
Adaptive Traffic Sign Recognition

Frank Lindner
06. February 2012



Dissertation

Erstgutachter:
Prof. Dr. Franz Kummert
Universität Bielefeld

Zweitgutachter:
Prof. Dr. Rainer Ott
Universität Stuttgart

Acknowledgements

This thesis would not have been possible unless my whole family, my parents Helga and Günter Lindner, my sister Antje Lindner and foremost my wife Hella had not pushed me into publishing the results of my work in this thesis.

I am grateful to Professor Dr Ott for supporting my wish to write this thesis and proofreading it.

My thanks go to Dr Ulrich Kressel for many helpful discussions and the constructive criticism concerning the evaluations.

I am thankful to Prof Dr Christian Woehler for his support and fruitful arguments concerning the algorithms used in the thesis.

In addition I would like to thank Dr Lars Krueger for his help with all kinds LaTeX problems I encountered during the writing of this thesis and for providing beautiful 3D-graphics.

Furthermore I thank all my colleagues who provided the environment in which the thesis could be written.

Contents

1	Introduction and State of the Art	1
1.1	Introduction	1
1.2	Characterization of Speed Limit Signs in Europe	2
1.3	State of the Art	3
1.4	Traffic Sign Recognition Research Groups	8
1.4.1	PROMETHEUS Groups	9
1.4.2	Radial Symmetry Detector	12
1.4.3	AdaBoost Approach	12
1.5	Scope of the Thesis	13
1.5.1	Modular and Adaptable Traffic Sign Recognition System	14
1.5.2	Supplementary Signs	14
1.5.3	Internationalization and Classifier from Scratch	15
1.5.4	Evaluation of a Complex System	17
1.6	Thesis Overview	18
2	Runtime System Design - and Implementation	19
2.1	General Requirements	19
2.2	System Overview	20
2.3	Camera Requirements	21
2.4	Traffic Sign Detector	22
2.4.1	Coarse Detector	23
2.4.2	Detection Verification	24
2.5	Traffic Sign Tracker	26
2.5.1	General Setup	27
2.5.2	Prediction Without Egomotion	28
2.5.3	Prediction Using Three Dimensional Object Position Estimation	29
2.5.4	Three Dimensional Positioning Formulas	31
2.5.5	Selfcalibration of the Three Dimensional Positioning System	34
2.5.6	Solving Assignment	38
2.5.7	Use of the Tracker as Blackboard	38
2.6	Classifier	39
2.6.1	Normalization	41
2.6.2	Single Pattern Classifier	43
2.6.3	Track Classification	45
2.7	Supplementary Sign Detection and Recognition	45

2.7.1	Supplementary Sign Detection	46
2.7.2	Supplementary Sign Verification	49
2.7.3	Supplementary Sign Recognition	50
2.8	Extension Modules for Improving Traffic Scene Interpretation . .	51
2.8.1	Use of the Three Dimensional Information	52
2.8.2	Refinement Classifiers	52
2.8.3	Fusion with Additional Inputs	53
3	General Classifier Adaptation Approach	54
3.1	Classifier setup	54
3.2	Obtaining Sample Sets	54
3.3	Training of the Classifier for a New Country	55
3.4	Training of Classifiers for Country Sets	57
4	Offline System and Classifier Training	58
4.1	Classifier Design	58
4.1.1	Training of the Principal Component Analysis	61
4.1.2	Training of the Polynomial Classifier	62
4.2	Classifier Internationalization	62
4.2.1	Obtaining Training Samples	65
4.2.2	Synthetic Samples	67
4.2.3	Knowledge-Based Autonomous Sample Collection	70
4.3	Design and Implementation of a System for the Automation of Sample Collection	71
4.3.1	Bootstrap Training and Classifier Refinement	73
4.3.2	Selection of Samples in the Bootstrapping Process	75
4.4	Combination of Classifiers of Different Countries	77
5	Design of the Evaluation System	78
5.1	Evaluation of the Camera Control	78
5.2	Creation of the Training and Testing Set	79
5.3	Evaluating the Detector	83
5.4	Tracker	84
5.5	Single Sign Classification	84
5.6	Track Classification	86
5.7	Traffic Scene Interpretation	87
6	Experimental Investigations and Evaluation	89
6.1	Examining Camera Exposure Time	89
6.2	Training and Testing Sets	91
6.3	Traffic Sign Detector	93
6.3.1	Edge Detector	94
6.3.2	Hough Detector	97

6.3.3	Detector Fine Positioning and Verification	100
6.3.4	Detection Verification Fine Tuning	101
6.3.5	Summary of the Detector Results	105
6.4	Tracker	105
6.4.1	Tracker without ego motion	107
6.4.2	Advantage of Ego Motion Information	108
6.4.3	Three Dimensional Pose Estimation	111
6.4.4	Summary of the Three Dimensional Positioning Systems Evaluation	123
6.5	Over all Performance of the Detector and Tracker	123
6.6	Classifier	124
6.6.1	Normalization	125
6.6.2	Single Pattern Classifier	125
6.6.3	Track Classifier	130
6.6.4	Summary of the Classifier Evaluation	133
6.7	Classifier Internationalization	133
6.7.1	Synthetic Generated Sign Samples	134
6.7.2	Sample collection by use of the driver behaviour	137
6.7.3	Management of Multiple Sub-Classifiers for International- ization	139
6.7.4	Results of the complete classification system	144
6.7.5	Summary of the Internationalized Classification System	145
6.8	Supplementary Sign Detection and Recognition	146
6.9	Separation of Active and Exit Signs	150
6.10	Over all Results of the Traffic Sign Recognition System	152
7	Summary and Outlook	157
7.1	Future Work and Outlook	160
A	Appendix	161
A.1	Definitions and Acronyms	161
A.2	AdaBoost	162
A.3	Chamfer Matching	163
A.4	Camera Model	164
A.5	List of References For Colour as Primary Detector	166
A.6	Bayer-Pattern	166
A.7	Camera and Sensor Specifics	167
A.8	Composition of the Classifier Training Set	167
A.9	Composition of the System Evaluation Set	168
A.10	Implementation on Mobile Devices	172
A.11	Generalization of the Coarse Detector	173
A.12	Derivation of Equations in the Three-Dimensional Measurement	174
A.13	Types of Supplementary Signs	175

Contents

A.14 Active Versus Blue Exit Signs Utilizing Colour Information	175
A.15 Screen shot of the Evaluation Tool	176
Bibliography	180

Abstract

Abstract:

In this thesis an automated video based speed limit recognition system, in the following called *SpeedLimitAssistant*, is introduced together with the framework for adaptation of the system to the special characteristics of the signs in different countries. The system is to be used as a vehicle mounted driver assistance system.

Autonomous infrastructure detection has to be international in our globalised world. Many publications concerning the detection and recognition of traffic infrastructure have been published during the term of, and following the PROMETHEUS project 1987 - 1995. Most prominent of the infrastructural parts addressed are lane markings, traffic signs and traffic lights. The literature does not address the challenge of extending the systems to be capable of coping with the - sometimes subtle, sometimes distinct - differences met when considering a system to be designed to recognize infrastructure in a variety of countries instead of just one. In this thesis the traffic sign recognition is given as an example application for the internationalization of an autonomous recognition system. The term *internationalization* is used to express the necessity to adapt the system and especially the classifiers involved to the special characteristics of the traffic signs encountered in different countries. This process of adaptation is supported by the framework developed and implemented in this thesis with the goal of reducing human intervention in this process to a minimum.

The necessity of internationalization is especially true for traffic signs since their representation in different countries is not similar even if the countries belong to the 52 states that signed the Vienna Convention on road traffic from 1968 [[United Nations Economic and Social Council, 1968](#)]. In addition to the internationalization, the necessary and yet in the literature still disregarded extensions to a successful traffic sign recognition will be designed and evaluated. This includes a supplementary sign recognition, a three dimensional position estimation and a scene interpretation. For system training and test a huge number of samples has to be gathered to let the conclusions be significant. To support this task bootstrapping labelling and classifier construction tools have been developed and evaluated.

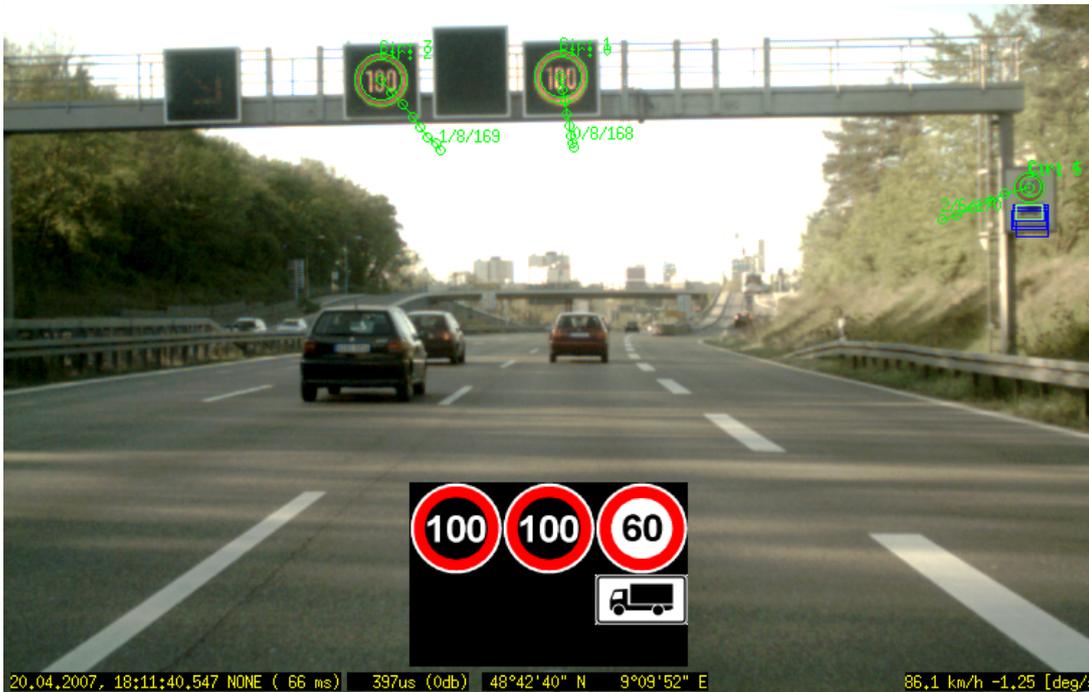


Figure 1: Input image and system output for *Active Signs* and *Supplementary Signs* with timestamp (April 2007), time since last frame (66 ms), exposure time (397 μ s), GPS coordinates, vehicle velocity (86,1 km/h) and vehicle yaw rate (-1.25 Deg/s).

The main contribution of this work to the topic of traffic sign recognition are the following:

- A framework for adapting classifiers on international traffic signs with a minimum of required human interaction.
- The detection and recognition of supplementary signs using a priori knowledge and the classifier internationalization framework.
- A three dimensional scene analysis to enhance the robustness of the system.
- A flexible modular framework that allows traffic sign recognition to be run on general purpose hardware and embedded control units in real time without source code changes.

1 Introduction and State of the Art

1.1 Introduction

Given the objective of enhanced safety in road traffic and drivers comfort through autonomous or assistance systems the recognition and comprehension of traffic infrastructure is one crucial part. Especially traffic signs are valuable references when trying to understand the environment from the position of the driver. They advise the driver of directions, dangers, right of way and provide multiple further information. Many accidents occur due to too high velocity of the traffic participants. In dangerous road sections the authorities have restricted the maximum speed to a save limit. Modern traffic routing has introduced variable speed limitations that react to traffic density, visibility and road condition. These signs as well as temporary signs in construction sites cannot be added to maps due to their temporal short term character. This is why an optical system for recognizing the traffic signs is necessary when not going to the lengths of equipping all signs with transponders. Since the currently existent signposting is made for the human driver to be perceived with his *visual sensors*, namely the eyes for image acquisition and the brain for image interpretation, it is reasonable to use the technical system *camera* as visual sensor to acquire images and the processor to detect and recognize the signs in the autonomous system.

A helpful system for the driver can emphasise the existence of the speed limit via additional optic, haptic or audio indication of the allowed maximum speed. The system could influence the cruise control system in the vehicle, set the limiter of the vehicle autonomously or via acknowledgement by the driver. The introductory functionality of a driver assistant system is a visual reminder of the current limits, thus helping the driver in a complex traffic environment especially when variable limits are installed or the driver is unfamiliar with the region he is travelling in. In this work the focus is on the recognition of *speed limit* signs and the *end of speed limit* signs. The recognition of other signs can be solved analogous and necessary changes will be mentioned at the appropriate location.

Special emphasis will be made on the recognition of traffic signs all over the world. The different appearances of signs from various countries have to be taken into account when looking for a system capable of operating in several countries instead of just one, as it is proposed in most of the literature. In this thesis the main attention is placed upon the 52 states that signed the Vienna Convention on road traffic from 1968 [[United Nations Economic and Social Council, 1968](#)]. Nearly all industrially developed countries signed the treaty, even the United States of America, which is one of the very few countries not abiding the regula-

tions concerning *speed limit* signs. Since the signs in the USA look very different from those in most of the rest in the world the system proposed in this thesis will be based on the detection and recognition of the standardized sign type, but will allow for extra modules to be inserted to attend to the deviations introduced by US-American signs.

1.2 Characterization of Speed Limit Signs in Europe

The traffic signs to be detected and recognized belong to several groups sharing similarities and having certain dissimilarities, both of which are to be described in this section. Some of the differences have to be considered for the design of the detector, others have impact on the implementation of the classifier, while some influence both the detector and the classification system.

The main features are depicted in figure 1.1. In this thesis we concentrate on the detector and classifier for circular signs. For the rare cases of speed limit signs without circular rim, e.g. active speed regulation in the Netherlands, a matched filter algorithm, not elaborated in this thesis, can be used for the detection, while the tracking and recognition part will stay the same as the ones for circular signs. The detection and recognition of rectangular speed limits like in the United States of America are not discussed in this thesis.

There are two types of relevant signs examined in this thesis, the limiting signs and the end of limits. Each of these types has different characterizing attributes like the red rim for limits, which's relative width varies from country to country, or the angle and texture of the black crossbar for end of limit signs.

Another main separation feature is the differentiation between passive, mostly black letters on white ground and active, mostly white letters on black ground, signs.

Passive signs usually consist of sheet metal plates with a printed reflective foil attached on the front side. In rare cases tripods or other constructs holding plastic foil with the sign pattern printed on are used. The foils used vary strongly in reflectivity. The foils for the signs on sheet metal are produced in three qualities of retro reflective capacity, depending on the road type they are placed at. The digits are, with a few exceptions, placed in the centre of the sign. Exceptions are the signs in Denmark, where the additional text *km* is placed beneath the digits, moving the digits upward on the sign. Another exception are older signs, for instance in Germany, where the letters *km* previously placed to the right of the digits are painted over in white, leading to a placement of the digits left of the centre of the sign.

Active signs are powered by electricity and can change the type and value of the limit over time. The sign is either composed of small light bulbs or, in the newer versions, of light emitting diodes. If the exposure time is longer than one activation cycle of the diodes there is no noticeable difference between light bulb

and diode signs. At shorter exposure times the diode signs weaken in brightness in the image, if the exposure time of the camera falls into the gap between two activations of the diode. Since many images of the signs are taken when passing the sign the probability of having taken pictures with diode activation and open shutter is high enough for diode signs not to compose a problem. Especially some rare brands of active signs have different activation cycles for different segments, leading to images where sometimes only the red rim or only the digits or parts of the digits are visible in the image. This problem can be solved by using varying exposure settings or variable apertures. Since this is more a sensor problem than one of the algorithms used for detection and recognition this will not be further elaborated in this thesis. Both the bulbs and the diodes have a directional characteristic allowing them to be seen a far way ahead, while being dark when seeing the signs under an angle, for instance when passing them. This makes it necessary to detect and recognize these signs from farther away than the passive signs.

Characteristics differing for both active and passive signs are the font and size of the digits printed on the sign, especially regarding signs of different countries, see figures 1.3 1.4. The real world size of the signs and the placement of the signs relative to the road vary depending on the road type and the country as well and have to be considered when designing a traffic sign recognition system.

The view of a speed limit sign as shown in figure 1.2 consists of three main features. The *outer* red rim, the *inner* inlay and the digits showing the maximum allowed speed. Later in the dissertation there will be references to *inner* and *outer cutouts* meaning the pixels in the raster image shown in the examples in figure 1.2 as green boxes. For end of limit signs the three defining parts are the outer rim, the crossbar and the digits. Since there is just one circular shape there are just *outer* detections and *cutouts* for end of limit signs.

1.3 State of the Art

The detection of traffic signs using a camera appears in vision literature in the year 1987 [Akatsuka and Imai, 1987]. Consecutively there were many papers concerning the topic in the years, until 1995 most of them connected with the European Project *PROMETHEUS (PrograMme for a European Traffic System of Highest Efficiency and Unprecedented Safety, 1987-1995)*. The algorithms used in that period were based on colour pixel classification and a following shape forming or shape recognition based on a connected component analysis [Mandler and Oberlaender, 1990] as described in the theses of W. Ritter [Ritter, 1996] and S. Estable [Estable, 1996].

Colour is an obvious clue for traffic signs, since the signs were designed to stand out for the human eye and thus catching the attention of the driver. The most popular type of detection still is the segmentation of colour in the image,

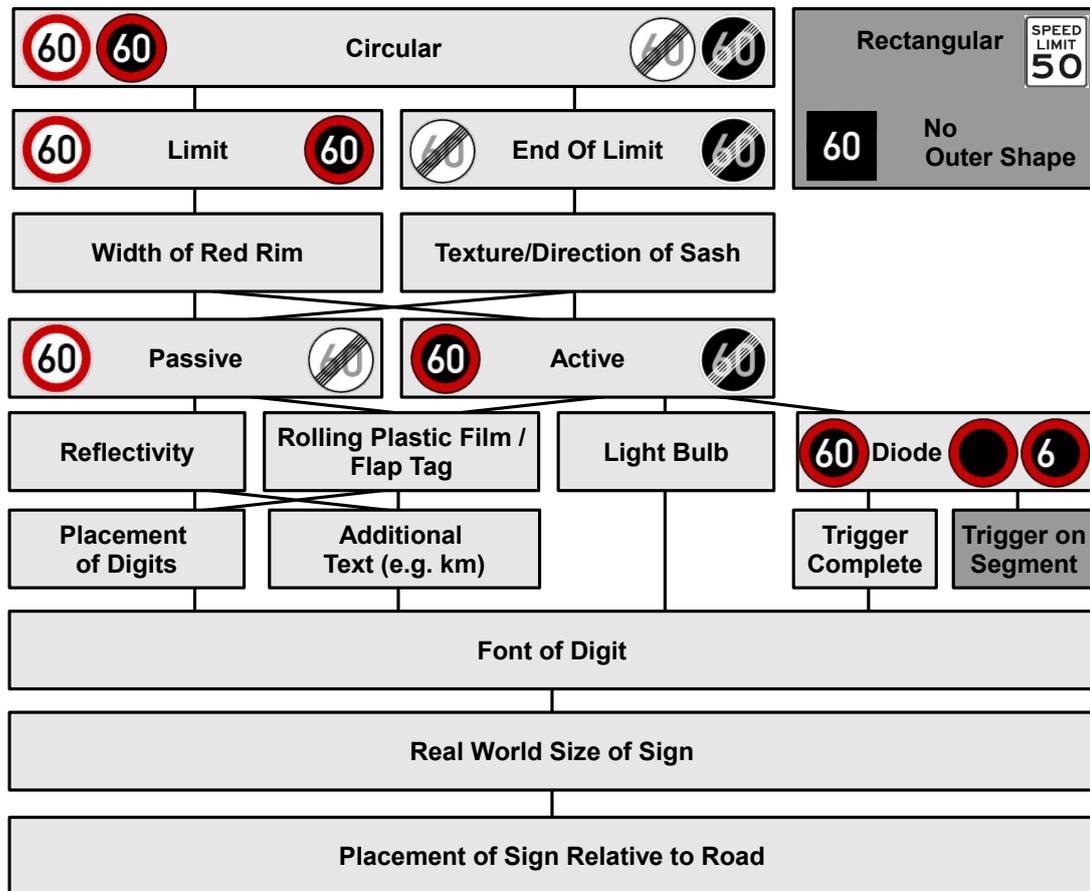


Figure 1.1: Features of traffic signs. Signs possessing features marked in dark grey boxes are not considered in the further examinations. Solutions for the detection and recognition of these signs are given in the appendix, but not part of this thesis.

thus reducing the area to be examined considerably. Most authors address the choice of colour space, such as RGB, CMYK, HSV, HSL and many others. The closing of gaps in the segments and the recognition of the shape in question is another prominent subject worked on, see the appendix section A.5. The colour schemes are not up to the task of detecting the achromatic *end of limit* signs and even many coloured signs have fading colours, which are not detectable by most algorithms. Sometimes even the growth of algae or lichen on the surface of the signs taint the colour of the signs. For the representation in the image coloured light sources at night such as vapour discharge lamps or different types of vehicle headlamps influence the perceived colour on the image. An additional factor for the resulting RGB values on the imager is the white balance, which not only slightly changes with the light intensity, but changes with the ageing of the sensor as well.

