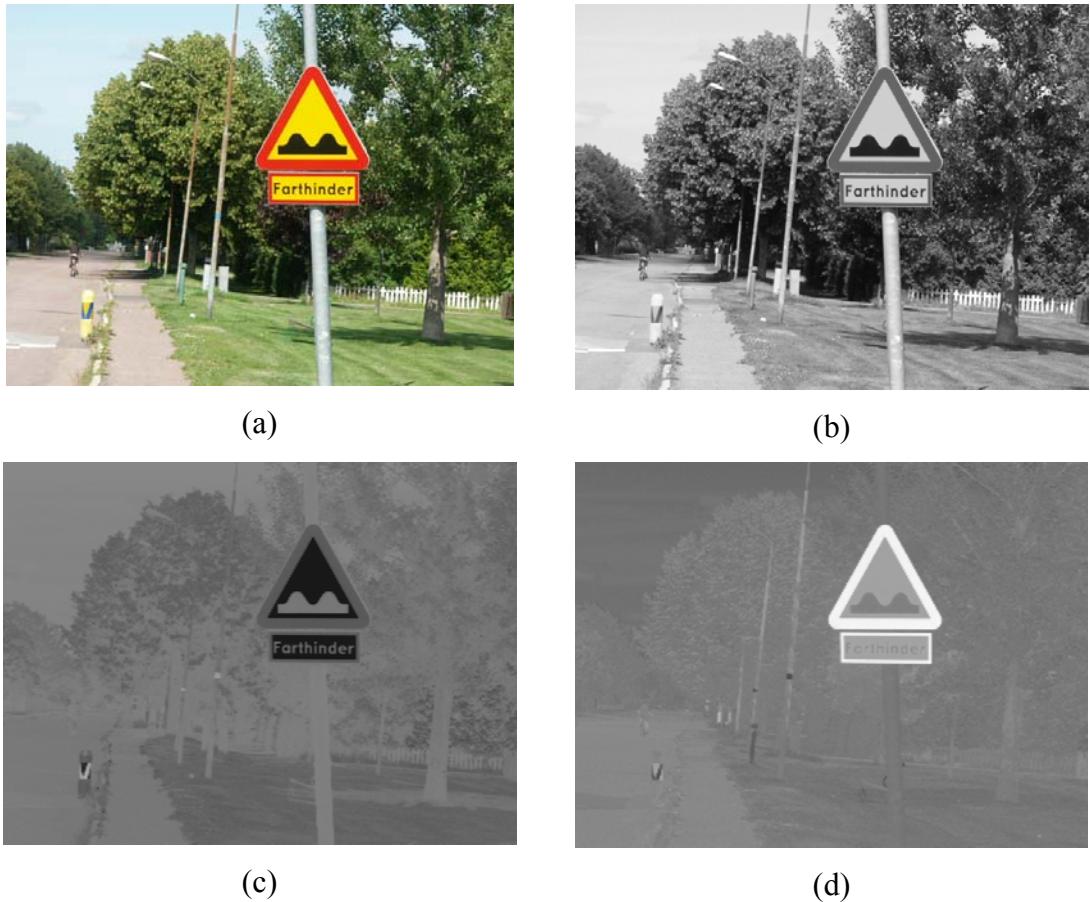


Figure B.23: (a) RGB image, (b) Y image, (c) U image, and (d) V images.

B.3.14 The YCbCr Colour Space

This colour space is similar to the YIQ and YUV colour spaces, but it was developed to work with digital television. As before, Y is the luminance component and Cb and Cr are the chroma component. This colour space is often confused with the YUV colour space, but they are, in fact, two different colour spaces.

The conversion can be achieved straightforwardly by a matrix form. Assuming that the RGB are gamma corrected and the values are in the range [0,1], the corresponding YCbCr values can be computed as follows [118]:

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.2989 & 0.5866 & 0.1145 \\ -0.1688 & -0.3312 & 0.5000 \\ 0.5000 & -0.4184 & -0.0816 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (B.55)$$

In this equation, the value of Y is in the range of [0,1], while the range of the chroma (Cb and Cr) is in the range [-0.5,0.5].

The conversion from the YCbCr to the RGB colour space is given by the following set of equations assuming that the range of the Y component is [0,1] and the range of Cb and Cr is [-0.5,0.5] [128]:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.0000 & 0.0000 & 1.4022 \\ 1.0000 & -0.3456 & -0.7145 \\ 1.0000 & 1.7710 & 0.0000 \end{bmatrix} \begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} \quad (B.56)$$

Figure B.24 shows the RGB image and the corresponding YCbCr images normalised to [0,255] grey levels.