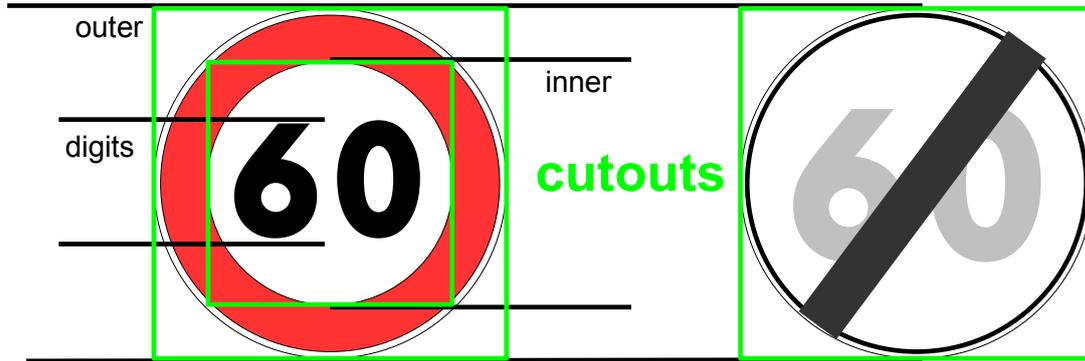


Figure 1.2: Naming conventions for the parts of a traffic sign.

Trackers were not used since follow-up detections were infrequent and due to the very few sign candidates per image, both real and false detections, connecting the detections or predicting their appearance in the next frame to be processed was not of importance, though sometimes used as in [Estable et al., 1994].

The classification of traffic sign patterns was based on direct correlation with the ideal sample pattern or with a limited number of samples collected and labelled, using k-nearest neighbour (kNN) or neural network techniques on the limited data sets (figure 1.5). Large sets of data were hard to be obtained due to the limited availability of frame grabbers and the comparatively low capacity of hard drives. Publications proposing this type classifier setup are [Akatsuka and Imai, 1987], [Ritter, 1992], [Kehtarnavaz et al., 1993], [Piccoli et al., 1994], [Wei, 1994], [Zheng et al., 1994], [Ritter et al., 1995a], [Ritter et al., 1995b] and [Murino et al., 1995].

Following the Prometheus project and with more computing power available the scope of work moved from Detector-Classifer systems to systems detecting, tracking the sign candidates in consecutive images and classifying the candidates. Some authors, notably [Gavrila, 1998, Kressel et al., 1999] were moving to grey value detection of the signs thus avoiding the major drawbacks mentioned above. Sample sets still remained small even though frame grabbers were getting available and storage capacities were growing. In addition to the classifiers used in the beginning of the 1990's radial basis classifiers [Powell, 1992] were used by [Gavrila, 1998, Kressel et al., 1999] as well as polynomial classifiers in classifier hierarchies [Kressel et al., 1999].

The newest detection algorithms are the fast radial symmetry detector introduced by [Loy and Barnes, 2004] and AdaBoost, see description in the section A.2, as presented by [Viola and Jones, 2001] and used by [Bahlmann et al., 2005], [Keller et al., 2008].

In the 2000's trackers are widely in use, but the sample sets stayed at the level reached at the end of the nineties, except for [Escalera and Radeva, 2004], where in addition to one ideal sample per class distortions and noise were added

AT	BE	CH	DE	DK	ES	FR	IT	NL
								
								
								
								
								
								
								
								
								
								
								
								
								
								
								

Figure 1.3: Variety of restriction signs

AT	BE	CH	DE	DK	ES	FR	IT	NL

Figure 1.4: Variety of end of restriction signs

to the ideal sample to allow the classifiers to better adapt to realistic scenes. Additional classification schemes for the traffic sign recognition were introduced. Among others a Fisher Discriminant Analysis ([Fisher, 1936], [Duda et al., 2000]) followed by a Bayesian Network were used by [Bahlmann et al., 2005] and [Keller et al., 2008].

The algorithms mentioned above yield information about the processing chain from an image to the recognition of a single real world traffic sign, but the meaning of a traffic sign is influenced by additional factors. One factor influencing the relevance of the signs is their placement at the street, e.g. validity for certain lanes only. Another important information source are the supplementary signs, which constrain the validity of the sign to, for instance a certain time of day, certain weather conditions or some vehicle types and weights.

Most papers use daytime sequences with medium to good lighting conditions, leaving out dense rain and especially night time, where additional problems for the detection and recognition of traffic signs arise, for example motion blur due to long exposure time or colour changes due to active lighting. In the literature the feasibility of the algorithmic approaches were in the centre of the research. In this thesis in addition to the feasibility the robustness of the algorithm as well as the possibility to implement the system on a hardware being affordable and efficient enough for series production is investigated. If the system is to be used in a commercial driver assistance system, the power consumption has to be reduced to limit the heat generation of the system. Not the least factors are the price and availability of the necessary hardware. Thus algorithms having very high computational demands, requiring large amounts of memory or having the necessity of specialized hardware for their processing chain were not considered. In the literature there is no mentioning of the flexibility of the algorithms where the internationalization of the algorithms is concerned. For a traffic sign recognition system at least the signs in the 52 countries signing the Vienna Convention [United Nations Economic and Social Council, 1968] should be covered by the detection and recognition system. Since there are additional rules influencing the valid velocity or other traffic regulations in most countries these have to be adapted to the laws of the land in question, while the image processing module should still be based on the same algorithms.

1.4 Traffic Sign Recognition Research Groups

In figure 1.5 prominent research and development groups working on traffic signs recognition and were influencing the whole community are referred to. In the years up to 1995 the Prometheus project was the core around which most activities were clustered. Paclik [Somol et al., 1999] tried a different approach still relying on colour as the main cue, but expanding the algorithm to allow for grey value sign classification as well. In 2002 Barnes et al. ([Loy and Barnes, 2004]) pub-



Figure 1.5: Time line of publications on traffic sign recognition

lished a method for fast grey value based detection of traffic signs that was used by other groups as well. The methods currently most referenced are the boosting algorithms ([Viola and Jones, 2001]) first applied on traffic signs by Bahlmann et al. ([Bahlmann et al., 2005]).

The results given in the literature for the performance of the detection and classification systems depend on their respective learn and test sets, so the numbers give only a rough estimate of the algorithms comparative quality.

1.4.1 PROMETHEUS Groups

The work in the Prometheus project was mainly based on colour detection and the classification of the detected objects. The specification for the project included the preparation of demonstration vehicles which apply the developed algorithms online. To be able to fulfil this task the most advanced mobile computer hardware available at the time was installed in the vehicles, even if the hardware filled the whole luggage space of the vehicles. Due to the low speed, compared with today's resources, of computers at the time the tracking was rudimentary [Estable et al., 1994] or completely set aside by the authors since the signs were seen by the system only once or twice in a standard passing of a road sign. Limited storing capability available at the time led to comparatively small sample sets, which usually were well below 1000 samples split on the classes to be recognized. This fact influenced the choice of classifiers used.

The main challenge in the detection step was to perform a powerful and robust colour segmentation of the signs. Different colour spaces were investigated allowing even slightly bleached signs to be at least partially segmented. Varying

lighting conditions such as rising and setting sun, street lamps, and other influences led to attempts of using colour consistency approaches, usually looking at the pavement for a clue to what the current colour deviation is to be able to normalize the colour appearance. Bleached signs were usually disregarded as well as *end of limit* signs were usually ignored, since due to their being nearly or completely monochrome the colour detection would not work.

Focussing the detector on colour features induced an additional field of work, namely searching for ways to describe the shape of signs in a pixel segmented image. The first step was to assign each pixel a colour class, some allowing fuzzy membership values, then doing morphological adjustments and after that combining the created segments to traffic sign forms via **B**inary **C**onected **C**omponents, short BCC, or **C**olour **C**onected **C**omponents, short CCC ([Zuniga et al., 1982], [Mandler and Oberlaender, 1990]). In addition to the problem of not finding a sign in an image due to missing colour cues there are situations in which the detector finds too many segments for the shape finder to match in the given time.

Especially Y-J. Zheng, W Ritter, R Janssen added a verification step based on normalized colour to ensure the sign candidate [Zheng et al., 1994] in a single frame. The classification of the image object *sign* was, partly due to the limited sample sets for classifier adaptation, limited to nearest neighbour classifiers and rudimentary neural networks. Internationalization was not an issue in these early steps of traffic sign recognition, though there were some groups, notably at the Daimler Benz Research Institute detecting and recognizing German and French signs using a single k-nearest neighbour classifier [Zheng et al., 1994]. The term *internationalization* is used to describe the algorithms necessary for the adaptation to the different characteristics of traffic signs encountered in varying countries. Representatives using the colour detector type in the traffic sign recognition system are pointed out in the following:

1. Daimler Benz Research Institute, being the first to include a rudimentary tracker [Estable et al., 1994], a colour based shape verification [Zheng et al., 1994] and a very fast polynomial classifier and lookup table based colour pixel detector [Bartneck and Ritter, 1992], allowing about 200ms processing time on a four CPU 40 MHz transputer system. The thesis of Werner Ritter [Ritter, 1996] gives a good overview of the algorithm proposed by the Daimler Benz research team, as does the thesis of Stephane Estable [Estable, 1996]. The advantage of the colour based system presented by the Daimler Benz Research Institute is a very fast detection of traffic signs allowing for the first so called *in-step* system, meaning that the system is capable of recognizing traffic signs on the given hardware online in a vehicle moving in regular traffic. The system was presented at the final presentation of the Prometheus project 1994 in Paris, recognizing coloured round, triangular, rectangular and octagonal signs of up to 60 different

classes using a *k-Nearest Neighbour* network.

2. University of Koblenz-Landau introducing the Colour Structure Code as a fast (800ms per frame on a 4 CPU 40 MHz Transputer system) detector for colour shapes [Priese et al., 1993], [Priese et al., 1994], [Rehrmann et al., 1995]. The system was developed as an alternative detector for the Daimler Benz system.
3. N. Kehtarnavaz [Kehtarnavaz and Ahmad, 1995] at the A&M University of Texas used a neural network for binary colour segmentation in YIQ colour space. The binary connected components are transformed to a log-polar representation which is then Fourier transformed. This new representation of the data is fed to a back propagation network that decides for the shape type the of the sign. The system was used to find stop signs and yield signs where the shape defines the meaning. The computation time is stated as 80 seconds on a Sun Sparc Server 1000 having a 40 MHz CPU. Special focus was set on the different types of distortions a sign might encounter when represented in an image. Six types of noise were distinguished:

centroid noise	error in object centre leading to a translation of the pictogram pattern
occlusion noise	occlusion of the sign by another object or a partly coverage of the sign (dirt, wear & tear)
Gaussian noise	distortions occurring due to dirt on the lens, bad weather or similar factors
motion noise	due to camera movement and long shutter times
shear noise	distortions through view angle
maximum distance	scaling noise of the object

Adding the idea of looking at the distortions opened the path for the use of *Synthetic Samples* for the adaptation of the traffic sign classifier as described in section 4.2.2.

4. At the Universidad Carlos III de Madrid [de la Escalera and Moreno, 1997] a rudimentary colour detection via RGB thresholding and a corner detector with separated masks on the binary image are used for detecting the signs. The more interesting part is the sample set used for the adaptation of the classifier used for the pictogram recognition which is composed of ideal sign representations which are rotated $\pm 6^\circ$, modified by three different types of Gaussian noise and translated ± 3 pixel in both directions, thus creating 1620 templates sized 30x30 pixel from one ideal sample. The sample set is fed into the training of a back propagation neural network. The processing duration on one 256x256 pixel frame was given as 1440 ms on a Intel 486 with 33MHz. The algorithm was used on 9 different triangular sign classes and 9 different circular sign classes.

Colour based systems have the advantage of allowing a fast segmentation of the image for all different physical forms of signs. The differentiation between the different contours takes place in a follow up step and thus on a much smaller amount of input data.

1.4.2 Radial Symmetry Detector

In the year 2002 Loy and Zelinsky from the National Australian University proposed the Fast Radial Symmetry Transform [Loy and Zelinsky, 2002, Loy and Zelinsky, 2003] as a detector for circles and symmetrical polygons. The results are very promising and led to the traffic sign recognition proposed in [Barnes and Zelinsky, 2004, Loy and Barnes, 2004]. The emphasis is on the detection part, while tracker and recognition are rudimentary. Connecting elements closer than 20 pixel to one of nearly equal size in previous frame is used as tracker algorithm and thresholding on a cross correlation coefficient as classification step. The tests were done on a comparatively small set of images having 1107 frames, meaning 55 seconds of video at 20 Hz. The detector yielded 90% correct detections and 10% false positives on this set. There were 152 detected real world objects in all, with an unspecified number of misses. The speed on an - in the year 2004 - up to date computer was 20 Hz on a QVGA (320x240 pixel) image. Since the authors concentrated on the detector there is no mention of the follow up chain like the decision unit or internationalization of the system.

1.4.3 AdaBoost Approach

In the papers [Bahlmann et al., 2005, Keller et al., 2008] the use of the AdaBoost was proposed, see the appendix section A.2, algorithm as presented by Viola and Jones 2001 [Viola and Jones, 2001] for the detection of traffic signs. Additional Haar wavelet [Haar, 1910] features based on colour cues were added, reducing the false positive detections by the order of one magnitude. The detection module is followed by a histogram based brightness normalization for the detected regions and scaling module to a standardized size for these regions in the raster image.

The classifier used to discriminate the different sign classes is based on multivariate Gaussian distributions used on the top 25 linear discriminant analysis (LDA) features of the normalized samples. The system is based on the analysis of the whole track of signs detected in a passing by considering the weighted sum of single classification results, the weight being the higher, the newer the sample is in the track. Thirty minutes of video including 4000 circular sign representations of 23 different classes were used as training set. The detector yielded 98.6% of the signs while having 0.03%, or 0.3% using grey features only, false positives. Since the detector kernel is used in varying scales at an unmentioned raster the false positive number does not allow to conclude how many false positive candidates were detected per frame. The classification scheme rejected 15% of the positive

samples and had an additional 6% error rate between different sign classes. Every 600 frames a false positive was registered, this meaning one every minute at the given frame rate of 10Hz.

The computational demands are very high, allowing for a frame rate of only 10Hz on a 2.8GHz Xeon CPU. Since the AdaBoost detector system is based on the signs inlay information, here the numbers in the sign, indicating the speed limit, as well as on the outline of the sign, it is expected to have less detection performance when a larger variety of signs are encountered, especially with putting attention to internationalization. The drawback of this solution is the fact that the complete icon, meaning representation of the sign in the image, is trained in the adaptation phase of the classifier. Thus forcing a training on the complete set of all countries' sign representations and a time consuming retraining whenever a sign type of a new country is to be added to the operational area or a new sign type is added to the required signs set. The advantage of this solution is the universality of the approach. Given enough computation time this algorithm is capable of detecting and recognizing all types of traffic signs.

1.5 Scope of the Thesis

As shown above there are many recognition system approaches for traffic signs. Most of them are based on colour, which, as explained in the previous sections, will not solve the problem of finding the black and white *end of limit* signs, cope with bleached signs or lighting conditions influenced by active lighting. The learning and testing image sequences described in the papers mentioned were usually comparatively small compared with the huge diversity of traffic situations encountered, such as daylight, darkness, tunnels, bleached or overgrown signs, vehicles blocking the line of sight and many more. None of the papers above has mentioned the problems when not only building a detector and classifier for one chosen country, but many - with the added necessity for generalization capabilities of the system as well as the testing required that are implied by this step.

The main contributions of this work to the subject of traffic sign recognition are a reliable, modular and fast detection and recognition framework. In this dissertation the set of signs is expanded to include supplementary signs, see section 1.5.2. Furthermore for the first time the internationalization of the classifiers is addressed in depth in this work.

To be able to deal with the huge sets of data that have to be reviewed to get a solid evaluation of the algorithms in a real world environment, a complete tool chain with semi automatic labelling processes and automatic evaluation criteria has to be developed and implemented. Another area left untouched in the literature are algorithms for the detection and recognition of supplementary signs which are necessary to interpret the meaning of the associated signs. In the

following chapters the work done on the design and implementation of an adaptive traffic sign recognition system, capable of being adapted to the variations encountered in different countries, is illustrated.

1.5.1 Modular and Adaptable Traffic Sign Recognition System

The framework proposed in this thesis is designed and implemented for the detection and recognition of *Speed Limit* signs together with the necessary auxiliary modules. The system is based on grey value image interpretation and implemented in a fashion that allows to integrate further algorithms for shape detectors or colour detection methods, thus the system represents a flexible generic framework for the detection and recognition of traffic infrastructure. Thus the systems design is capable of working on all kinds of traffic infrastructure and was used on traffic lights, *stop signs* and triangular traffic signs as well, see the appendix section A.11 and [Lindner et al., 2004].

1.5.2 Supplementary Signs

The validity of a traffic sign can be altered by a supplementary sign. Since an autonomous system should be able to integrate this knowledge into its decision, the additional signs have to be recognized as well. There are constraining and explaining supplementary signs. For example "for lorries only" being constraining, while "curve ahead" is an explanatory sign. For the relevance of the sign the constraining signs are of importance only. For autonomous driving the hint for a curve ahead or the high probability of animals on the road might be important information for the inclusion of a priori knowledge in lane detection or obstacle detection algorithms. Since we are looking at a "Speed Limit Assistant" we will restrict the algorithms to work on the constraining supplementary signs.

The presented algorithm fits well into the modular concept of the described traffic sign recognition framework. The algorithm is fast and solves the detection and recognition task in a hierarchical approach presented first in this thesis.