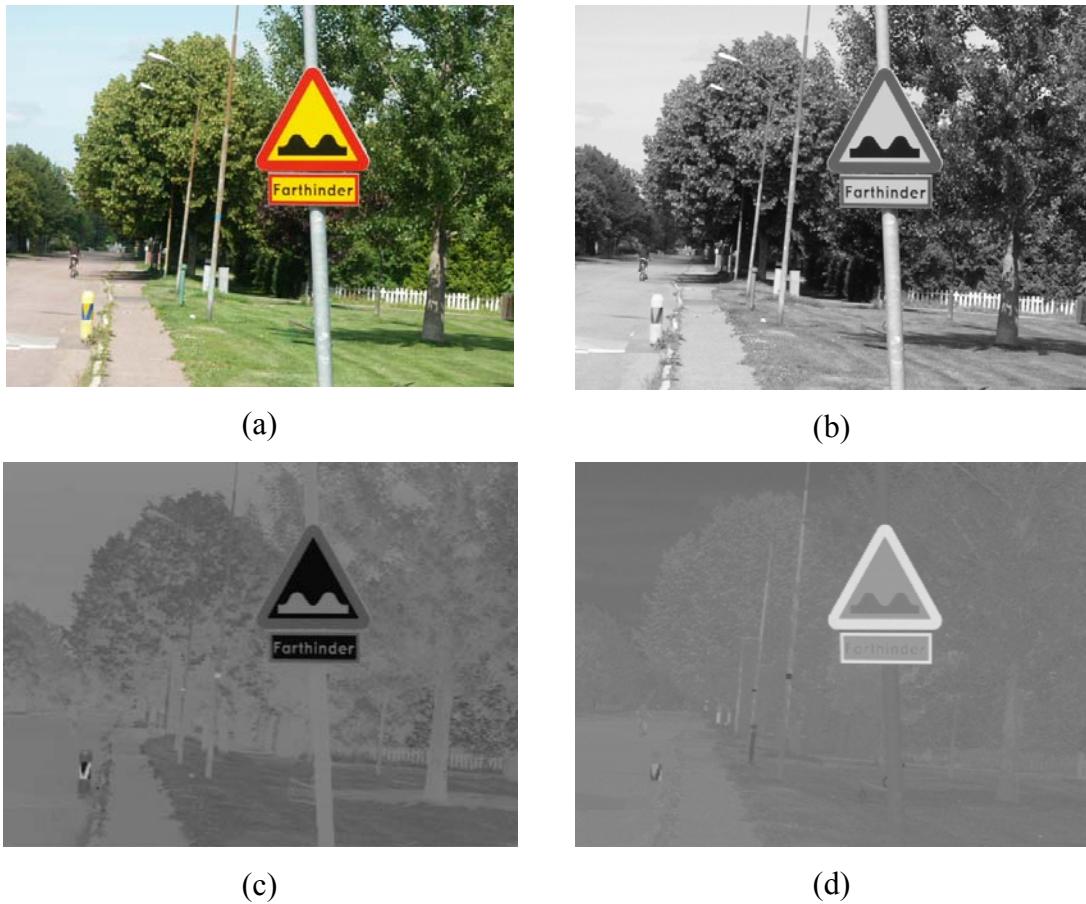


Figure B.24: (a) RGB image, (b) Y image, (c) Cb image, and (d) Cr images.

B.3.15 The L*a*b* Colour Space

This model is the most complete colour space because it can describe all the colours visible to human eye. It was developed by the CIE in 1976 as a refinement of the XYZ colour space because of the desirable property that a perceptual colour difference can be quantified by the Euclidian distance. The L*a*b* is device independent and represents

every colour through three components. The L value represents luminance which ranges from 0 for black to 100 for white. The a and b are represented as +a/-a for red/green and +b/-b for blue/yellow. The range of a* is [-500,500], while the range of b* is [-200,200], Figure B.25.

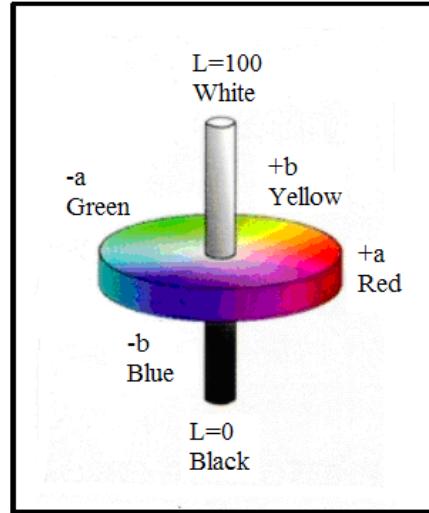


Figure B.25: The L*a*b* Colour Space.

The transformation from the RGB to L*a*b* coordinates is done by transforming the RGB into the XYZ; and the latter is transformed into L*a*b* where a linear relationship between them is assumed.

The transformation from XYZ to L*a*b* is given by:

$$L^* = 116 \left(\frac{Y}{Y_W} \right)^{\frac{1}{3}} - 16 \quad \text{if } \frac{Y}{Y_W} > 0.008856 \quad (\text{B.57})$$

$$L^* = 903.3 \left(\frac{Y}{Y_W} \right) \quad \text{if } \frac{Y}{Y_W} \leq 0.008856 \quad (\text{B.58})$$

$$a^* = 500 \left[f \left(\frac{X}{X_W} \right) - f \left(\frac{Y}{Y_W} \right) \right] \quad (\text{B.59})$$

$$b^* = 200 \left[f \left(\frac{Y}{Y_W} \right) - f \left(\frac{Z}{Z_W} \right) \right] \quad (\text{B.60})$$

Where

$$f(t) = t^{1/3} \quad \text{for } t > 0.008856 \quad (\text{B.61})$$

$$f(t) = 7.787t + \frac{16}{116} \quad \text{for } t \leq 0.008856$$

Where X_w , Y_w , and Z_w are reference white tristimulus values-typically the white of a perfectly reflecting diffuser under CIE standard D65 illumination which is defined as $x = 0.3127$ and $y = 0.3290$ in the CIE diagram.

The conversion from the $L^*a^*b^*$ to XYZ for ($Y/Y_W > 0.008856$) is given by:

$$X = X_W (P + a^*/500)^3 \quad (\text{B.62})$$

$$Y = Y_W P^3 \quad (\text{B.63})$$

$$Z = Z_W (P - b^*/200)^3 \quad (\text{B.64})$$

Where

$$P = (L^* + 16)/116 \quad (\text{B.65})$$

B.3.16 The $L^*u^*v^*$ Colour Space

In 1976 the CIE made another attempt to linearise the perceptibility of colour differences. The L component has the range [0,100], the u component has the range [-134,220], and the v component has the range [-140,122]. The non-linear relations for the L^*,u^* and v^* are given below:

$$L^* = 116 \left(\frac{Y}{Y_n} \right)^{\frac{1}{3}} - 16 \quad \text{if } \frac{Y}{Y_n} > 0.008856 \quad (\text{B.66})$$

$$L^* = 903.3 \left(\frac{Y}{Y_n} \right) \quad \text{if } \frac{Y}{Y_n} \leq 0.008856 \quad (\text{B.67})$$

$$u^* = 13L^*(u' - u'_n); \quad v^* = 13L^*(v' - v'_n) \quad (\text{B.68})$$

The quantities u'_n and v'_n refer to the reference white. For the 2° observer and illuminant C, $u'_n = 0.2009$, $v'_n = 0.4610$. The values of the u' and v' are calculated from the equations given below:

$$u' = 4X/(X + 15Y + 3Z) = 4x/(-2x + 12y + 3) \quad (\text{B.70})$$

$$v' = 9Y/(X + 15Y + 3Z) = 9y/(-2 + 12y + 3) \quad (\text{B.71})$$

Appendix C – Explicit Form of Zernike Polynomials

(p,q)	$R_{pq}(r)$
(0,0)	1
(1,1)	r
(2,0)	$2r^2 - 1$
(2,2)	r^2
(3,1)	$3r^3 - 2r$
(3,3)	r^3
(4,0)	$6r^4 - 6r^2 + 1$
(4,2)	$4r^4 - 3r^2$
(4,4)	r^4
(5,1)	$10r^5 - 12r^3 + 3r$
(5,3)	$5r^5 - 4r^3$
(5,5)	r^5
(6,0)	$20r^6 - 30r^4 + 12r^2 - 1$
(6,2)	$15r^6 - 20r^4 + 6r^2$
(6,4)	$6r^6 - 5r^4$
(6,6)	r^6
(7,1)	$35r^7 - 60r^5 + 30r^3 - 4r$
(7,3)	$21r^7 - 30r^5 + 10r^3$
(7,5)	$7r^7 - 6r^5$
(7,7)	r^7
(8,0)	$70r^8 - 140r^6 + 90r^4 - 20r^2 + 1$
(8,2)	$56r^8 - 105r^6 + 60r^4 - 10r^2$
(8,4)	$28r^8 - 42r^6 + 15r^4$
(8,6)	$8r^8 - 7r^6$
(8,8)	r^8
(9,1)	$126r^9 - 280r^7 + 210r^5 - 60r^3 + 5r$
(9,3)	$84r^9 - 168r^7 + 105r^5 - 20r^3$
(9,5)	$36r^9 - 56r^7 + 21r^5$
(9,7)	$9r^9 - 8r^7$