The auxiliary system for the detection and recognition of supplementary signs will have to be easily adaptable to the conditions of different countries and low in hardware requirements, since it will have to run in addition to the original traffic sign recognition and thus only part of the computational power will be accessible. Since the supplementary information always concurs with a *speed limit* sign or *end of limit* the supplementary sign detection module will have information on where and in which size range to search for the supplementary signs. The position, size and pictogram of supplementary signs are not as well standardized as they are for the limit signs. This results in the countries sign specifics having far more impact on the location and appearance than they have for the detection and recognition system for the circular signs, meaning additional care has to be taken in the flexibility of the algorithm.

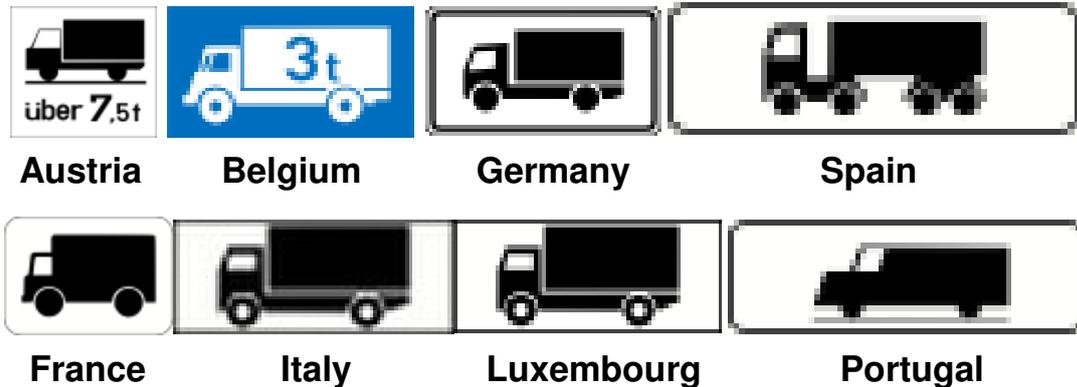


Figure 1.6: Representations of For Lorries Only supplementary signs in Europe

1.5.3 Internationalization and Classifier from Scratch

In the literature concerning traffic sign recognition very little is said about how to handle the differences between signs in different countries. Since most detectors from the literature - with the exception of the AdaBoost algorithm - are independent of the inner layout of the signs, the differences have to be regarded in the classifier only. Only very few references are to be found about the algorithms working in multiple countries.

There are three ways to cope with the classifier systems setup:

- Having one classifier covering all countries.
- Having one classifier per country.
- A mixture of the two above.

Of course the classifier used could be general enough to be able to cope with the differences encountered. This usually either dilutes the rejection capabilities of the classifier, allowing for more false positives or it reduces the rate of signs correctly classified, thus enlarging the false negative rate. For this reason an alternative way is depicted in this thesis.

There are different ways of deciding in which country the system is currently used.

- By an additional sensor, namely GPS combined with a map.
- By discovering it by itself through classifying the detected signs with a classifier dedicated to the task of discerning the country by their appearance.
- By using multiple classifiers, which are dedicated to the classification of traffic signs in one country each and detecting the current country by deciding which of the single classifiers performs best and thus has most likely been trained on the signs of this country.

When the country the vehicle is currently driving in is ensured, separate classifiers previously adapted to the respective countries traffic signs are used, which are more specific, hence allowing better classification results. The drawback is the much higher adaptation cost and the very high consumption of persistent memory in the control unit or computer.

The third possibility is a combination of the two approaches above. Allowing one classifier to cover more than one country, but still having multiple classifiers allowing for countries with very different sign appearances.

In this thesis all three approaches for using a classification system in multiple countries were tested. The third scheme, namely using the same classifier for groups of countries, has been identified as the best trade off. A new scheme for the classification process has been implemented in using a modified version of the third scheme.

All the classifier adaptation strategies need sample sets from all countries the system should employable in. There are multiple ways to obtain the necessary samples.

- Gathering data in the respective countries and labelling it.
- Take model signs from the rule books and distort them, creating what in the following will be called *synthetic signs*.
- A mixture of the two approaches.

Gathering samples is a strenuous task, especially when many countries are involved, labelling is even more so. Once there is a functioning system available this system can be used as bootstrapping device for gathering more data, in the sense of supervised or unsupervised learning, with or without a human in the loop for verifying the labels given to the object. This eases the task of labelling considerably. The learning set, enlarged by the new samples, is used to adapt a new classifier which is better adapted for the classification of the traffic sign. To adapt a first complete classifier system for a given country the *synthetic signs* can be used to build a first version of a workable classifier. The distortions used on the pictures are derived from the distortions encountered in the countries already adapted.

N. Kehtarnavaz [Kehtarnavaz and Ahmad, 1995] writes about the different factors influencing the appearance of a traffic sign in an image. The parameters were manually adapted for the different distortions and the algorithm was used to adapt a classifier from sign prototypes. Though it is not mentioned in the paper this can of course be expanded to be used for the adaptation to different countries sign layouts. The drawbacks are that the parameters are guessed and not verified by a sample set, which does not perform as well as the strategy proposed in this thesis. A newer work from the Nagoya University [Ishida et al., 2007] gives an algorithm for rotation, blur and translation of the sign representatives in images to

be learned from a previously gathered sample set. The paper by Ishida describes a method close to the new approach proposed in this dissertation and which was published by the author of this thesis and colleagues in [Hoessler et al., 2007] on the subject of generating a new sample image set for classifier adaptation and will be described later in this work.

Some authors claim that their system would allow for different countries differences to be compensated by their recognition algorithm. At the "Ecole Nationale Superieure des Mines de Paris" some work has been done to eliminate most differences between the countries sign appearances by segmenting the single numbers in a binarized version of the cut-out holding the signs centre and using optical character recognition to recognize the meaning of the sign [Moutarde et al., 2007]. The paper states that French signs were recognizable in about 90% of the cases and most of the false negatives were due to the numbers not being separable from each other. The results given were achieved on a set of 281 signs and are to be interpreted in relation to a test set of this size. The system trained on french signs was tested on few German signs and was working there as well, without further information for the tests in Germany being given in the paper. The problem of separability of the numbers on the sign is a limiting factor in this strategy, allowing not more than 90% of the signs to be recognized in this manner.



Figure 1.7: Representations of Speed Limit/End 40 km/h signs in Europe

1.5.4 Evaluation of a Complex System

Most evaluations are based on a small set of test samples and the score is either based on the detection rate or the classification of single signs. In some papers the classification result for a whole track is taken into account, thus representing the real world traffic sign object. In this dissertation in addition to detection rate and classification rate based on the correctness of the real world traffic sign object there will be an additional step, namely the scene evaluation.

Especially on motorways the signs are placed on both sides of the road or in rows of up to one sign per lane above the road. Some of the signs are bound to be far away from the current position of the vehicle. For the interpretation of the current traffic scene the correct classification of one of those signs usually is sufficient. The complete image sequence in which the allowed speed is to be

determined and the detection systems results can be matched against the labelled ground truth in three ways. Firstly per event, counting the limiting signs and events resulting in a change of the allowed maximum speed, secondly per time slice or per kilometre travelled with the correct speed limit being shown to the driver. In this dissertation a scheme for evaluating and optimizing of the whole traffic sign system is proposed and has been implemented and tested. For this type of evaluation elaborate tools and different views on the recognition and detection data have been developed and analysed.

1.6 Thesis Overview

The description of the algorithm presented in the thesis begins in chapter 2 with an overview of the requirements of the traffic sign recognition system and a description of the algorithm for the detection and recognition of traffic signs. This includes the detection 2.4, detection verification subsec:RuntimeDetectionVerification and tracking 2.5 of circular sign candidates. The three dimensional positioning of the circles tracked in subsequent images in the real world relative to the camera position is explained in section 2.5.3. In the following classification of the circular image candidates 2.6 and connecting the classification results for the decision on the traffic sign class of a real world object 2.6.3 is explained. At the end of chapter 2 necessary modules are explained, most important the supplementary sign detection and recognition module 2.7.

In chapter 4 the offline system necessary for collecting training samples and adapting the classifiers introduced in section 2.6 are explained. In this chapter the algorithms developed for allowing the expansion of the field of application of the traffic sign recognition system from one country to a multitude of different countries are presented.

The chapter 5 gives short overview on how the system is to be evaluated and the obtainment of the necessary evaluation sets.

Chapter 6 contains the results of the evaluations made on the detection and recognition system. This includes considerations concerning the camera settings 6.1, the adjustment of parameter settings in all modules based on the optimization of example sets (6.3, 6.3.3, 6.4) and the evaluation of the performance of the modules in the detector, tracker and three dimensional positioning systems (6.3.5, 6.5, 6.4.4). In the second part of chapter 6 the performance of the classification systems and their expansion to ten European countries is evaluated 6.6. This includes the evaluation of the algorithm for the creation of realistic synthetic training samples 6.7.1, the adjustment of the number of necessary classifiers 6.7.3, as well as the detection and classification performance of supplementary signs 6.8. The results of the final traffic sign recognition system on a per kilometre base using a traffic scene interpretation in the ten countries are given 6.10.

The chapters 7 and A hold the summary and the appendix, respectively.

2 Runtime System Design - and Implementation

2.1 General Requirements

A traffic sign recognition system to be introduced to modern vehicles has to comply to strict constraints. It has to work day and night, at high vehicle speeds, be feasible in small control units having a low power consumption and using memory sparingly. It has to cope with limited distortions of the signs in the image encountered due to imperfect sign installation in the real world and perspective deformations, as well as with defects and weathering and fading of the signs representation in the image. The system has to be able to detect and recognize traffic signs even when they are occluded by other traffic participants, the windscreen wiper or bushes for some part of the period the sign would normally be visible for the camera. This precondition leads to the necessity of detecting a wide variety of sign sizes in the image at a high frame rate, since it can not be guaranteed that the sign will be visible while still being far away and thus having a low resolution in the image or still visible when being close and thus in high resolution.

The installation sites of traffic signs vary widely as there are signs in construction sites close to the road surface and others installed at bridges crossing the road above the passing vehicle. The signs may be passed at low distance or as far as several lanes away for multi lane highways. Thus the area in the image for the detection of traffic signs is not easy to be reduced. In figure 6.3 a data base of over nine hundred thousand signs' circles are analysed. Each centre of a traffic sign in a frame increments the value in the figure by one.

The set of signs to be recognized is limited to the set mentioned in the Introduction section 1.1, thus the speed limits, *non overtaking* signs and ends thereof. Most triangular warning signs have immediate effect and thus have to either execute a direct influence of the system on the vehicle or warn the driver which may distract him in the timespan where he should watch the road most intently due to the immanent danger. Speed limits and *non overtaking* signs as well as their cancellations influence long stretches of road. This allows for the implementation of a reminder function without direct distraction of the driver or a slight and easily over-steerable influence on the vehicle. This could be a gradual adaptation of the cruise control or a slight moment on the steering wheel for instance when using the turn signal at high speed in an *non overtaking* zone.

The often mentioned detection and recognition of Stop signs is problematic since in western Europe most traffic lights at major crossings have stop signs as fall-back stage in case of power failure or in low traffic periods. Since stop signs as do *right of way* signs and yield signs presuppose direct action there would have

to be clear warnings or even engagement in the system. The uncertainty imposed by the existence of traffic lights at the scene would require a recognition of the traffic lights and their current status as well.

This work will concentrate on designing and realizing a flexible framework for the detection and recognition of *Speed Limit* signs. The flexibility and modularity allows the simple integration of additional modules for the detection and recognition of further object types. For the system to be easily ported to mobile devices or different control units and keeping the power consumption down at acceptable rates only algorithms not necessitating specialized hardware (FPGA, ASIC, customized DSP or similar) will be taken into account.

2.2 System Overview

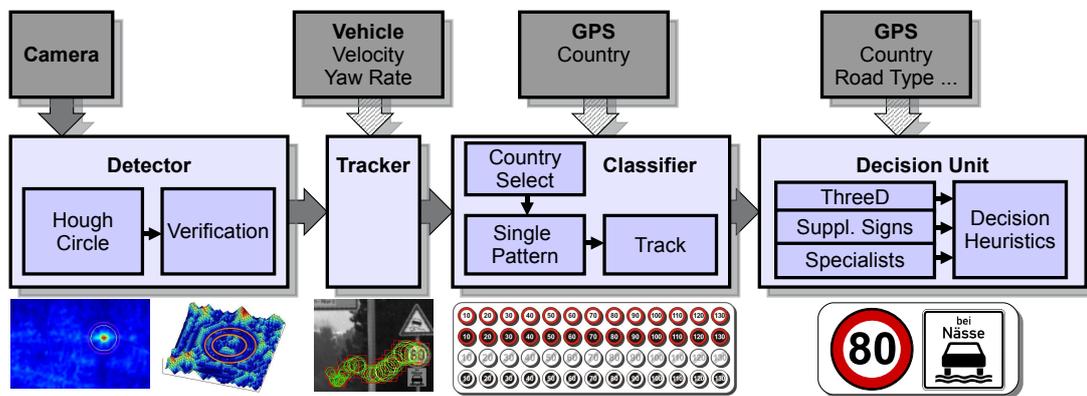


Figure 2.1: System overview with main modules

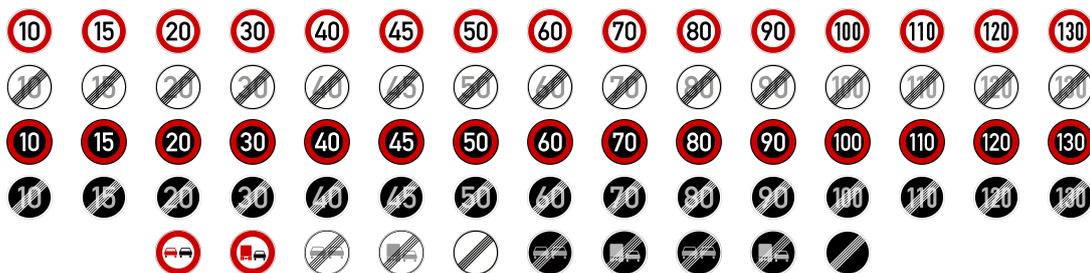


Figure 2.2: Signs to be detected and classified by the system

The overview of the system and the types of traffic signs to be recognized are shown in figures 2.1,2.2. The first input are sequences of images of sufficient resolution and luminance dynamics, acquired by a camera mounted in the vehicle. After obtaining the image a detector, in our case a detector for circles, is used to reduce the number of candidates for signs in the image. Even general classifiers

as AdaBoost are used as detectors only ([Bahlmann et al., 2005]). Following the primary detector is an optional position and size refinement step. The tracker combines traffic sign shapes previously sighted in the images, to tracks over the sequence of input images. A classifier validates the single objects and then the classification results for one tracked object are combined to a result for the probability of there being a real world traffic sign. The last step in the system is the scene evaluation, which might include additional modules like supplementary sign detection and recognition, three dimensional measurement as well as minor modules solving problematic cases, e.g. the separation of active signs versus minimum speed signs for example, (figure 2.18).

2.3 Camera Requirements

The camera obviously has to be installed forward looking, since the signs are readable in this direction only. To allow for cleaning and better viewing angle the camera usually is installed behind the windscreen, at the highest point accessible with respect to installation preconditions and screen cleansing purposes (wiped region). The sensor has to have a reasonably high dynamic luminance range to allow for bright and dark regions in the image, the sun in the centre view and the signs without illumination at the sides for example. In addition the signal to noise ratio has to be high to permit night time use of the system. Long exposure times at low lighting conditions are limited due to the resulting motion blur. This factor currently limits the resolution of the sensor to about VGA (640x480 pixel) to XGA (1024x768 pixel) even at low apertures. Another factor is the minimum light exposure of the camera since the sun reflecting in a traffic sign can lead to saturation of the sensor making the sign indecipherable in the image. The field of view is dictated by the regions where signs may occur in road architecture and the relative position of the vehicle.

If the camera is to be used by more than just the traffic sign recognition system there has to be trade-off between the different users. Possible additional algorithms include lane departure functions looking at the road directly in front of the vehicle, traffic light detection looking skyward as far as possible and smart cruise control systems in the vehicle, needing high resolution in the centre of the image to measure the distance to vehicles at high distances.

The main two types of imaging sensors available are CCD and CMOS imagers. Both types allow for the requirements above. The CCD sensors usually yield a better signal to noise ratio at low light conditions and have a much lower fixed pattern noise compared to the CMOS sensors, while the CMOS sensors allow for multiple different slopes in the characteristic curve, derived by performing multiple exposures and thus nominal higher dynamics.

The camera with which the tests in this thesis were executed has the following features as listed in the appendix section A.7. These are a CMOS imager with

a 752 x 480 pixel resolution and a 10 bit grey value depth, an additional *Bayer-Pattern* [Lukin and Kubasov, 2004] was applied to allow colour comparison and a lens with a 7mm focal length and an approximate aperture of 2.0 . This reference camera will be used in all further examinations, unless noted otherwise.

For the transformation from real world to image the pinhole camera model was used, allowing for principal point (u_v, v_v) and focal length only. The local distortions for all cameras tested were low enough not to interfere with the detector and classifier, so no rectification was necessary. For the reference camera, as described in section A.7, the radial and skew distortions as explained in the appendix section A.4 were small enough to be disregarded even for the three dimensional reconstruction used in section 2.5.3. The translations of the sensor relative to the vehicle were determined by the use of a laser distance meter. The roll part of the rotation matrix (\mathbf{R}_x) was adjusted to be close to zero, while yaw (\mathbf{R}_z) and pitch(\mathbf{R}_y) were determined by finding the vanishing point.

2.4 Traffic Sign Detector

The detection step is using the most computational time of the complete system. Due to the given processing power some algorithms were not suitable for the task at hand, for example correlation over the whole set of sizes and sign types in the entire image or using AdaBoost. Colour as primary detector was ruled out due to the necessity of detecting achromatic signs. The detector using *Chamfer* distance as proposed by Gavrilu [Gavrila, 1998] yields results close to the ones derived by the algorithm proposed in his thesis, but is challenged when there is a partial occlusion of the circle, leading to very high distance values, meaning low detection scores, in the distance transform, see section A.3. The best trade off between allowed computation time and quality of the result is the proposed type of the fast *Hough Transform*, detecting the circular outline of the signs.

The maximum sizes of circles to be detected in the images can be derived from the cameras field of view \mathbf{fov} in radians, the pixel width of the sensor w , the real world size of the signs s in metres and the minimum passing distance d of the vehicle in metres. The resulting formula for the maximum size in pixels is given below.

$$p = \frac{w}{2} - \frac{w}{\mathbf{fov}} \cdot \arctan \left(\frac{d - s/2}{d + s/2} \cdot \tan \left(\frac{\mathbf{fov}}{2} \right) \right) \approx \frac{w \cdot s}{2d + s} \quad (2.1)$$

The variables with the exclusion of the sizes of the signs are constant for all countries. The algorithm can be adjusted for each country depending on the sign sizes commonly used in their traffic infrastructure. For Germany this leads to a maximum sign diameter of about 60 pixels, given a sign diameter of 0.6 metres on rural roads according to the rule book for traffic sign placement [Bald and Giesa, 2002] and a lateral distance of about 3 metres, on higher level roads the signs are larger, but the minimal lateral distance to the camera increases as well. The smallest

size of signs to be detected in the image is derived from the minimum resolution necessary to be able to classify the circle for being a sign and the type of the sign. This leads to the minimum size of 15 pixels in diameter. For the distribution of traffic sign sizes in the image see figure 6.2.