(9,9)	r^9
(10,0)	$252r^{10} - 630r^8 + 560r^6 - 210r^4 + 30r^2 - 1$
(10,2)	$210r^{10} - 504r^8 + 420r^6 - 140r^4 + 15r^2$
(10,4)	$120r^{10} - 252r^8 + 168r^6 - 35r^4$
(10,6)	$45r^{10} - 72r^8 + 28r^6$
(10,8)	$10r^{10} - 9r^8$
(10,10)	r^{10}
(11,1)	$462r^{11} - 1260r^9 + 1260r^7 - 560r^5 + 105r^3 - 6r$
(11,3)	$330r^{11} - 840r^9 + 756r^7 - 280r^5 + 35r^3$
(11,5)	$165r^{11} - 360r^9 + 252r^7 - 56r^5$
(11,7)	$55r^{11} - 90r^9 + 26r^7$
(11,9)	$11r^{11} - 10r^9$
(11,11)	r^{11}
(12,0)	$924r^{12} - 2772r^{10} + 3150r^8 - 1680r^6 + 420r^4 - 42r^2 + 1$
(12,2)	$792r^{12} - 2310r^{10} + 2520r^8 - 1260r^6 + 280r^4 - 21r^2$
(12,4)	$495r^{12} - 1320r^{10} + 1260r^8 - 504r^6 + 70r^4$
(12,6)	$220r^{12} - 495r^{10} + 360r^8 - 84r^6$
(12,8)	$66r^{12} - 110r^{10} + 45r^8$
(12,10)	$12r^{12} - 11r^{10}$
(12,12)	r^{12}

Appendix D – Results of Training and Testing of SVM

Table D.1: Classification rates of Training and Testing using Binary Images
SVM Type: C-SVM, Kernel = Linear, C=1

Experiment	Shapes		Speed-Limit	
	Training%	Testing%	Training%	Testing%
1	100	100	100	100
2	100	100	100	96.0
3	100	100	100	100
4	100	100	100	96.0
5	100	100	100	97.3
6	100	100	100	97.3
7	100	100	100	100
8	100	100	100	97.3
9	100	100	100	97.3
10	100	100	100	100
Ave	100	100	100	98.1

Table D.2: Classification rates of Training and Testing using Geometric moments
SVM Type: C-SVM, Kernel = Linear, C=1

Experiment	Shapes		Speed-Limit	
	Training%	Testing%	Training%	Testing%
1	95.7	95.5	81.1	70.6
2	96.2	98.9	80.0	68.0
3	95.7	92.2	74.8	69.3
4	95.2	96.6	77.7	72.0
5	96.2	96.6	81.7	60.0
6	95.7	97.7	80.6	70.7
7	96.6	93.3	74.3	74.7
8	95.2	97.7	81.7	65.3
9	96.6	95.5	76.0	68.0
10	95.9	97.7	77.7	68.0
Ave	95.8	96.2	78.5	68.6

Table D.3: Classification rates of Training and Testing using Zernike moments
SVM Type: C-SVM, Kernel = Linear, C=1

Experiment	Shapes		Speed-Limit	
	Training%	Testing%	Training%	Testing%
1	100	100	97.7	89.3
2	100	97.7	98.8	85.3
3	100	96.6	100	82.7
4	100	100	99.4	89.3
5	100	97.7	100	86.7
6	100	100	97.7	93.3
7	100	98.9	99.4	85.3
8	100	100	98.8	89.3
9	100	100	98.2	90.7
10	100	98.9	98.2	89.3
Ave	100	98.9	98.8	88.1

Table D.4: Classification rates of Training and Testing using Legendre moments
SVM Type: C-SVM, Kernel = Linear, C=1

Experiment	Shapes		Speed-Limit	
	Training%	Testing%	Training%	Testing%
1	100	98.9	100	98.7
2	100	100	100	98.7
3	100	100	100	98.7
4	100	100	100	97.3
5	100	98.9	100	97.3
6	100	98.9	100	97.3
7	100	100	100	98.7
8	100	100	100	97.3
9	100	100	100	98.7
10	100	100	100	100
Ave	100	99.7	100	98.3

Table D.5: Classification rates of Training and Testing using OFMM moments
SVM Type: C-SVM, Kernel = Linear, C=1

Experiment	Shapes		Speed-Limit	
	Training%	Testing%	Training%	Testing%
1	97.6	94.4	43.4	54.7
2	97.1	90.0	61.7	54.6
3	95.2	98.8	45.7	54.7
4	96.6	96.6	59.4	50.6
5	97.1	94.4	62.8	50.6
6	97.1	92.2	49.1	42.7
7	97.1	93.3	50.6	43.0
8	96.2	97.8	48.0	46.7
9	95.7	97.8	61.7	57.3
10	96.7	92.2	51.4	53.3
Ave	96.6	94.7	53.4	50.0