For reasons of speed-up and memory economy the primary detection in the image is run on a resolution pyramid, starting on a sub scaled by the factor of two for the primary detection algorithm. For the classification the sign candidate in the image should be segmented as accurately as possible, thus there is a second detection step verifying the detection and fine positioning the centre as well as determining the exact radius of the sign. Since the limiting signs consist of two concentric circles the verification and fine positioning stage is used to split the detection to deliver both radii.

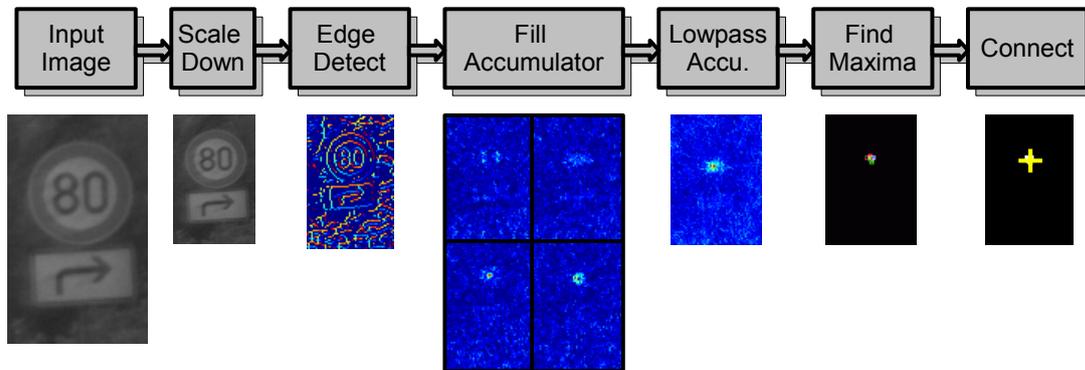


Figure 2.3: Scheme of the *Coarse Detector*

2.4.1 Coarse Detector

The primary detection of the circles in the images is performed on an image sub scaled by the factor of two. The factor is derived by the minimum diameter of a sign to be recognizable, which is about fifteen pixels and the smallest radius of three pixels for the detector to deliver a verifiable peak in the accumulator. Thus when a sign of radius seven pixels has to be detected, to allow for far off signs to be found before leaving the field of vision and to have a higher chance to detect and track the signs in the image in case they are obstructed by other vehicles or infrastructure in the following frames, the maximum factor for scaling the input image down for optimizing computation time is two. Another beneficial factor is that by the scale of two eventual adverse effects from camera sensors employing *Bayer-Pattern* filters, see the appendix section A.6, are countermanded.

Edge Detection

When using the fast *Hough Transform* the first step is to determine the position and phase of the gradients of the object in the image. For the detection of the edges an algorithm offering a sufficient precision in the phase gradient while being computational efficient was to be found. The Laplace filter does not allow for detecting the phase, Roberts Cross delivers too much noise in the resulting phase and Kirsch generates phase information in four directions only. Edge detectors examined more closely were Sobel filter, Prewitt operator and Canny operator, where the Sobel filter provided the best performance in speed as well as detection rate and thus is used in the system.

To ease the computational cost in later steps of the algorithm and to better define the position of the edges a non-maximum suppression scheme is used.

Since a large part of the image has to be processed this part of the algorithm has to be optimized for computation time very thoroughly.

Hough Accumulator

The main detection step is established by the use of the fast generalized *Hough Transform* [Duda and Hart, 1972, Li et al., 1986, Illingworth and Kittler, 1987]. To reduce memory consumption and computation time the radius range and centre positions to be scanned is quantized. Due to the previously executed computation steps the existence of an edge and the direction thereof at all image points is known at this point of the system. If a single pixel belongs to the rim of a circle the centre of this circle has to lie on a line either in the direction of the gradient or in the opposite direction. The expected radius defines the distance from the edge point where the accumulator is incremented.

Low Pass Filter on Accumulator and Maximum Extraction

In this step the highest entries of the accumulator image are extracted and entered into a list structure. A low pass filter is used on the accumulator images and a fixed number of maxima extracted and passed on to the next step. The list of points is submitted to a connected components search. The weighted centre of the connected components comprise the candidates handed on to the detection verification stage.

2.4.2 Detection Verification

The circle candidates detected in the previous steps are quantized to two pixels accuracy in position and to a limited number of radius range segments in size. For later tracking and classification the circles have to be measured more accurately. As an additional effect this step verifies the circle candidates. This is necessary since through the rough estimation in the coarse detector phantom candidates

are created, especially in areas containing many edges, tree branches or bushes for example. These phantom candidates are eliminated in this step. The module used to accomplish the task is depicted in figure 2.4.

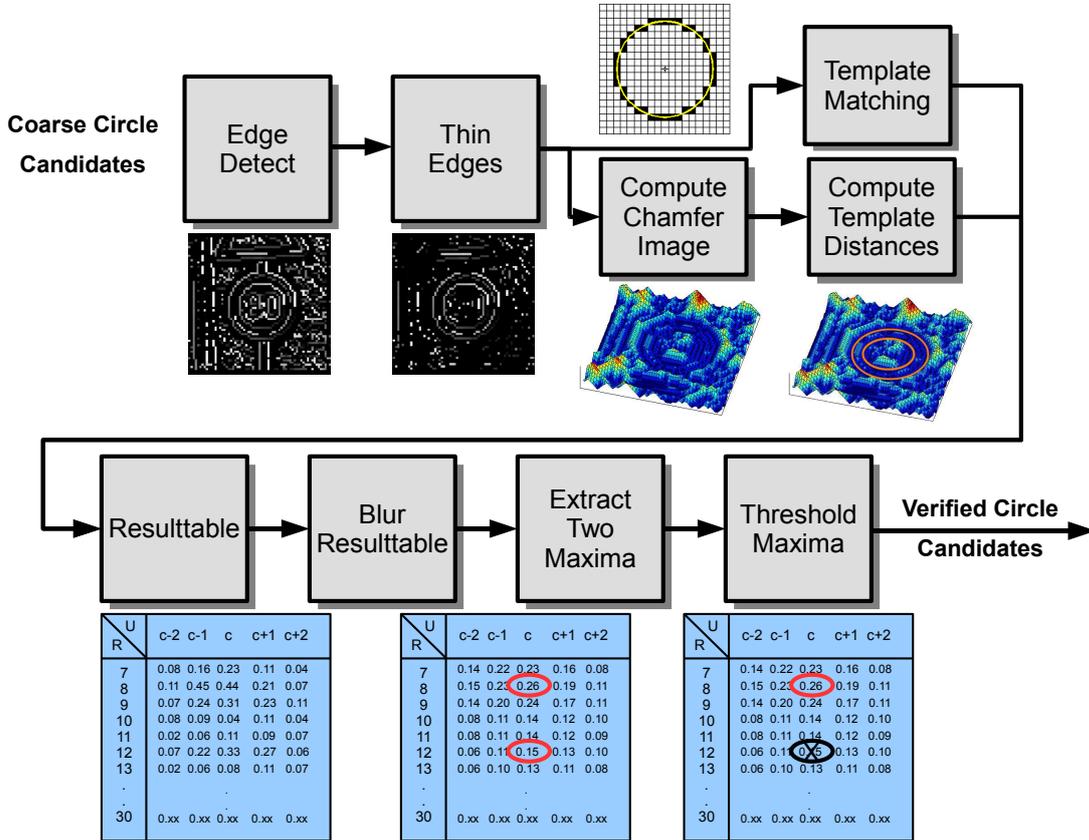


Figure 2.4: Scheme of the *Detector Verification*

There are two fast algorithms available that allow positioning and verification. One is edge correlation, the other is *Chamfer Matching*. Both work on edges in full image resolution and are robust to minor distortions of the detected circles. The edge detection used in the preprocessing step to both algorithms has to be insensitive to the effects introduced by the *Bayer-Pattern*. The resulting edge direction image is filtered, keeping only edges fitting to the templates' spatial and directional patterns and removing unconnected speckle pixels using the morphological erosion with a 3x3 pixel filter.

Edge-Correlation

This scheme for positioning and verifying the circle candidates uses the phase gradient image to correlate with ideal circle edge templates. The fitting edges on the circular template are counted and set into relation to the number of pixels on

the template. The template with the highest score has to exceed a threshold to be accepted as a verified circle candidate. For the detection of slightly distorted circles, such as signs being rotated along their vertical axis by a few degrees, the scores of templates having the same centre and adjacent radii are averaged, thus including pixels of ellipsoids into the score.

Chamfer Matching

The algorithm is based on a distance transform in which the value of each pixel in the transformed image represents the distance to the closest pixel set in the binary input image, in this case the edge image. The output is the sum of accumulated values under a mask applied to the distance transformed image, the so-called *Chamfer* image. The basic idea of *Chamfer Matching* are explained in the appendix section A.3. In the paper of Gavrilu et al. [Gavrila, 1998] this method is used as the main detection step. The algorithm explained in the paper and its follow ups requires a huge amount of memory for the 8 split directional edge images to be held in the system. This can be circumvented when it is used as a position refinement only, since the filtering of the edge directional image as explained in section 2.4.2 renders the use of multiple discrete gradient images unnecessary.

Retrieving the Verified Position

Following the correlation or *Chamfer Matching* step the three dimensional result table, holding values for centre coordinate and radius is filled with the correlation or matching scores. From the table the position of the best score is extracted as well as the position of the second best result with a minimum radius difference to the first result. The results have to be better than the threshold for the verification step to be accepted as a circle object to be passed to the tracker. For the correlation the threshold is given as the minimum percentage of the circle's edge pixels to be detected, for the *Chamfer Matching* it represents the maximum mean distance from the circle templates pixels to the next detected edge.

2.5 Traffic Sign Tracker

The tracker is the central component in the system. It is used to collect the traffic sign candidates in the image sequence from the detector, accumulate the detected circles over time and position and hand the tracks over to the classification system, administrating the classification results as well. The type of tracker used is a $\alpha\beta\gamma$ – tracker [Kalata, 1984], used on the centres of sign positions in the image and the signs sizes in pixels. When the ego motion of the camera is known the distance to the sign can be estimated and is used for tracking as well. The result of the tracking process is the connection of all detected circles in the image

sequence to real world objects, allowing their classification and the computation of their three dimensional world position relative to the vehicle.

The tracker is run before the classification of the circle candidates since the classification step is a computationally expensive step and should only be used on image objects that are confirmed by the tracker being able to predict their movement and thus building up a track. Some circles in the image are non-circular three dimensional objects in the real world that only seem to be circles in some views while driving past them. Typical candidates for these objects are tree branches or bushes. Another type of image objects to be removed by the tracker before presenting them to the classifier are moving circular real world objects like the wheels of crossing cars.

2.5.1 General Setup

The main task of the tracker is to combine circle candidates from the same and from consecutive images to tracked real world objects. The combination of circles

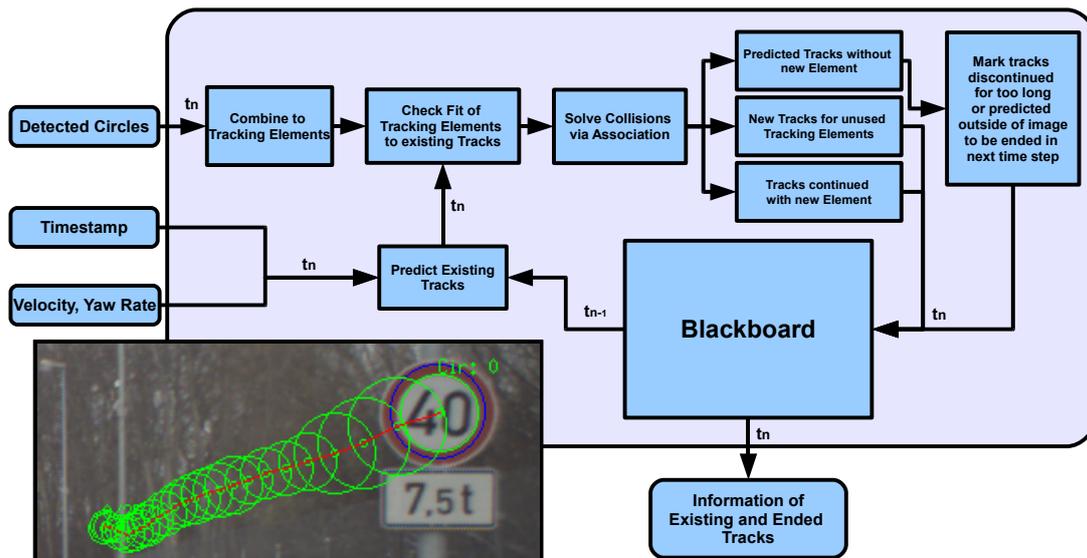


Figure 2.5: Scheme of the tracking system to assign tracks in the image sequence to real world objects

in a single frame is necessary since traffic signs are composed of concentric circles. These are the inner, usually white, circle holding the digits, the outer red circle and if the resolution is high enough the outermost white rim around the red circle. As depicted in figure 2.5 the position of the next element for already existing tracks is predicted either from the image coordinates and time difference

between the two frames only, or, if the information is available, with the use of additional information about the movement of the camera between the two time steps.

The existing tracks in the image sequence are checked for possible image objects for continuation close to the predicted position. Should one image object fit to more than one track or more than one object fit to one track an association module is called to solve the collision. Tracks that have no perpetuating image object get an empty image object at the predicted position to allow the continuation of the track in the next time step. Image objects not used to continue an existing track open up new tracks.

A track ends when its prediction puts the expected continuation image objects outside of the image area or the perpetuating objects were empty for more than a predefined timespan. Here this means the track did not find a fitting image object for a long time.

2.5.2 Prediction Without Egomotion

Should there be no information about the motion of the camera the whole tracking is based on the previous sightings of objects in the images belonging to the respective track. Since the task of connecting the tracks is not very difficult a simple movement pattern is used for the prediction. The model of movement used assumes image objects to have a roughly constant speed in pixels per second. Actually they have a constant acceleration in image coordinates given a straight movement of the camera at a constant speed, but since the movement of the camera is not known, the acceleration part of the prediction would allow even minor yaw movements of the camera to result in very high acceleration values in the prediction.

The prediction uses the $\alpha\beta\gamma$ – tracker [Kalata, 1984], there the parameter setting necessary to reach the optimal Kalman filtering [Kalman, 1960] given white noise is explained. The filter equation is depicted in equation 2.2. The state vector $\mathbf{x} = (s, v, a)^T$ consists of the position, velocity and acceleration of the tracked object in the image in pixels. The currently detected position is s_{det} and the position as predicted from previous detections is called $s_{\text{predicted}}$

$$\mathbf{x}_{t_{n+1}} = \mathbf{x}_{t_n} + \begin{pmatrix} v\Delta t + \frac{a}{2}\Delta t^2 \\ a\Delta t \\ 0 \end{pmatrix} + \begin{pmatrix} \alpha \\ \beta/\Delta t \\ \gamma/\Delta t^2 \end{pmatrix} (s_{\text{det}} - s_{\text{predicted}}) \quad (2.2)$$

Further elaboration on this prediction type is to be found in the diploma thesis of Stefan Eder [Eder, 1999], where the parametrization for α , β and γ is explained in detail. For tracking presuming constant velocity the terms including the acceleration are set to zero and the values for α and β change. The value for α is chosen by the equation given in [Kalata, 1984], see equation 2.3, where σ_w is the moving capability of the image object and σ_n is the measurement noise.

Using the values from our system this leads to an α between 0.80 and 0.95.

$$\alpha = \frac{-\Lambda^2 + \sqrt{\Lambda^4 + 16\lambda^2}}{8} \quad \text{with} \quad \Lambda = T^2\sigma_w/\sigma_n \quad (2.3)$$

	α	β	γ
constant position	α	0	0
constant velocity	α	$2(2 - \alpha) - 4\sqrt{1 - \alpha}$	0
constant acceleration	α	$2(2 - \alpha) - 4\sqrt{1 - \alpha}$	$\frac{\beta^2}{\alpha}$

Table 2.1: Tracker parameter sets

The new tracks always start as a single detector image object. The object in the next slot, the second for this track, is searched in an area centring around the previous detected coordinates, since velocity and acceleration are not yet initialized. In the following time steps the new position is determined from the movement of the previous sightings. Since there is always some error in the prediction of the movement the coordinate where the next image object is expected is expanded to an area around the predicted spot. The dimensions of this area depend on the accuracy of the previous prediction, using a small area when speed and coordinate were predicted well in the last frame, or a large margin when the previous prediction was inaccurate, indicating a change in the movement of the camera.

2.5.3 Prediction Using Three Dimensional Object Position Estimation

In the case of a known movement of the vehicle and thus the camera, the prediction can be more accurate. There are four factors influencing the pixel position of a given object in the next frame. In addition to these variables the constants the focal length and the coordinates of the focus of expansion in the image have to be known.

- The dominant factor is the yaw movement of the vehicle measured by the yaw rate sensor. This factor is easily compensated, since it translates directly into a pixel offset.
- The vertical pitch movement is not measured in the current setup, but is much smaller than the movement introduced by the yaw rate.
- The forward movement of the vehicle introduces another translation of the object in the image. This translation depends on the distance of the real world object and the distance from the centre of expansion in the image as well.

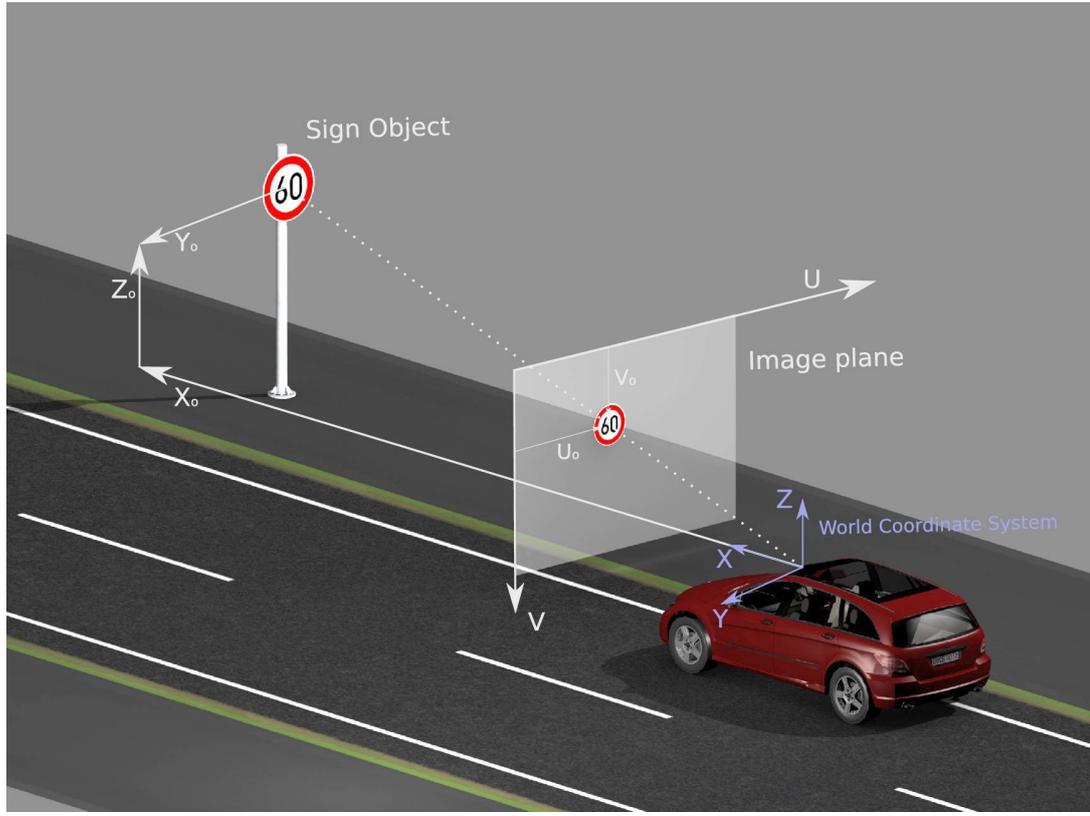


Figure 2.6: Coordinate systems

- The last factor is the lateral movement introduced by driving in a curve. For vehicles this factor can be neglected since it is in the range of a few centimetres between two frames only, leading to a minute change of the image objects image positions.

By computing the prediction from the vehicle movement as derived from the Ackermann Model (section 2.5.4), the distance to the sign in the real world can be estimated by finding the closest point to all lines of sight to the object from the different camera positions. A sketch of the coordinate system and the setup is shown in figure 2.6. The basic equations for the computation of the three dimensional position of the signs were evaluated in the diploma thesis by Elmar Tarajan [Tarajan, 2004], which was tutored by the author of this dissertation. Knowing the position of the sign in the real world and the vehicle's movement from frame to frame the expected position of the object in the next frame can be predicted. Due to the comparatively small field of view of the camera the main part of the movement in the image is due to the rotation of the vehicle. Given the unknown world position of the real world object when the system detects the first circle belonging to that object, the first estimation of the reappearance of the object in the next frame is imprecise. The estimation is based on the minimum

and maximum real world size of signs, the detected radius of the circle in the image and the ego motion of the camera, see figure 2.7. After two detections an estimate of the real world objects distance is known via triangulation and the margins for the pixel position of the object in the next frame can be adapted.

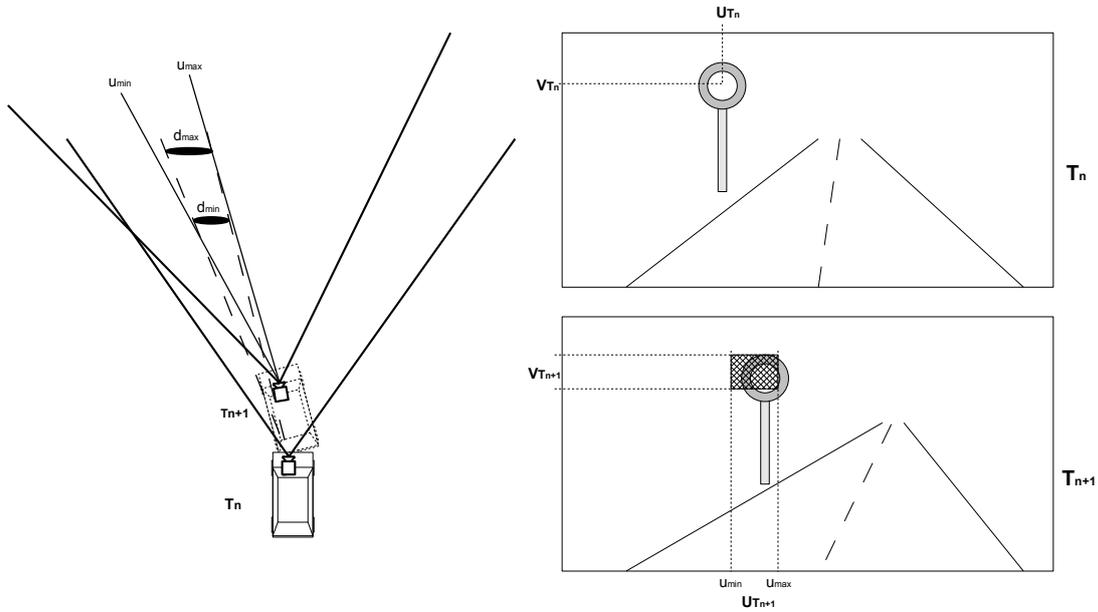


Figure 2.7: Deriving pixel positions u_{min} and u_{max} for the reappearance of the circle in the next image from the previous image position, the objects minimum and maximum size, d_{min} , d_{max} and the cameras movement.

2.5.4 Three Dimensional Positioning Formulas

The movement of the vehicle is approximated according to the Ackermann steering geometry by Rudolph Ackermann (1764-1834), reduced to a single track or bicycle model, expecting vehicles to run on a circle with a radius depending on the steering angle of the front wheels. The Yaw rate sensor proved to be much more accurate than the use of the wheel angles due to the slippage on the wheels. Since the system is much more sensitive to errors in the angle than in inaccuracies in forward movement the wheel angles are disregarded and only the yaw rate from the acceleration sensors are used. The movement is applied on the centre of the rear axle. The movement of the camera, which is fixed on the vehicle is obtained by first transforming the coordinates of the previous camera position into the centre of the rear axle, then applying the vehicle movement and after that transforming the position back into the camera's coordinate system, see [Tarajan, 2004].

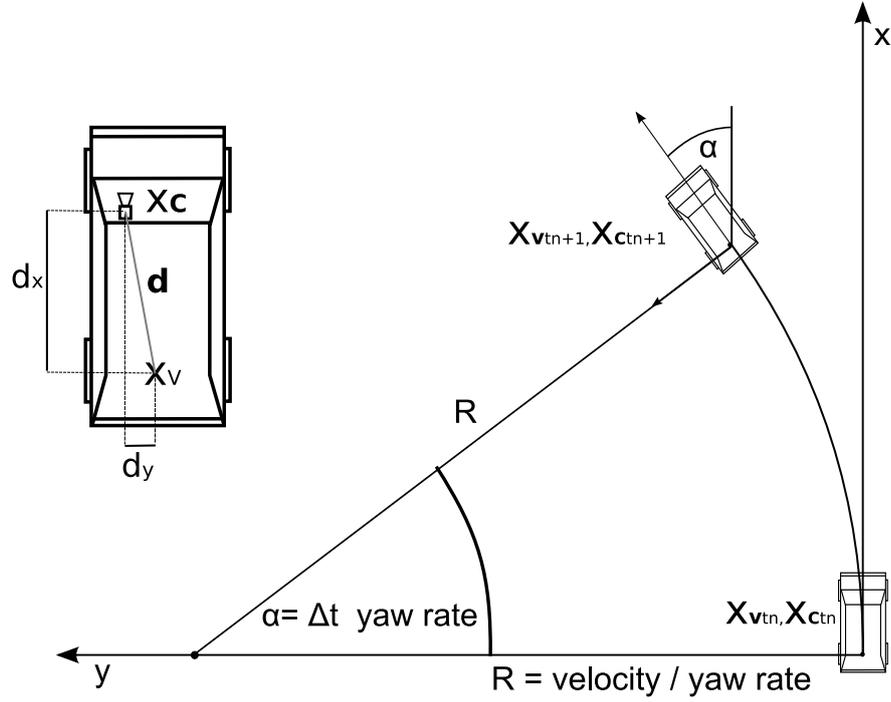


Figure 2.8: Ackerman vehicle movement and variable description with the parameters $R = \frac{\text{velocity}}{\text{yaw rate}}$ the radius of the circle the vehicle is supposed to move on and $\alpha = \text{yaw rate} \cdot \Delta t$ the angle moved

$$\mathbf{x}_{C_{t_n}} = \mathbf{x}_{V_{t_n}} + \mathbf{d} \quad (2.4)$$

$$\mathbf{x}_{V_{t_{n+1}}} = \frac{\text{velocity}}{\text{yaw rate}} \begin{pmatrix} \sin(\alpha) \\ 1 - \cos(\alpha) \end{pmatrix} \quad (2.5)$$

$$\mathbf{x}_{C_{t_{n+1}}} = \mathbf{x}_{C_{V_{t_{n+1}}}} + \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix} \mathbf{d} \quad (2.6)$$

The task is to find the real world position of the sign relative to the vehicle. The inputs are the lines of sight from the different vehicle positions to the sign as determined by detecting the signs centres in the images. The task then is to find the point being closest to all the lines of sight in the track. The vectors are given in three dimensional world coordinates relative to the start of the track. The resulting sign position is \mathbf{x} , the lines of sight from the respective camera position to the sign are the vectors \mathbf{g}_i and the vector from the closest point of the line of sight vectors to the sign position are \mathbf{c}_i . The sum of square distances of this vector has to be minimized. To find the minimum the formula is differentiated

for x, y and z . The resulting sign position is as elaborated in [Eder, 1999]:

$$\mathbf{x} = (x, y, z)^T \quad (2.7)$$

The line of sight vectors from the respective camera position to the sign is:

$$\mathbf{g}_i = \mathbf{s}_i + l_i * \mathbf{w}_i \quad (2.8)$$

The vector from the line of sight to the sign position at the closest point:

$$\mathbf{c}_i = \mathbf{x} - (\mathbf{s}_i + l_i * \mathbf{w}_i) \quad (2.9)$$

with:

$$l_i = (\mathbf{x} - \mathbf{s}_i) \cdot \mathbf{w}_i / (\mathbf{w}_i \mathbf{w}_i) \quad (2.10)$$

The goal of the algorithm is to find the point that has the lowest sum of squared distances to all lines of sight:

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \left(\sum_i |\mathbf{c}_i^T(\mathbf{x}) \mathbf{c}_i(\mathbf{x})| \right) \quad (2.11)$$

To solve for $\hat{\mathbf{x}}$ we differentiate the distance $\|\mathbf{c}\|^2$ by the position of the sign.

$$\begin{aligned} \sum_i (d|\vec{c}_i|^2/dx) &= \sum_i (f_i \cdot ((w_{iy}^2 + w_{iz}^2)x - w_{ix} \cdot w_{iy} \cdot y - w_{ix} \cdot w_{iz} \cdot z)) - \\ &\quad \sum_i (f_i \cdot ((w_{iy}^2 + w_{iz}^2)s_{ix} - w_{ix} \cdot w_{iy} \cdot s_{iy} - w_{ix} \cdot w_{iz} \cdot s_{iz})) \\ &= 0 \end{aligned} \quad (2.12)$$

with

$$f_i = 2 / (w_{ix}^2 + w_{iy}^2 + w_{iz}^2) \quad (2.13)$$

The differentiations of $d|\mathbf{c}_i|^2/dy$ and $d|\mathbf{c}_i|^2/dz$ are analogue. Solving the three equations for \mathbf{x} leads to

$$\mathbf{A} \cdot \mathbf{x} = \mathbf{b} \quad \text{with} \quad \mathbf{A} = \sum_i (\mathbf{A}_i/d_i^2) \quad \text{and} \quad \mathbf{b} = \sum_i (\mathbf{b}_i/d_i^2) \quad (2.14)$$

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{12} & a_{22} & a_{23} \\ a_{13} & a_{23} & a_{33} \end{pmatrix}, \quad \mathbf{A}_i = \begin{pmatrix} a_{i11} & a_{i12} & a_{i13} \\ a_{i12} & a_{i22} & a_{i23} \\ a_{i13} & a_{i23} & a_{i33} \end{pmatrix} \quad \text{and} \quad \mathbf{b}_i = \begin{pmatrix} b_{i1} \\ b_{i2} \\ b_{i3} \end{pmatrix} \quad (2.15)$$

with the squared length of the line of sight vector

$$d_i^2 = d_i^T d_i = w_{ix}^2 + w_{iy}^2 + w_{iz}^2 \quad (2.16)$$

and

$$\mathbf{A}_i = \begin{pmatrix} w_y^2 + w_z^2 & -w_x w_y & -w_x w_z \\ -w_x w_y & w_x^2 + w_z^2 & -w_y w_z \\ -w_x w_z & -w_y w_z & w_x^2 + w_y^2 \end{pmatrix} \quad (2.17)$$

$$\mathbf{b}_i = \mathbf{A}_i (s_{ix}, s_{iy}, s_{iz})^T \quad (2.18)$$

The equation can easily be solved by use of the Cramer rule. When either the x or the y component of the line of sight vectors have a very low variance in a track the equation can be solved disregarding the third dimension since it does not add information in the system. The equations simplify as shown in the following equations when using the information of the x, y dimensions only:

$$\mathbf{A}_i = \begin{pmatrix} a_{i11} & a_{i12} \\ a_{i12} & a_{i22} \end{pmatrix} = \begin{pmatrix} w_{iy}^2 & -w_{ix} w_{iy} \\ -w_{ix} w_{iy} & w_{ix}^2 \end{pmatrix}, d_i^2 = \frac{1}{w_{ix}^2 + w_{iy}^2} \quad (2.19)$$

$$\mathbf{b}_i = \begin{pmatrix} b_{i1} \\ b_{i2} \end{pmatrix} = \begin{pmatrix} w_{iy}^2 s_{ix} - w_{ix} w_{iy} s_{iy} \\ -w_{ix} w_{iy} s_{ix} + w_{ix}^2 s_{iy} \end{pmatrix} \quad (2.20)$$

and when using the information of the x, z dimensions only:

$$\mathbf{A}_i = \begin{pmatrix} a_{i11} & a_{i12} \\ a_{i12} & a_{i22} \end{pmatrix} = \begin{pmatrix} w_{iz}^2 & -w_{ix} w_{iz} \\ -w_{ix} w_{iz} & w_{ix}^2 \end{pmatrix}, d_i^2 = \frac{1}{w_{ix}^2 + w_{iz}^2} \quad (2.21)$$

$$\mathbf{b}_i = \begin{pmatrix} b_{i1} \\ b_{i2} \end{pmatrix} = \begin{pmatrix} w_{iz}^2 s_{ix} - w_{ix} w_{iz} s_{iz} \\ -w_{ix} w_{iz} s_{ix} + w_{ix}^2 s_{iz} \end{pmatrix} \quad (2.22)$$

solving to:

$$x = (a_{11} b_2 - a_{12} b_1) / (a_{11} a_{22} - a_{12}^2) \quad (2.23)$$

$$y \text{ or } z = (a_{22} b_1 - a_{12} b_2) / (a_{11} a_{22} - a_{12}^2) \quad (2.24)$$

As a measure for the quality of the solution of the equation the condition k of the matrix \mathbf{A} is used, with the inverse k' for being easier to be used in graphs since it has a clear upper bound of 1.

$$\begin{aligned} k &= \text{Ev1}(\mathbf{A}) / \text{Ev2}(\mathbf{A}) \\ k' &= 1/k \end{aligned} \quad (2.25)$$

with Ev1 being the larger eigenvalue of the matrix \mathbf{A} and Ev2 the smaller one. Since the matrix is 3x3 only, the eigenvalues can be computed without iteration procedures being necessary.

2.5.5 Selfcalibration of the Three Dimensional Positioning System

For the computation of the ego motion of the camera and determining the viewing angle the sign was detected under relative to the systems coordinate system some

calibration values have to be measured or estimated. These values are the

- The position of the vanishing point in the images, meaning the pixel in the image where the point of infinity is depicted when driving straight forward, thus the angle the camera is turned relative to the vehicle. This position in the image is called (u_0, v_0) for horizontal, respectively vertical pixel position.
- The position of the camera relative to the turning point of the vehicle in the Ackermann model used to compute the ego motion. The values are (l_x, l_y) for the offset from the turning point to the camera to the front and to the right respectively. The vertical position l_z is not part of the equations and thus does not have to be computed in the system.

For the traffic sign recognition system to be operational the camera has to point roughly ahead, thus the values for (u_0, v_0) are close to the centre of the image. The rotation of the camera along the x-axis has to be low for the classification system to work.

For the driver, especially if the system is not mounted permanently, but is fixed detachable in the vehicle, the autonomous measurement of the calibration values in question is highly desirable. The values could be measured or adapted by the operator of the system every time the mounting changed or the mounting was stressed by temperature changes or rough road conditions, but this would greatly reduce the usability of the system.

The algorithm for the autonomous computation of the calibration values is based on the fact that the mean distance of the lines of sight from the camera to the sign $\Delta_m = \min u_0, v_0, l_x, l_y \sqrt{(\sum_i |\mathbf{c}_i^T \mathbf{c}_i|)}$ is rising when the wrong calibration values are used, since the rays do not intersect in one point any more in a decalibrated system. Using this the algorithm for self calibration is as follows:

1. start the traffic sign recognition system with reasonable values, meaning (u_0, v_0) being in the image centre and the cameras offset to the turning centre being about 1.5m to 2m in front without lateral offset. This rough setting allows the tracking system to work, while being too inaccurate for measuring the three dimensional positions of the signs.
2. Gather tracks, verified by the classifier to belong to a traffic sign, that have a low condition k , thus the solution of the three dimensional equation can be trusted.
3. Divide tracks in two types. The tracks that were recorded while the vehicle was going straight, meaning not driving curves, eliminating the influence of the camera position relative to the turning centre of the Ackermann equation and the tracks recorded in curves, meaning tracks in which the relative position of the camera affects the computation.

4. Pick the tracks from straight movement of the vehicle with low variance of the vertical viewing angle, meaning signs left and right of the road. Then minimize the distance mean Δ_m by varying the value for the images horizontal centre point u_0 .
5. Pick tracks from straight movement of the vehicle with low variance of the horizontal viewing angle, thus signs being above the lane. Then minimize the distance mean Δ_m by varying the value for the images vertical centre point v_0 .
6. Pick the tracks with $\sum |\alpha| > \text{threshold}$, meaning the vehicle driving a curve. Use the values determined for the centre point of the image above (u_0, v_0) and find the values for the translation of the camera by minimizing $\Delta_m(l_x, l_y)$.

The equations below show the case of just two time steps in which the object was detected to clarify the effect of the camera not being mounted in the vehicles turning point see figure 2.9 and the following equations.