Table D.6: Classification rates of Training and Testing using Haar Features
SVM Type: C-SVM, Kernel = Linear, C=1

Experiment	Shapes		Speed-Limit	
	Training%	Testing%	Training%	Testing%
1	99.5	98.8	97.1	93.3
2	99.5	98.8	99.4	93.3
3	99.5	98.8	98.3	96.0
4	99.5	100	97.1	94.7
5	99.5	100	96.0	98.7
6	99.5	100	99.4	96.0
7	100	98.8	98.8	90.7
8	100	97.7	98.3	94.7
9	99.5	100	98.3	94.7
10	99.5	98.8	98.3	94.7
Ave	99.6	99.1	98.1	94.7

Appendix E – The Access Database

The database presented in this appendix is to manage the big number of images collected in this research. The total number of images collected in Sweden is 3415 images. There are collected between the years 2003 and 2006 in different parts of Sweden. Most of the images are collected at Dalarna province, Stockholm and Malmö while the rest are collected in other places such as Goteborg province. It is a tedious work to classify these images manually; therefore an automatic system can be very helpful. The database is created using Microsoft Access 2000.

Basically, there are two objects which can be classified: pictures and signs. These objects are represented in two tables shown in figures E.3 and E.4. However, there are other supplementary tables which are used to describe the phrases used for different conditions. Figure E.1 depicts the tables used in this database, while Figure E.2 shows the relationships among these tables.

Pictures are collected in different light conditions, different weather conditions, different light geometry, and they may contain some defects. Further, as signs belong to different categories they have different colours and shapes, and they may be found in different conditions. To represent all these conditions in the database, a number of nine tables are created to represent the phrases used for different conditions, among them four tables are used to represent picture conditions and five tables to represent sign conditions. These tables are shown in Figure E.5.

Beside this number of tables, the database contains three forms to enter the specifications of the pictures and signs in the corresponding tables. The first form is invoked to access the picture table, the second to access the sign table and the third is to access both tables at the same time. These forms are depicted in figures E.6-E.8, respectively.

A number of queries are prepared for different conditions depending on the set of requirements involved in the search through the database. Figure E.9 shows one example of a query.

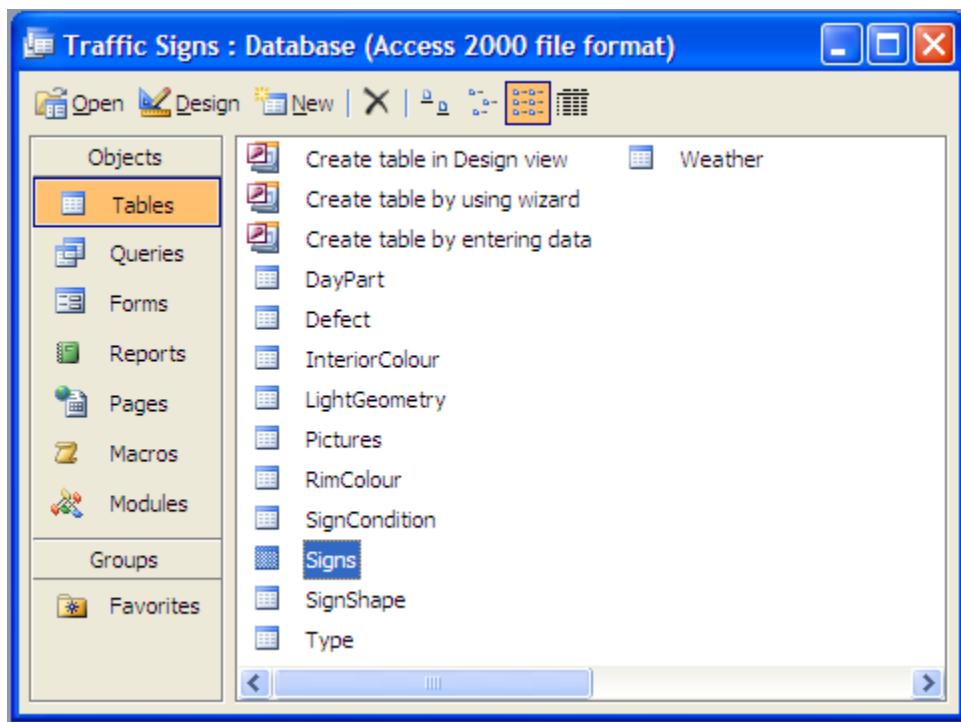


Figure E.1: Tables used in the database.