The positions of the turning point of the vehicle in the Ackermann model in two time steps \mathbf{x}_{V0} , \mathbf{x}_{V1} and the respective positions of the camera \mathbf{x}_{C0} , \mathbf{x}_{C1} are given in the equations 2.26.

$$\begin{aligned}
 \mathbf{x}_{V0} &= \begin{pmatrix} x_{V0x} \\ x_{V0y} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \mathbf{x}_{V1} = \begin{pmatrix} x_{V1x} \\ x_{V1y} \end{pmatrix} = d \begin{pmatrix} \cos(\alpha) \\ \sin(\alpha) \end{pmatrix} \\
 \mathbf{x}_{C0} &= \begin{pmatrix} x_{C0x} \\ x_{C0y} \end{pmatrix} = \begin{pmatrix} l_x \\ l_y \end{pmatrix} \\
 \mathbf{x}_{C1} &= \begin{pmatrix} x_{C1x} \\ x_{C1y} \end{pmatrix} = d \begin{pmatrix} \cos(\alpha) \\ \sin(\alpha) \end{pmatrix} + l_x \begin{pmatrix} \cos(\alpha) \\ \sin(\alpha) \end{pmatrix} + l_y \begin{pmatrix} -\sin(\alpha) \\ \cos(\alpha) \end{pmatrix}
 \end{aligned} \tag{2.26}$$

The equations for the intersections when disregarding the difference between turning point and camera mounting deliver the distance r_V from the position at time step 1 to the intersection. The correct distance is r_C when observing the mounting point of the camera relative to the turning point. The values r'_V and r'_C are derived by using the approximations for small angles $\sin(\alpha) = \alpha$ and $\cos(\alpha) = 1$. The approximations are added to clarify the dependencies. The complete derivation is in the appendix section A.12.

$$r_V = d \frac{\cos(\alpha) \sin(\beta_0) - \sin(\alpha) \cos(\beta_0)}{\sin(\beta_1 + \alpha) \cos(\beta_0) - \cos(\beta_1 + \alpha) \sin(\beta_0)} \quad (2.27)$$

$$r'_V = d \frac{\beta_0 - \alpha}{\alpha + \beta_1 - \beta_0} \quad (2.28)$$

$$(2.29)$$

$$\begin{aligned} r_C = & d \frac{\cos(\alpha) \sin(\beta_0) - \sin(\alpha) \cos(\beta_0)}{\sin(\beta_1 + \alpha) \cos(\beta_0) - \cos(\beta_1 + \alpha) \sin(\beta_0)} + \\ & l_x \frac{(\cos(\alpha) - 1) \sin(\beta_0) - l_x \sin(\alpha) \cos(\beta_0)}{\sin(\beta_1 + \alpha) \cos(\beta_0) - \cos(\beta_1 + \alpha) \sin(\beta_0)} + \\ & l_y \frac{-\sin(\alpha) \sin(\beta_0) - (\cos(\alpha) - 1) \cos(\beta_0)}{\sin(\beta_1 + \alpha) \cos(\beta_0) - \cos(\beta_1 + \alpha) \sin(\beta_0)} \end{aligned} \quad (2.30)$$

$$r'_C = d \frac{\beta_0 - \alpha}{\alpha + \beta_1 - \beta_0} + l_x \frac{-\alpha}{\alpha + \beta_1 - \beta_0} + l_y \frac{-\alpha \beta_0}{\alpha + \beta_1 - \beta_0} \quad (2.31)$$

In the equations above the names of the variables are used from figure 2.9. The variables \mathbf{c}_0 , \mathbf{c}_1 , \mathbf{v}_0 , \mathbf{v}_1 are the position of the camera and the turning centre according to the Ackermann equation at time step 0 and 1. The value d is the distance travelled between the two time steps and α the angle the vehicle has turned. The vector \mathbf{s} describes the direction of the line of sight from the respective point to the crossing point of the lines of sight and thus the expected position of the sign. When computing the crossing point \mathbf{s} the distance from the points coordinate at the second position of the vehicle and camera is given in the variable r . Due to the small absolute values of the angles α , β_0 and β_1 involved the values for r are given with the approximation $\sin(\alpha) \approx \alpha$ and $\cos(\alpha) \approx 1$ for clarification in r' . There the influence of the vector \mathbf{l} can be evaluated. As can be seen when the vehicle is moving straight ($\alpha = 0$) the vector \mathbf{l} does not influence the result of the triangulation. When the vehicle is turning the effect of the longitudinal distance of the camera to the turning point has a much higher impact than the lateral offset, since l_y has the small angle β_0 as additional factor diminishing the influence of the lateral offset on the distance measurement. The effect of the lateral camera offset on the computed real world position of the sign has proven to be neglectable and thus not further used in the system.

The experiments presented in section 6.4.3 have shown that the traffic signs three dimensional positions can be measured to better than one meter longitudinal accuracy.

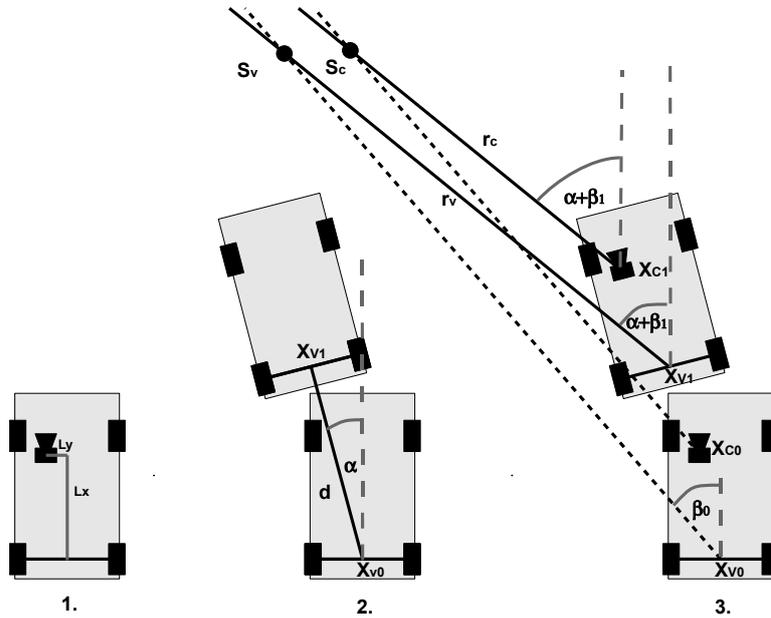


Figure 2.9: Scheme of the self calibration with

1. the offsets of the camera position relative to the turning centre of the Ackermann model
2. the movement of the vehicle
3. the points where the visual lines cross, from the model with and without providing for the camera not being in the turning centre

2.5.6 Solving Assignment

There are some cases, especially when the prediction has to be made without knowledge about the ego motion, in which more than one image object fits to one track or more than one track fits to one detector object. A standard case are two signs on one post above each other.

The easiest solution is simply taking the closest image object and track prediction pair, connect them and go for the next pair. Since the pitch rate is unknown an abrupt pitch movement introduced by a bump in the road might lead to having one pair, one unconnected track and an unused detector object. To prevent this the sum of distances of all connections should be optimized, giving unconnected members very high distance values, thus minimizing the number of unconnected tracks or image objects. The algorithm used to solve this problem is the *Hungarian Method* [Kuhn, 1955].

2.5.7 Use of the Tracker as Blackboard

As mentioned above the tracking module is not only used to compute how to connect the detected circles in the image sequence to tracks, but as a blackboard

to store the gathered data for all modules in the system as well, see figure 2.5. The following elements are stored in this blackboard:

- The positions and qualities of the detected circles.
- The part of the raster image holding the circle, further on called *cutout*.
- The estimated three dimensional position of the real world object,
- The classification results for each *cutout*.
- Information about eventual supplementary signs, their *cutouts* and classification results.
- Information necessary for the extension modules described in section 2.8.

The storing of the *cutouts* allows the implementation of additional modules rechecking the complete track in refinement steps as described in section 2.8.2. Keeping the whole images would be possible as well, of course, but would necessitate the use of much more memory. Running the extension modules for all *cutouts* and not just the ones requiring the use of the additional modules often forbids itself due to the computational cost.

2.6 Classifier

The classifier to be used has to cope with a huge amount of data due to the possible variations in the observed signs. In addition the image representations of the signs are distorted as explained in section 4.2.2. To achieve the goal of firstly determining if the image object to be classified is a relevant traffic sign and secondly which specific type of sign the detected patterns have to be classified. The first task in the classification process is to provide meaningful features from the image patterns to be classified. The representation of the patterns at the input side is a raster image of the circular objects found in the image by the detector. These images vary in their size, brightness, grade of blurring and more characteristics, as explained in section 4.2.2. To remove as many systematic differences as possible between the patterns to be recognized, a normalization process is used as described in the following section 2.6.1.

There is a huge variety of classifier types available to the scientific community. To decide which type to use the characteristics of the problem has to be considered. The patterns to be classified differ even when belonging to the same class, for instance due to the use of different fonts for the digits in speed limits. In addition there is a high number of classes to be discerned.

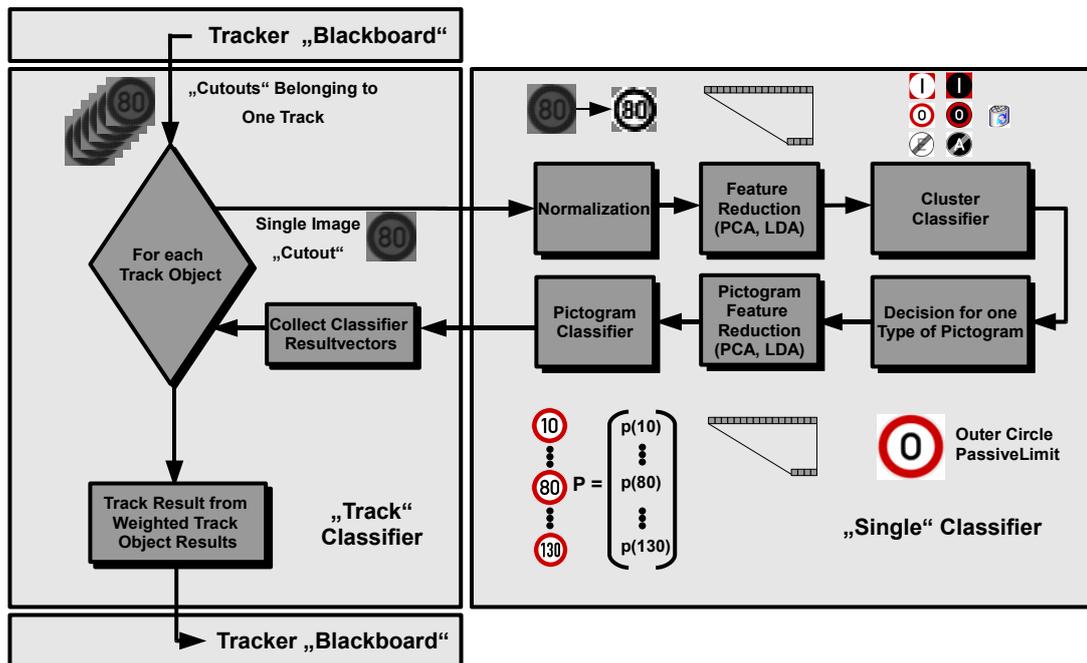


Figure 2.10: Scheme of the speed limit classification

- Neighbourhood classifiers like nearest neighbour or k-nearest neighbour tend to grow very fast in their computational complexity when facing this variety of inputs since their capability to generalize is gained mainly by adding further labelled examples to the feature space.
- Heuristic classification failed since the problem is too complex to conceive the rules necessary for such a network.
- The best way to cope with the high demands of variability and generalization capabilities required by the problem was found in probabilistic classifiers.

The first step for object classification is to remove as many systematic differences between the patterns to be recognized as possible. Given enough examples in the learning set and a high amount of computation time these deviations could be learned as well, but this would introduce unnecessary complexity into the system. The sign representations at the input side of the classifier have sizes from a few hundred to thousands of pixels, depending on the real world distance to the sign and its real world size, in a depth of 8 to 12 bit, depending on the image sensor.

For the confirmation and stabilisation of the classification the single classification results are accumulated and a weighted result extracted as shown in

section 2.6.3. The classification of more than one pattern belonging to the same tracked object, but taken from consecutive frames, as shown by Christian Woehler [Woehler and Anlauf, 1999] for the classification of pedestrians is not considered useful for traffic signs, since the object to be classified is rigid and thus no additional information besides confirmation of the current result can be derived by simultaneously classifying temporally successive patterns.

The selection of features to use for the classification is based on the following reasons:

- All signs are installed vertically except for few degrees, thus no rotation invariant features are used. This includes histogram based features and power spectrum features, meaning the squared magnitude of the the Fourier transforms frequency components.
- The histograms of oriented gradients as used in the verification stage for the supplementary signs, see section 2.7.2, rely on stable edge directions which cannot be extracted from the very small structures in the signs patterns.
- The Fourier transform components or the elements of the discrete cosine transform are not used due to the high frequencies the patterns have in position space as shown in figure 6.1.
- Wavelet features like those produced by the Haar- or Daubechies wavelet transform could be used to get scale invariant features, but since the scale of the patterns is known through the diameter of the detected circle and the discrete wavelet transform is inaccurate given the high quantisation noise that originates in the small size of the input patterns.
- The pixel pattern as detected in the image can be used directly as input to the classifier. To get a constant number of input values for the classifier the pattern is scaled to a standard size using bilinear interpolation. The patterns are normalized with respect to brightness and grey value variance as well as described in the following section.

2.6.1 Normalization

The normalization is used to prepare the input feature vector for the following classification. The base information for the classification consists of the pixels in the detected circular region of the image. For ease of use the enclosing rectangle is extracted from the image and the area not belonging to the circle is masked out later in the process. The rectangular raster image regions extracted are further on called *cutouts*. The input of the classifier is a vector of constant length. Thus when using the pixels values of the *cutout* as input the *cutouts* have to be resized to an equal width and height. The algorithm used for resizing is bilinear interpolation.

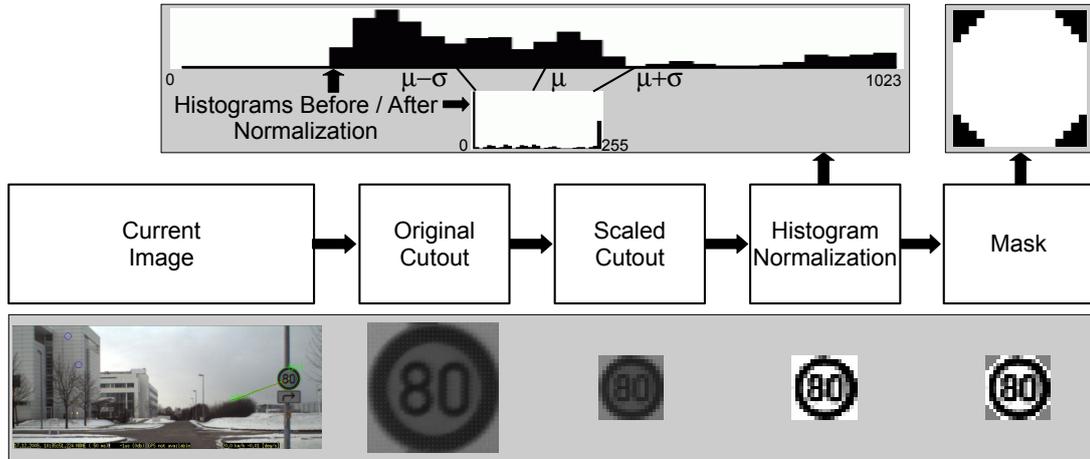


Figure 2.11: Scheme of the normalization steps for the raster image regions holding the sign candidates

To determine the best output size of the resizing and thus the input dimension of the classifier a large number of *cutouts* was resized and visually inspected. The smallest size where almost all *cutouts* were still recognizable was 17x17 pixels and thus 289 input features for the classifier. This was verified by training classifiers for different scales. Patterns of smaller sizes reduced the classifiers quality, while larger sizes lead to higher computational cost without yielding better classification results.

The resized *cutouts* vary strongly in the grey value mean and the variance of their grey values. These differences do not add to the recognizability of the signs. To enhance the similarities of *cutouts* showing traffic signs the patterns are normalized for their grey value mean and their grey value standard deviation. Again the visual inspection by a human observer, verified by adapting classifiers for different settings, was used to determine the best algorithm.

The output of this normalization step is a 17x17 pixel raster image of an 8-bit grey value range, thus the values from 0 to 255, with a mean value of 128. The values are spread linearly from $\mu - \sigma$ equalling zero and $\mu + \sigma$ equalling 255. Lower values are clipped to 0, higher values to 255 respectively.

The last step in the normalization stage is the masking of the rectangular raster image region which is outside of the detected circle.

The presented traffic sign recognition system has worked on images acquired from mobile phone cameras, from several industrial cameras and in a control unit especially designed for it without mayor performance decrease, thus the steps taken to normalize the *cutouts* can be considered useful for most camera types.

2.6.2 Single Pattern Classifier

The classifier has two tasks. Firstly it has to verify that the image pattern represents one of the traffic signs searched for and secondly it has to discriminate between the different classes of signs. Even on the strongly regulated roads in Germany only every few kilometres a *speed limit* or *end of speed limit* sign is to be found. Thus the number of tracks depicting circles that are not a sign of interest or no sign at all are in the vast majority. This implies the classifier has to be very strict in rejecting false positive patterns.

The complete set of classes searched for consists of 70 classes as depicted in figure 2.2. Even when combining classes of similar meaning, here the different types of *end of limit* signs, the reduced set as shown in figure 2.12 still consists of 42 different classes. Most types of classifiers degrade in selectivity when too many classes are to be separated.

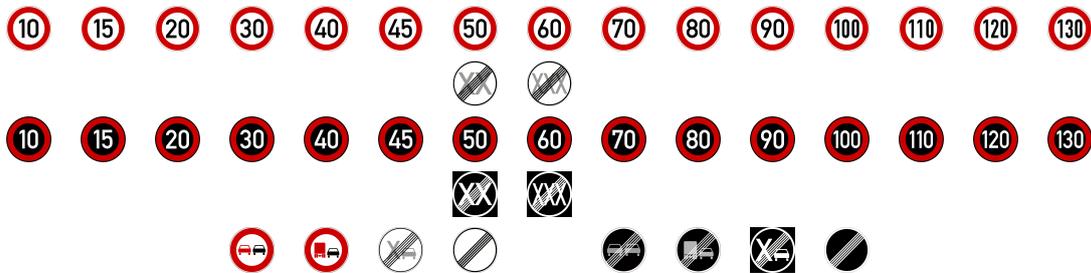


Figure 2.12: Merged set of signs to be classified by the system

The considerations above lead to the implementation of a two stage hierarchical classifier. The first stage is used to reject the main part of the patterns not representing a sign and has the second function of determining the type of the sign. As different types the following six were identified: outer circle of a limit, inner circle of a limit, end of limit and the same for active signs, see figure 2.13. The blue arrow signs are an add on, since they present a very common type of circular sign encountered on European roads and thus were considered important as well, if only as a help to reject signs not belonging to the set as defined in figure 2.12. An additional class used in every classifier is the rejection or garbage class, representing all objects not belonging to one of the classes the classifier is adapted for.

For the classification of one stage in the hierarchy the number of features presented to the classifier has to be reduced, in our case by using the principal component analysis (PCA). The dimensionality of the classification problem has to be reduced to allow an efficient training of the classifier. Different schemes for feature reduction were considered, e.g. using a neural network [Kressel et al., 1999] or using a linear discriminant analysis, but the best performance in generalization and ease of retraining was reached using the PCA [Ott, 1977].

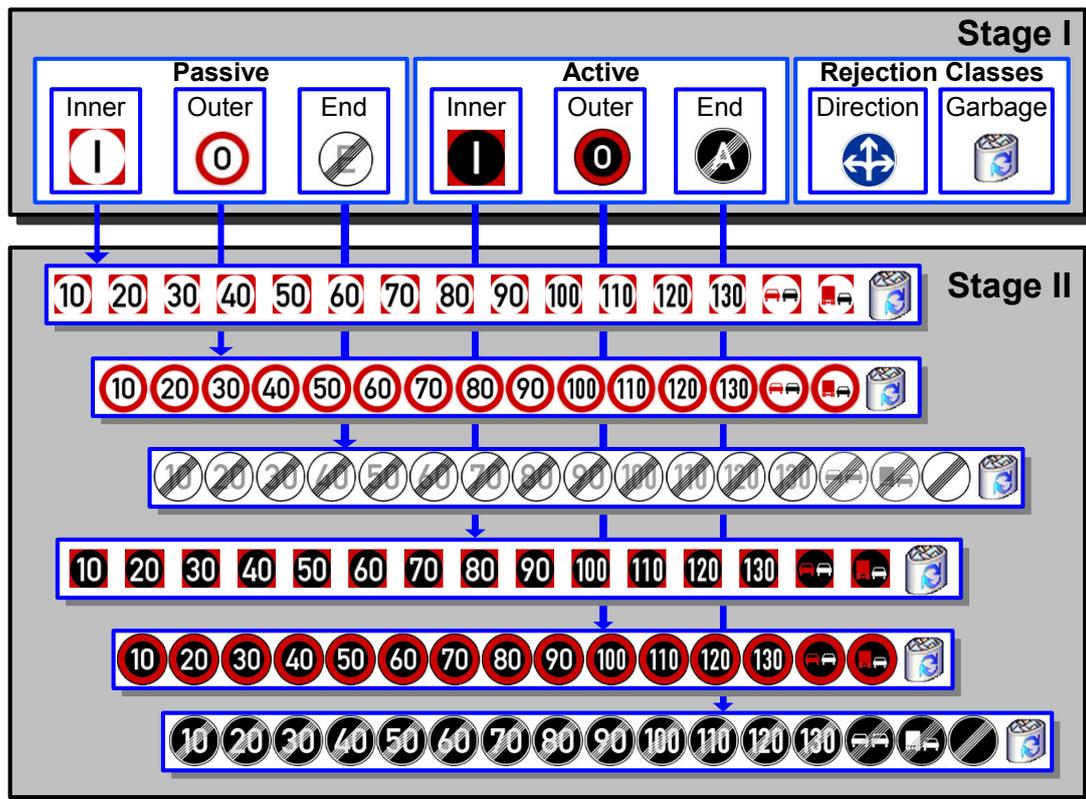


Figure 2.13: Hierarchical classifier setup, one cluster classifier in stage I, followed by six pictogram classifiers for four pictogram types in stage II, directional arrows used as rejection class only, as well as the *garbage* class

For the classification of the reduced feature set many different classifier types are feasible. The main requirements are classification performance, computational speed and ease of retraining. Considering these factors the polynomial classifier of second degree was selected [Ott, 1977]. As shown in [Kressel et al., 1999] a set of receptive field neural networks combined with radial basis classifier performed equally well where computational speed and recognition rate are concerned, but the retraining with new training sets added to the huge existing sets uses much more, up to two magnitudes, computational effort than retraining the polynomial classifier. The polynomial classifier has the additional capability of delivering a reliability of the classification result which can be used to select which of the single results to use in the track classification, see the section 4.1. More information about the mathematical foundations and the adaptation of the classifier and the principal component analysis will be given in the chapter 4.

The above classification system is adapted to the appearances of traffic signs of a specific country. To allow the use of a limited set of classifiers for a large

number of countries, countries in which the signs have compatible appearances share the same classifier. The decision which classifiers to use is done using the *Global Positioning System* and a map. The algorithm discerning which countries appearance types to recognize with a single classifier is explained in sections 4.4 and 6.7.3, an exemplary setup is shown in figure 4.3.

2.6.3 Track Classification

The last step in the classification process is the combination of the single pattern classification results to a conclusion for the whole track, meaning to conclude from a sequence of image objects which class the corresponding real world object belongs to. The tracks have very differing lengths, depending on the vehicle speed, lateral offset of the sign, obstacles in the line of sight and lighting conditions. The classification results have a reliability depending on the size of the sign in the image and depending on possible motion blur. All these factors have to be taken into account when merging the single classification results.

The proposed method uses the reliability, depending on the size of the signs, computed from the training set as weights for the output probabilities of the single pattern classifiers. The accumulated result is the weighted sum of these probabilities, the decision being made by the maximum relative probability. The relative probability has to be above a threshold, otherwise the classification is rejected and set to the result class *Garbage*. An additional rejection step has been introduced by thresholding the sum of single probabilities for the resulting class. This is necessary since the relative probability compared to the other classes may be high, but the absolute sum might still be low when the track was short or many of the entries could not be identified due to the being rejected by the single pattern classification system.

2.7 Supplementary Sign Detection and Recognition

Supplementary signs are positioned close to the sign they are corresponding to. There are very different types of sizes and positions relative to the associated sign, see section A.13. The vast majority of the sign positions are centred beneath the circular sign. The low resolution of the supplementary signs in the image does not allow the use of *optical character recognition* (OCR). In speed limit signs there are two to three digits, while on the supplementary signs often whole sentences are written on a sign of the same area. This necessitates the use of alternative algorithms.

There are two types of supplementary signs, explanatory and restricting. The explanatory type has no impact on the sign above, while the restrictive regulates for whom or when the sign above is valid. Thus the restricting signs are regarded in this system only. The rough position of restrictive supplementary signs is known, since almost all of them mounted directly below the circular sign. The

few signs not positioned underneath the main sign can be detected and recognized with the algorithm presented in this chapter as well, since the concept is generic with respect to position and size of the sign, but the implementation was realized regarding the standard position of the signs only.

Additional challenges in detecting and classifying the supplementary signs are the higher variety in shape and size as well as the edges usually being much less prominent in the image than for the circular signs.

The general setup of the supplementary sign detection and recognition is depicted in figure 2.14. The detection step shows two possible variants explained in the following sections. An AdaBoost system has been tested for detection as well, but was found to be slower and having less detection capability than the custom designed detectors. After the detection of possible sign candidates a verification step based on *Histogram of Oriented Gradients* features checks which of the candidates is to be passed on to the following classification stage which is designed similar to the scheme for the circular signs.

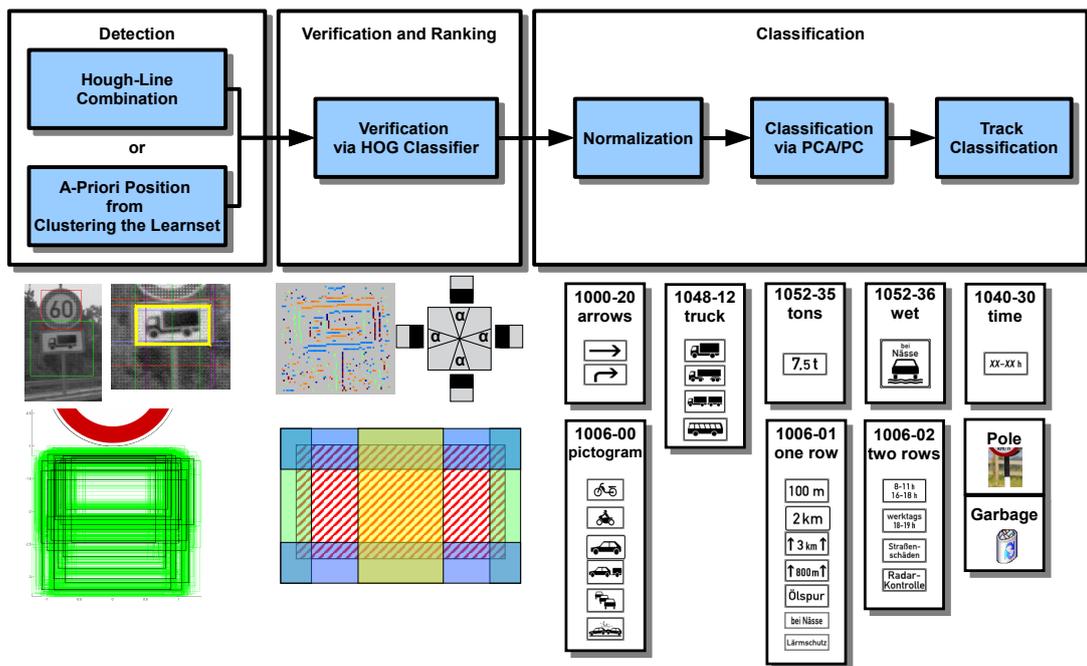


Figure 2.14: Scheme of the supplementary sign detection and recognition

2.7.1 Supplementary Sign Detection

The rectangular supplementary signs are less well defined in their shape and looks than circular signs. The aspect ratio varies between 1/12 to 1/1 width to height ratio. The outline has less distinct edges compared even to *End of Limit* signs, since the black lines printed on the edges of the signs are thinner or even

missing. Algorithms based on edge information are problematic in a second way since the pictogram of the signs often possess strong vertical and horizontal edges themselves, leading to a number of detections for rectangles in the inner area of supplementary signs.

Two algorithms can be used in the first detection step. Both are used to find the outline of the supplementary signs, since the variability of the pictogram is too great to be used as a detection cue.

The first algorithm is based on the *Radon Transform* [Radon, 1917], which in this case is closely related to a marginal distribution for the detection of horizontal and vertical lines as used in this case. The idea is to accumulate the horizontal edges along elements of the same vertical coordinate to one histogram, the horizontal marginal distribution, and the vertical edges along the same horizontal coordinate into another, the vertical marginal distribution. The peaks in these distributions represent hints to the existence of horizontal, respectively vertical lines. Using the transform on the horizontal and vertical edges in the region of the image where presumably are supplementary signs leads to a number of hypothesis for the position of horizontal or vertical lines.

The combination of the lines detected in the radon transform step to rectangles leads to candidates for the position of supplementary signs. In this process slight rotations of the rectangles have to be taken into account. This was not necessary for the circular signs, but for the rectangular shapes rotations of up to 10 degrees were recorded due to tilted sign posts, even if over 99% of the signs were tilted less than 7 degrees. Another factor is the size and aspect ratio to be taken into account. The rough size can be derived by adapting the search domain to the size of the circular sign the rectangle is belonging to. The aspect ratio depends on the sign types searched for. For the restrictive signs the aspect ratio from 0.5 to 1.2 are sufficient to cover next to all supplementary signs.

The following equation shows the actual algorithm providing a histogram for each angle examined. The realization in software uses Bresenham lines [Bresenham, 1965] saved in tables to speed up the accumulation of the histograms and uses a low-pass on the two dimensional, in angle and position, accumulators to get distinct peaks. Due to the comparatively high noise in the angle the results are close to the use of a one dimensional accumulator, in position only. Thus adding up the results for the angles being close to being horizontal or vertical in the margin of ± 7 degrees, as mentioned above. The following equations show the Radon Transform at the pixel position u and v . for an horizontal/vertical angle offset Δa .

$$H(u, \Delta a) = \sum_{v_0}^{v_1} I(u_{v_0} + v \tan(\Delta a), v) \quad (2.32)$$

$$H(v, \Delta a) = \sum_{u_0}^{u_1} I(v_{u_0} + u \tan(\Delta a), u) \quad (2.33)$$

The second algorithm is used to allow for supplementary signs of which the contrast is too low to be detected by the previous algorithm. It is based on the

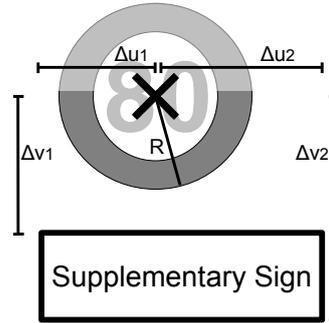


Figure 2.15: Parameters to be clustered Δu_1 , Δv_1 , Δu_2 and Δv_2

labelling of a set of supplementary signs and their relative size and position compared to the circular sign they belong to. This leads to a set of five parameters per labelled sign, this being upper left corners u and v pixel coordinate relative to the circular signs centre position, lower right corners u and v pixel and rotation angle. In the following steps the rotation angle is neglected due to being comparatively small. The main idea is to use the labelled positions as a priori knowledge where supplementary signs are positioned. Since the large number of positions cannot be tested in the following steps due to computation time restrictions the positions have to be clustered first, leading to typical relative supplementary sign positions. The remaining four parameters Δu_1 , Δv_1 , Δu_2 and Δv_2 relative to the circular signs centre position, are clustered, see figure 2.15. The following three approaches, all of them using Euclidean metrics, were tested:

- **Agglomerative** This clustering algorithm starts with as many clusters as there are samples and merging those which are closer than a threshold. The distances are computed to the cluster centres. The clustering process stops when there are no more possible additions to a cluster. Clusters with a small number of member samples are treated as outliers and dismissed in the following detection steps. The number of clusters is thus variable and depends on the sample set and the threshold as well as the minimum cluster size.
- **Divisively** The type of divisive clustering used is called vector quantisation [Schuermann, 1996a]. It starts with one cluster holding all samples. The cluster is split along the plane defined by being orthogonal to the line of highest standard deviation of the cluster and the centre of the cluster. This is repeated for the cluster with the highest remaining standard deviation until a minimum number of elements in the clusters is reached, the standard deviation per cluster is below a threshold, or a maximum number of clusters is reached.
- **K-Means** This scheme, as described in [MacQueen, 1967] starts with a

fixed number of clusters. The centres are either random coordinates, randomly drawn single samples' coordinates or scattered in a chosen raster. The algorithm assigns each sample to the closest cluster centre. After this assignment step the centres are recomputed and the assignment step is repeated. This process is rerun until there are no more changes in the cluster memberships. For this clustering scheme the number of resulting clusters is pre-set at the start of the process.

All three clustering schemes were evaluated and achieved cluster centres that were equally satisfying in covering the samples in the four dimensional position space and in combination with the following verification and classification steps. K-Means and divisive clustering reached roughly similar cluster centres after adding a rule allowing the removal of outliers. This rule includes the removal of clusters with too few members and the renewed computation of the remaining cluster centres after removing all samples farther away from the cluster centre than 70% or one standard deviation of all cluster members. With agglomerative clustering these rules were inherent in the clustering process by allowing clusters of a certain minimum number of elements only and thus removing the small clusters containing outliers. For ease of use and slightly better adaption to the following classification process agglomerative clustering was used for the following evaluations. As metrics for the distance determination in the cluster process three different algorithms were tested, the definitions for an N-dimensional metric are taken from [Duda et al., 2000].

$$\begin{aligned}
 \text{Manhattan : Minkowsky } L_1 : \quad d_m &= \left(\sum_{i=0}^N |x_{0i} - x_{1i}| \right) \\
 \text{Euclidean : Minkowsky } L_2 : \quad d_e &= \sqrt{\sum_{i=0}^N (x_{0i} - x_{1i})^2} \\
 \text{Chebyshev : Minkowsky } L_\infty : \quad d_c &= \max |x_{0i} - x_{1i}| \text{ with } i \in 1..N
 \end{aligned} \tag{2.34}$$

Since the positioning of the supplementary signs relative to the main sign varies strongly from country to country, the step for determining the clustered detection areas has to be done for each country separately. In the online system the according cluster rectangle set is selected for the country the vehicle is currently travelling in.

2.7.2 Supplementary Sign Verification

The detection process as described above leads to a comparatively high number of candidates for supplementary signs. To classify all these with the pictogram classifier would be computationally expensive and would lead to a high number of false positives. The step proposed here is to use a comparatively less costly

classification scheme to rule out candidates that do certainly not hold supplementary signs. Due to the pattern in the sign being very variable the classifier for the verification of the existence of a supplementary sign is using information from the edge of the candidate only, plus an area in the middle of the sign where mostly the sign post is positioned when there is no supplementary sign.



Figure 2.16: Examples of German supplementary signs

The areas inspected are schematically displayed in figure 2.16. The five inspected regions cover the four edges with a margin of $\pm 10\%$ of the sign size and the horizontal centre $\pm 10\%$ of the signs width for the whole signs height. From these regions *Histogram of oriented Gradient* features as described in [Dalal and Triggs, 2005] are extracted. These are combined to a vector which is classified by a polynomial classifier. More information on this algorithm is found in chapter 6.8. The output vector of the classifier is used to reject unlikely candidates for supplementary signs. In the case of too many elements passing the threshold for being accepted as supplementary sign the classifier is used for ranking the candidates and allowing only the candidates with the highest classification score to be handed over to the pictogram classifier. Should the clustering scheme have been used for the creation of the candidates an optional refinement step could be used. By shifting the borders of the candidate left and right, up and down in a narrow margin until the highest count of edge in the correct direction has been reached often a slightly better position of the candidate can be reached. This module is independant from the country the image sequences are recorded in.

2.7.3 Supplementary Sign Recognition

For the classification of the candidates yielded by the detector the combination of principal component analysis and polynomial classifier as described in chapter 2.6 is used. The normalization step is adopted from there as well since the supplementary signs are monochrome like the digits in the speed limits. Due to the small letters on the supplementary signs no optical character recognition

is used, but the sign types are clustered to classes of similar appearance and meaning, see figure 2.17. For German signs, as shown in the figure, this results in 8 *positive* classes and one garbage or reject class.

Since there can be more than one detected candidate for the same supplementary sign, the candidate having the best classification result for one of the non-garbage classes is taken to be the one being closest to the correct position of the sign and thus only its classification result is taken into account in the accumulation process over consecutive images. The final result of a track of supplementary signs is determined via the weighted sum of the single patterns classification results.

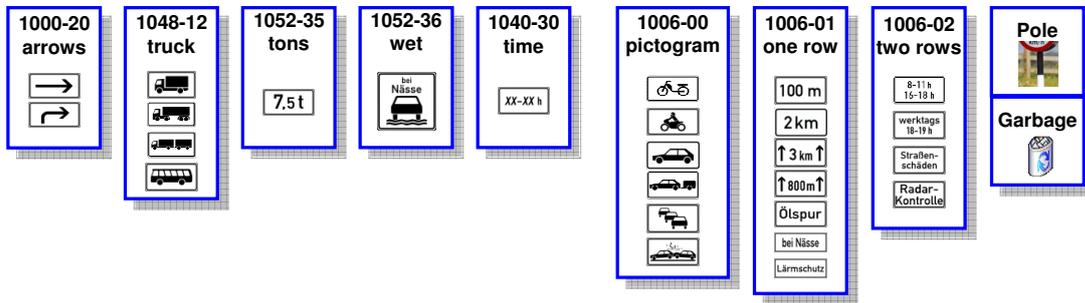


Figure 2.17: Clustered classes of German supplementary signs

Since the appearance of supplementary signs varies greatly in different countries a separate classifier is adapted for each country and the corresponding classifier is used. The country the vehicle is currently travelling in is determined via the GPS system. For some countries no supplementary sign recognition system is necessary, since no supplementary signs restricting the meaning of the corresponding speed limit sign are used in this countries. The differences in appearance are shown in figure A.10 in the appendix.

2.8 Extension Modules for Improving Traffic Scene Interpretation

A complex system like the traffic sign recognition is bound to have certain aspects where the main system of detecting and classifying the objects reaches its limits. This is often due to objects looking exactly like the ones searched for, but having a different meaning, or due to other systematic errors. To include the answer to these problems into the main system would weaken either the detector or the classifier in forcing a decision which can not be made by the given features or including a huge amount of features into the solution, thus slowing the system down. Thus the decision was made to use additional modules to cover these contingencies. By analysing the algorithm the following three main exten-

sion modules were identified as necessary additions to the traffic sign recognition system for the detailed interpretation of the traffic scene:

- The use of the three dimensional position information.
- The use of additional refinement classifiers.
- The fusion with additional inputs beside the camera images and the ego motion.

2.8.1 Use of the Three Dimensional Information

The three dimensional position of the real world traffic sign, when known to some extent is important since the position of the sign in the world has an influence on its meaning and the objects looking like signs in the images taken. Objects having different real world size or not being stationary, on the back of lorries for example. These can be rejected using the algorithm as proposed in section 2.5.3.

2.8.2 Refinement Classifiers

Another necessary additional module, for example in Germany, is a unit capable of distinguishing between *active limit* and *exit* signs. In Germany the exit number on directional indicators is similar to an active road sign in the normalized grey value image, see figure 2.18. This necessitates a module to separate the two types of signs after the classification process. Thanks to the blackboard function of the tracker the not normalized *cutouts* belonging to the track are still available and can be used to check if the sign is an active sign or a passive exit or minimum speed sign. The additional module for this task is run only if the main system decided for the sign to be an *active limit* and thus does not slow down the complete system for all images processed.

The algorithm solving this task has to evaluate the differences of the two objects in question. The main distinction of the *active limit* signs compared to passive signs, apart from their differing colour, is, that they are composed of active light sources whereas the blue exit signs are passive signs and are visible due to reflected light only. This leads to differences in the distribution of grey values in the *cutouts* depicting these signs. The classifier used for the few features is a polynomial classifier again to be able to reuse the existing module. The results of the scheme are presented in the section 6.9.