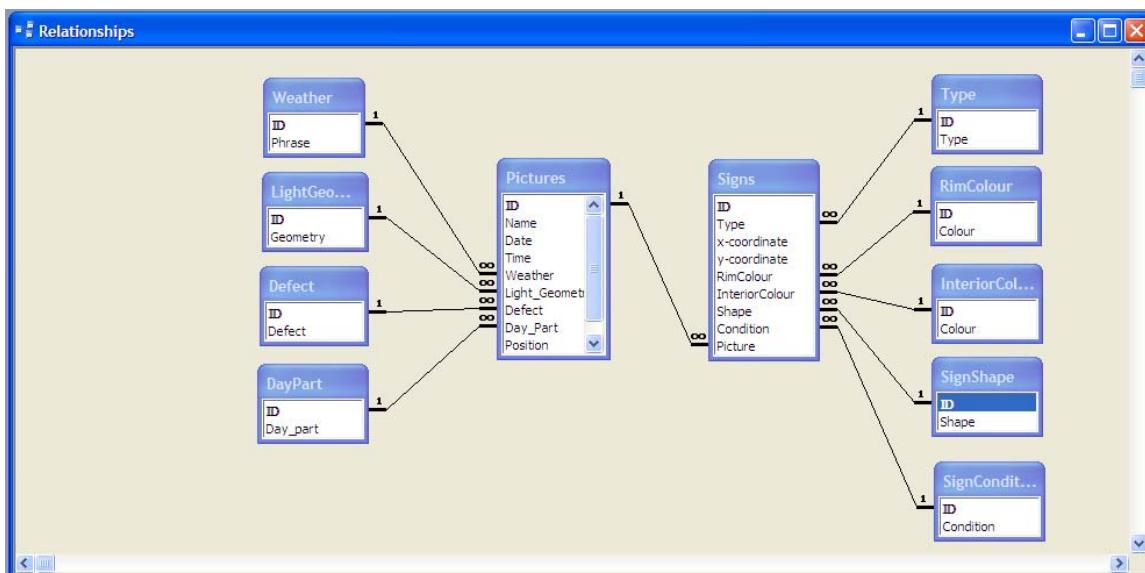


Figure E.2: The relationships among tables.

Pictures : Table

	ID	Name	Date	Time	Weather	Light_Geometry	Defect	Day_Part	Position
▶	1	PICT0001	31/05/2004	16:05:00	SN	G	N	DY	
+	2	PICT0002	31/05/2004	16:05:00	SN	G	N	DY	
+	3	PICT0003	07/03/2004	12:46:00	SN	G	N	DY	
+	4	PICT0004	06/03/2004	18:32:00	CD	B	N	DU	
+	5	PICT0005	29/08/2005	09:36:00	CD	G	N	DY	
+	6	PICT0006	06/03/2004	18:37:00	CD	B	N	DU	
+	7	PICT0007	06/03/2004	18:37:00	CD	B	N	DU	
+	8	PICT0008	06/03/2004	18:37:00	CD	B	N	DU	
+	9	PICT0009	06/03/2004	18:38:00	CD	G	N	DU	
+	10	PICT0010	06/03/2004	18:40:00	CD	G	N	DU	
+	11	PICT0011	06/03/2004	18:40:00	CD	G	N	DU	
+	12	PICT0012	18/12/2004	12:17:00	RN	G	S	DY	
+	13	PICT0013	18/12/2004	12:17:00	RN	G	S	DY	
+	14	PICT0014	18/12/2004	12:17:00	RN	G	S	DY	
+	15	PICT0015	18/12/2004	12:20:00	RN	G	S	DY	
+	16	PICT0016	18/12/2004	12:20:00	RN	G	S	DY	
+	17	PICT0017	06/03/2004	18:47:00	CD	G	N	DU	
+	18	PICT0018	06/03/2004	18:47:00	CD	G	N	DU	
+	19	PICT0019	06/03/2004	18:48:00	CD	G	N	DU	
+	20	PICT0020	06/03/2004	18:48:00	CD	B	N	DU	
+	21	PICT0021	18/12/2004	12:25:00	RN	G	S	DY	
+	22	PICT0022	18/12/2004	12:25:00	RN	G	S	DY	
+	23	PICT0023	18/12/2004	12:25:00	RN	G	S	DY	
+	24	PICT0024	18/12/2004	12:25:00	RN	G	S	DY	
+	25	PICT0025	18/12/2004	12:26:00	RN	G	S	DY	
+	26	PICT0026	18/12/2004	12:26:00	RN	G	S	DY	
+	27	PICT0027	18/12/2004	12:26:00	RN	G	S	DY	
+	28	PICT0028	31/05/2004	14:36:00	CD	B	N	DY	
+	29	PICT0029	31/05/2004	14:36:00	SC	G	N	DY	

Figure E.3: Pictures table.

Signs : Table

	ID	Type	x-coordinate	y-coordinate	RimColour	InteriorColour	Shape	Condition	Picture
▶	1	Warning	206	333 R	Y	UT	D		1
2	Supplementary	250	333 R	Y	R	G			1
3	Prohibitory	6	595 R	B	C	C			1
4	Warning	195	335 R	Y	UT	D			2
5	Supplementary	242	333 R	Y	R	G			2
7	Informative	333	471 B	B	R	G			3
8	Informative	392	465 B	B	R	G			3
9	Informative	143	160 B	B	R	G			4
10	Informative	182	451 B	B	R	G			4
11	Prohibitory	170	166 R	Y	C	G			5
12	Informative	145	444 B	B	R	C			6
13	Informative	164	186 B	B	R	G			7
14	Informative	169	377 B	B	R	G			7
15	Informative	99	64 B	B	R	C			8
16	Informative	92	323 B	B	R	G			8
17	Prohibitory	221	350 R	Y	C	G			9
18		221	391 W	Y	D	G			9
19	Informative	253	336 B	B	R	O			9
20	Informative	271	366 B	Y	R	O			9
21	Informative	148	326 B	B	R	G			10
22	Informative	276	326 B	B	R	G			10
23		324	356 W	W	R	G			10
24	Informative	236	300 B	B	R	G			11
25	Informative	259	297 B	B	R	G			11
26		273	297 W	W	R	G			11
27	Warning	195	310 R	Y	R	G			12
28	Warning	266	307 R	Y	UT	G			12
29	Supplementary	302	306 R	Y	R	G			12
30		305	62 W	Y	D	G			12

Figure E.4: Signs table.

Weather : Table

	ID	Phrase
+	CD	cloudy
+	FG	fog
+	RN	rain
+	SC	sunny/cloudy
+	SN	sunny
+	SW	snow

Record: [◀◀] [◀] 7 [▶] [▶▶]

DayPart : Table

	ID	Day_part
+	DU	dusk
+	DW	dawn
+	DY	day
+	SS	sunset
+	SU	sunrise

Record: [◀◀] [◀] 6 [▶] [▶▶]

LightGeometry : T...

	ID	Geometry
+	B	bad
+	G	good

Record: [◀◀] [◀] 3 [▶] [▶▶]

Defect : Table

	ID	Defect
+	B	blurred
+	D	dark
+	N	no
+	S	noisy

Record: [◀◀] [◀] 5 [▶] [▶▶]

Type : Table

	ID	Type
+	Informative	informative
+	Prohibitory	prohibitory
+	Supplementary	supplementary
+	Warning	warning

Record: [◀◀] [◀] 5 [▶] [▶▶]

RimColour : Table

	ID	Colour
+	B	blue
+	G	green
+	R	red
+	W	white

Record: [◀◀] [◀] 5 [▶] [▶▶]

InteriorColour : Ta...

	ID	Colour
+	B	blue
+	G	green
+	R	red
+	W	white
+	Y	yellow

Record: [◀◀] [◀] 6 [▶] [▶▶]

SignShape : Table

	ID	Shape
+	C	circle
+	D	diamond
+	DT	down triangle
+	O	octagone
+	R	rectangle
+	UT	up triangle

Record: [◀◀] [◀] 7 [▶] [▶▶]

SignCondition : Table

	ID	Condition
+	C	cut
+	D	damaged
+	F	faded
+	G	good
+	H	highlight
+	O	occluded
+	S	shadow
>		

Record: [◀◀] [◀] [▶] [▶▶] 8 [◀▶]

Figure E.5: The Phrases tables.

Pictures : Form

Traffic Sign Recognition

Pictures List

Name: PICT1013 Date: Time:

Weather: SN Light_Geometry: G Defect: N

Day_Part: DY

Record: [◀◀] [◀] [▶] [▶▶] 1 [◀▶] [◀▶] [◀▶] of 1012

Figure E.6: The Pictures list form.

Signs : Form

Traffic Sign Recognition

Signs List

Type: Warning x-coordinate: 206 y-coordinate: 333

Rim Colour: R Interior Colour: Y Shape: UT

Sign's Condition: D Picture: 

Record: [◀◀] [◀] [▶] [▶▶] 1 [◀▶] [◀▶] [◀▶] of 32

Figure E.7: The Signs list form.

Picturetbl

Name	PICT1000			
Date	31/05/2004			
Time	16:05:00			
Environment	SN			
Light Geometry	G			
Defect	N			
Signtbl				
Type	x-coordinat	y-coordinat	Border Color	Interior
Warning	206	333	R	Y
Supplementary	250	333	R	Y
Prohibitory	6	595	R	B
	0	0		

Record: [◀◀] [◀] [▶] [▶▶] [◀◀◀◀] [▶▶▶▶] of 4

Record: [◀◀] [◀] [▶] [▶▶] [◀◀◀◀] [▶▶▶▶] of 1000

Figure E.8: Pictures and Signs form.

sunny-good : Select Query

	Name	Weather	Light_Geometry	Date	Time	▲
	PICT0002	SN	G	31/05/2004	16:05:00	↑
	PICT0003	SN	G	07/03/2004	12:46:00	
	PICT0034	SN	G	12/03/2004	15:48:00	
	PICT0040	SN	G	07/03/2004	12:29:00	
	PICT0041	SN	G	07/03/2004	12:40:00	
	PICT0042	SN	G	07/03/2004	12:41:00	
	PICT0043	SN	G	07/03/2004	12:40:00	
►	PICT0046	SN	G	07/03/2004	12:41:00	
	PICT0047	SN	G	07/03/2004	12:41:00	
	PICT0048	SN	G	07/03/2004	12:41:00	
	PICT0049	SN	G	07/03/2004	12:41:00	
	PICT0050	SN	G	07/03/2004	12:42:00	
	PICT0051	SN	G	07/03/2004	12:42:00	
	PICT0052	SN	G	31/05/2004	14:24:00	
	PICT0053	SN	G	31/05/2004	14:25:00	
	PICT0054	SN	G	12/03/2004	15:55:00	
	PICT0055	SN	G	07/03/2004	12:42:00	
	PICT0058	SN	G	07/03/2004	12:43:00	
	PICT0059	SN	G	07/03/2004	12:43:00	
	PICT0060	SN	G	31/05/2004	14:36:00	
	PICT0061	SN	G	07/03/2004	12:43:00	
	PICT0062	SN	G	31/05/2004	14:37:00	
	PICT0063	SN	G	07/03/2004	12:43:00	
	PICT0064	SN	G	07/03/2004	12:43:00	▼

Record: [◀◀] [◀] [▶] [▶▶] [◀◀◀◀] [▶▶▶▶] of 401

Figure E.5: An example of a query.

Appendix F – List of Publications

1. Fleyeh, H., "Color Detection and Segmentation for Road and Traffic Signs" IEEE Conference on Cybernetics and Intelligent Systems, Singapore, December, 2004.
2. Fleyeh, H., "Traffic Signs Color Detection and Segmentation in Poor Light Conditions" Machine Vision Applications (MVA2005), Tsukuba Science City, Japan, 16-18 May, 2005.
3. Fleyeh, H., "Road and Traffic Sign Color Detection and Segmentation - A Fuzzy Approach" Machine Vision Applications (MVA2005), Tsukuba Science City, Japan, 16-18 May, 2005.
4. Fleyeh, H. and Dougherty, M., "Road and Traffic Sign Detection and Recognition" 10th EWGT Meeting and 16th Mini-EURO Conference, Poznan, Poland, 13-16 September, 2005.
5. Fleyeh, H., "A Novel Fuzzy Approach for Shape Determination of Traffic Signs" Second Indian International Conference on Artificial Intelligence (IICAI-05), Pune, India, 20-22 December, 2005.
6. Fleyeh, H., "Shadow and Highlight Invariant Colour Segmentation Algorithm For Traffic Signs" second IEEE Conference on Cybernetics and Intelligent Systems, Bangkok, Thailand, June, 2006.
7. Fleyeh, H., Gilani, S. , Dougherty, M., "Road Sign Detection and Recognition Using A Fuzzy ARTMAP: A Case Study Swedish Speed-Limit Signs" 10th IASTED International Conference on Artificial Intelligence and Soft Computing, Palma de Mallorca, Spain , 28-30 August, 2006.

8. Fleyeh, H., Dougherty, M., Aenugula, D., and Baddam S., "Invariant Road Sign Recognition with Fuzzy ARTMAP and Zernike Moments" IEEE-IV2007, Istanbul, Turkey, 13-15 June, 2007.
9. Fleyeh, H., Dougherty, M., Gilani, S.," Detection and Recognition of Road Signs Using Colour Information and Fuzzy ARTMAP" , IEEE Trans. Transportation (Review).
10. Fleyeh, H. and Dougherty, M.," SVM Based Traffic Sign Classification Using Legendre Moments" Third Indian International Conference on Artificial Intelligence (IICAI-07), Pune, India, 17-19 December, 2007.