Should the exposure time and the gain setting of the camera for the current frame be known, this would be a valuable additional information as well, since an approximate absolute intensity value could be derived from the pixel values and the sensor specifications.

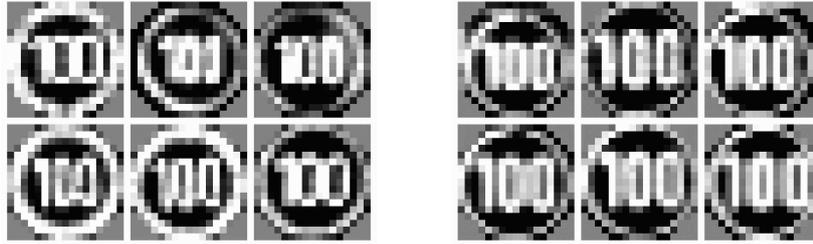


Figure 2.18: Examples of normalized active signs (left) and exit number or minimum speed signs (right)

2.8.3 Fusion with Additional Inputs

The system is designed to be open for the use of additional information from external sources. This thesis will not go into depth as how to use this information to advance the systems performance. The following list mentions the main resources of supplementary knowledge.

- The most important additional source of information is the use of a map. Navigation systems can be used to achieve information about implicit speed limits, upcoming exits, the number of lanes on the current road, or even the speed limits valid at the time of digitalization of the map. This information is not always accurate and ages quite quickly as signs are replaced.
- The laws differ depending on the country visited. Thus the rules effective in the current country have to be represented in the traffic sign system to give the driver the correct information.
- Further sources of hints as to the validity of traffic signs and the currently allowed speed can be achieved from various assistance systems like the lane detection software, the detection of other traffic participants or other systems yielding information about the surroundings of the vehicle.
- Possible future sources of information are beacons transmitting the current speed limit at crucial sections of the road or radio transmissions coupled with on vehicle GPS and map systems. Much work is currently flowing into car-to-car communication allowing several vehicles to combine their knowledge. Also upcoming are on-line updates of outdated parts of the navigation systems map by the use of the on-board recognition systems of multiple vehicles, [Hoehmann and Kummert, 2010].

3 General Classifier Adaptation Approach

One of the most important parts of the thesis is the *internationalization* of the system. The term *internationalization* is used to describe the algorithms necessary for the adaptation to the different characteristics of traffic signs encountered in varying countries. The part of the thesis influenced the most by the differences encountered in different countries is the classifier for the traffic sign pictograms. The detector and tracker are mostly independent from the country the vehicle is currently travelling in.

In this chapter the steps necessary to adapt a classifier and the extensions developed to adapt the classifiers to the appearance types of signs encountered in different countries are described.

3.1 Classifier setup

The classifier system consists of the parts described in chapter 2.6.2 and shown in figure 2.13. The first stage is used for the rejection of *garbage* patterns and the distribution of the classified *cutouts* to one of the six sub classifier types, restriction signs inner or outer circle and end of limit, both for active and passive signs. The second classifier is used to discern the correct speed limit and rejects the remaining *garbage* patterns. The classifier in stage I is kept generic for all appearance types encountered in the different countries the classifier should be functioning in. Of the classifiers in stage II there are more than one version, each version being dedicated to be used in a certain set or cluster of countries. Which classifier to use is determined by the country the vehicle is currently travelling in, derived by the use of the *Global Positioning System* (GPS).

3.2 Obtaining Sample Sets

For the adaption of classifiers sample sets have to be obtained. This usually includes a high labelling and recording effort. For the system to be operative in many countries this would mean an effort not being manageable. This necessitates an automatic or at least semi automatic labelling process with a minimum of human intervention.

The starting point is a large set being recorded in Germany. A small part of the recorded set was labelled and a first classifier adapted. This classifier was used to ease the task of labelling by yielding a correct result in roughly 90% on the up to this point unlabelled set, thus allowing a much faster labelling. The

result was a large labelled set of german traffic sign *cutouts*. This labelled set was used for developing the classifier and adaption scheme presented in this thesis. For the expansion of the systems operability to other countries the algorithm for the generation of *synthetic* signs was developed, see section 4.2.2. This system allows the generation of realistic sample sets from ideal samples obtained from the countries rule books or even pictures in the internet. These templates are geometrically and photometrically transformed to resemble images taken from a moving vehicle. The transformation parameters are learned from samples from the labelled German sample set. This algorithm is presented in the left half of the figure 3.1.

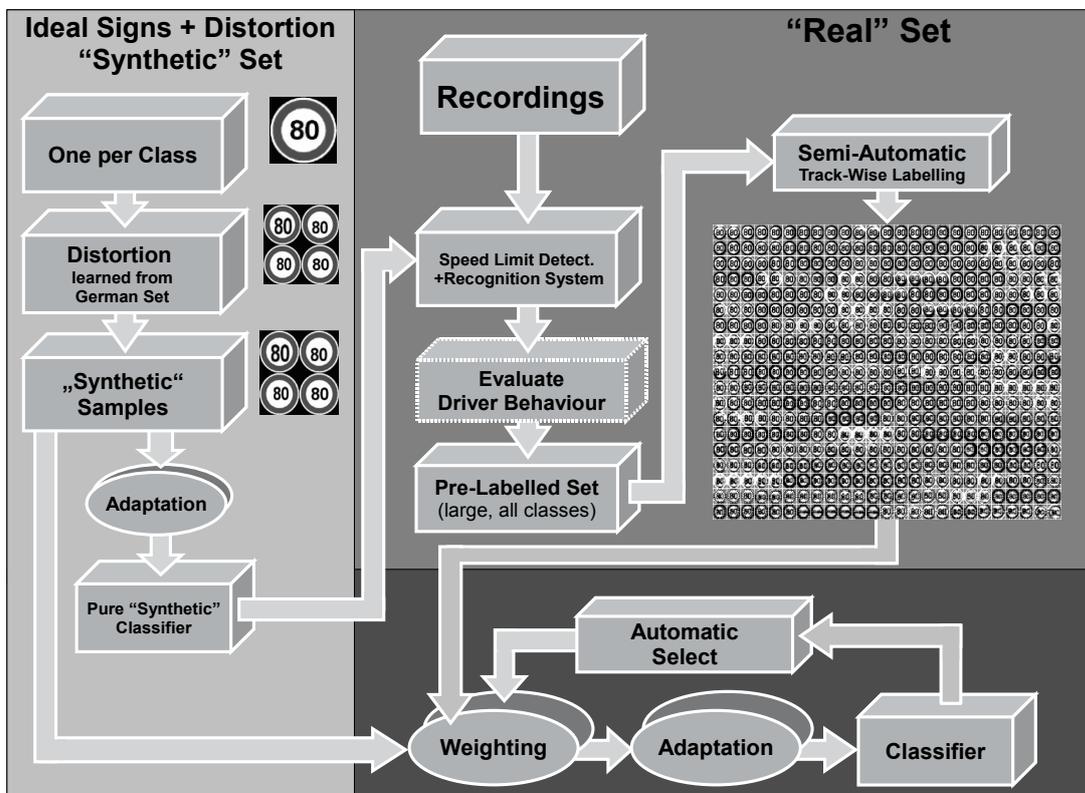


Figure 3.1: Scheme for the creation of the training sets and training of a classifier for a new country the system has to be working in.

3.3 Training of the Classifier for a New Country

To obtain samples in addition to the *synthetic* ones, recordings are made in the different countries. The already existing classifiers adapted with *synthetic* samples are used in the detection and recognition system to label the newly recorded images. As a further hint showing the label of the encountered object

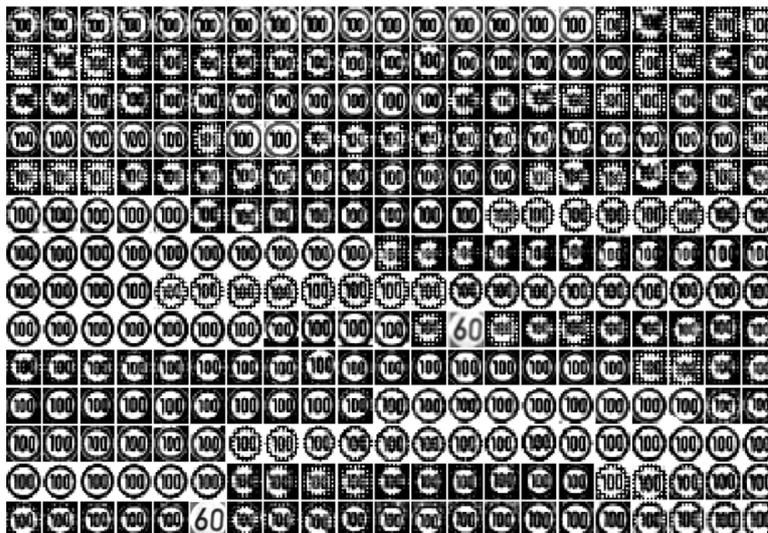


Figure 3.2: Example showing sample tracks pre-labelled by the system. The sample of high resolution and high classifier confidence is taken from a track and displayed as representative. The human labeler can easily correct errors by clicking on the sample, changing the label and thus changing the label of all elements in the particular track. Here two speed limit 60 signs are in the list of speed limit 100 signs.

the behaviour of the driver and thus the vehicle can be used, see section 4.3. The task of the human labeller is to decide if a complete track was classified correctly by the system and, if not, to correct the label. This kind of labelling is very fast, as shown in figure 3.2, but leads to label errors for single patterns when parts of the track do not belong to the same real world object as the rest of the track. Another problem are *cutouts* belonging to the traffic sign object, but being indecipherable due to occlusions or low resolution. Thus there has to be an additional selection process on a single image *cutout* base.

This would mean a very high labelling effort for a human labeller and there is no unambiguous solution, especially in cases of partly covered signs or distorted signs which are on the border of legibility. Due to the high effort otherwise necessary the decision which *cutout* is to be used is made automatically in a bootstrap loop, called *Automatic Select* in figure 3.1. The decision is based on the reconstruction error of the PCA on the single *cutout* and the classification result when using the current classifier, which is either adapted using *synthetic* samples only in the first loop, or the classifier adapted in the last bootstrapping loop. The samples are sorted by these values and the samples showing the highest errors are not regarded in the next bootstrapping adaption step. All remaining samples as well as the *synthetic* samples are used for adapting the next classifier. Using this new classifier all samples, even the ones rejected in the previous loop,

are classified and sorted by their classification values again. This loop is repeated for until no or only few changes in the removed samples occur. More on this subject is presented in section 4.3.1.

3.4 Training of Classifiers for Country Sets

To expand the system to be serviceable in more countries the classifier system has to adapted since the signs show different appearances in different countries. The differences are for instance differing fonts, sizes of the digits relative to the sign, the width of the red rim, the type of diagonal stripe, additional writing in the sign like *km*, see figures 1.3 and 1.4. For a few countries one classifier set per country could be adapted and used, for a large number of countries keeping one classifier set for each would be cumbersome. Thus the classifiers have to be used for a number of appearance types and thus in a number of countries. This means the appearance types and thus the countries have to be clustered, see section 4.4. As a proof of concept all possible compositions of clusters for all number of clusters have been tried. For each cluster of appearance types and thus sample sets classifiers have been adapted and the error rate computed. The results are shown in section 6.7.3. The clusters of countries can be different for each of the six stage II classifiers shown in figure 2.13. To add further countries to the domain the system is functionable in, the available samples of this country, synthetic and real, are classified by the already existing classifiers of the other countries. Based on the result of this classification the country is either added to one of the existing clusters of countries, if the classification results were adequate when using the according classifier or a new cluster could be added, holding the new country as only member. The classifiers for the cluster to which the newly added country was appended is re-adapted using the samples of all countries now being in the cluster if the results were not satisfying with the existing classifier, see figure 3.3.

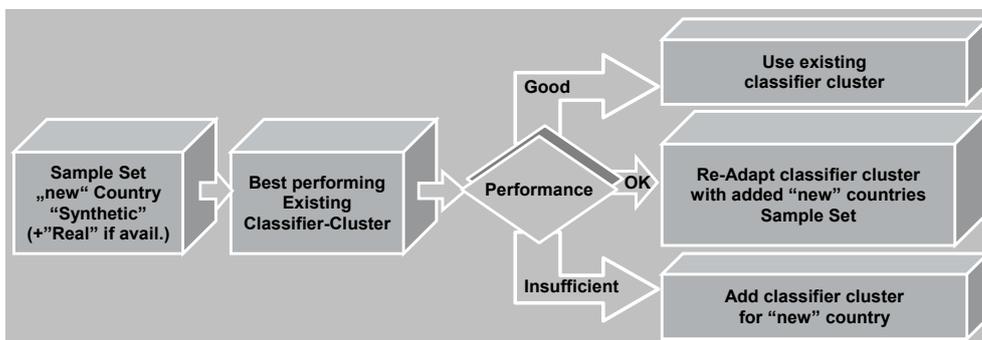


Figure 3.3: Scheme for adding a classifier to a classifier cluster for a new country the system has to be working in.

4 Offline System and Classifier Training

This chapter will explain the design and implementation of the offline part of the classification system. Further the creation and gathering of the training sets as well as the efficient training and bootstrapping of the classifiers is explained in this chapter. The creation of the testing sets is explained in depth in chapter 5.2. The basic design of the online classifier hierarchy used in the traffic sign recognition system is explained in section 2.6 and in figures 2.10 and 2.13.

The generation of training sets for the classifier adaptation requires a module for the creation of realistic traffic sign image samples. This step is necessary since in countries where real sign samples could not be recorded in sufficient numbers, a sample set adequate for the adaptation of a first satisfactory classifier has to be created. The samples have to be synthesised from a small number of recorded signs or from traffic sign icons or templates. For this task a process for distorting these exemplary signs or templates has to be implemented. Since the algorithms found in the literature [Kehtarnavaz and Ahmad, 1995, Escalera and Radeva, 2004] were insufficient for the creation of a realistic training set, a novel algorithm allowing the creation of a satisfactory training set was implemented and is presented here.

The main task of the offline part of the system is the training of the single classifiers and the bootstrapping of these classifiers. The composition of the training set is an integral part of the classifier adaptation. As classification algorithm the polynomial classifier [Ott, 1977] was chosen. The advantage of this algorithm beyond its classification and generalization capabilities is the very short readaptation time necessary when adding new samples as shown in chapter 4.3.1. The same scheme used is valid with other classifier types as well, but the time necessary for verification would increase unacceptably, especially where the multitude of classifiers for the different countries are considered.

4.1 Classifier Design

The classifier used is a polynomial classifier with a principal component analysis as feature reduction step. The formulas describing the algorithm are taken from [Schuermann, 1977] and [Schuermann, 1996b].

The principal component analysis, or Karhunen Loewe transform minimizes the reconstruction error of a high dimensional Gaussian distribution when linearly projecting it into a lower dimensional orthonormal space [Pearson, 1901]. The following equations show the transformation algorithm.

$$\text{Expectation vector } \bar{\mathbf{v}} = \mathbf{E} \{ \mathbf{v} \} \quad (4.1)$$

$$\text{Moment matrix } \mathbf{M} = \mathbf{E} \{ \mathbf{v} \mathbf{v}^T \} \quad (4.2)$$

$$\begin{aligned} \text{Covariance matrix } \mathbf{K} &= \mathbf{M} - \mathbf{E} \{ \mathbf{v} \} \mathbf{E} \{ \mathbf{v} \}^T \\ &= \mathbf{E} \left\{ (\mathbf{v} - \bar{\mathbf{v}}) (\mathbf{v} - \bar{\mathbf{v}})^T \right\} \end{aligned} \quad (4.3)$$

$$\text{Eigenvalues of } \mathbf{K} \quad \lambda = \{ \lambda_1, \lambda_2, \dots, \lambda_N \} \quad (4.4)$$

$$\text{Eigenvectors of } \mathbf{K} \quad \mathbf{B}_N = \{ \mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_N \} \quad (4.5)$$

$$\text{Transformed vectors } \mathbf{w} = \mathbf{B}_M^T (\mathbf{v} - \bar{\mathbf{v}}) \quad (4.6)$$

$$\text{Reconstructed vectors } \hat{\mathbf{v}}_M = \mathbf{B}_M \mathbf{w} + \bar{\mathbf{v}} \quad (4.7)$$

$$\begin{aligned} \text{Mean reconstruction error} \quad \bar{R}_M^2 &= \sum_{i=M+1}^N \lambda_i / \sum_{i=1}^N \lambda_i \end{aligned} \quad (4.8)$$

$$\begin{aligned} \text{Reconstruction error for one sample} \quad R_M^2 &= \sum_{i=1}^N (v_i - \hat{v}_{Mi})^2 / \sum_{i=1}^N v_i^2 \end{aligned} \quad (4.9)$$

$$\text{Whitened transform } \mathbf{w}' = \{ w_1/\lambda_1, w_2/\lambda_2, \dots, w_M/\lambda_M \}^T \quad (4.10)$$

In equations 4.1 to 4.8 \mathbf{v}_n represents one data vector of dimension N , in our case the grey values of the normalized *cutout* raster image. $\mathbf{E} \{ \mathbf{v} \}$ or $\bar{\mathbf{v}}$ is the expectation vector of all I available data vectors. λ and \mathbf{b} being the eigenvalues and eigenvectors of the covariance matrix sorted by the magnitude of their corresponding eigenvalues, the highest being the first. \mathbf{w} is the M -dimensional input vector \mathbf{v} transformed into the M -dimensional subspace defined by the first M eigenvectors. The mean reconstruction rate can given as \bar{R}_M^2 , meaning that over the complete set of N input vectors the reconstruction value averages to \bar{R}_M . The reconstruction error for a single sample is R_M^2 as depicted in equation 4.9.

An additional step can be taken by dividing each entry in the \mathbf{w} vector by the according eigenvalue, thus normalizing the distribution of all dimensions in the transformed vector to a zero mean, as defined by the principal component analysis, and a standard deviation of 1. This approach is called *whitening*. This step is necessary only for an easier implementation of a fixed point computation for the polynomial classifier in the step following the principal axis transform. On most integrated control units fixed point computation is — by orders of magnitude — faster than floating point computation, thus this step has to be kept in mind when designing the traffic sign recognition system.

The classifier used for the analyses is the polynomial classifier. This classifier,

as described in [Schuermann, 1996b], is based on the discriminant function

$$\begin{aligned}
 \mathbf{d}(\mathbf{w}) &= \mathbf{A}^T \mathbf{x}(\mathbf{w}) \\
 \mathbf{x}(\mathbf{w}) &= \left(\underbrace{1, w_1, \dots, w_N}_{\text{linear}}, \underbrace{w_1^2, w_1 w_2, \dots, w_N^2}_{\text{complete quadratic}} \right)^T \\
 \dim \{\mathbf{x}\} &= \begin{pmatrix} N + G \\ G \end{pmatrix} \tag{4.11} \\
 &\quad - \text{ for the applied complete quadratic setting}
 \end{aligned}$$

In the ideal case the entry d_k in the discrimination vector \mathbf{d} is 1 for the class the object having the feature vector belongs to and 0 for all other classes, thus being identical to the target function \mathbf{y} . The optimization of the matrix \mathbf{A} in the training of the classifier minimizes the Euclidean error between the ideal discrimination vector \mathbf{y} and the resulting vectors \mathbf{d} overall training samples. The polynomial structure vector $\mathbf{x}(\mathbf{w})$ contains all possible polynomial combinations up to the polynomial degree G . The dimension thus increases very fast with the number of dimensions G . This limited the tests that were performed to quadratic classifiers, i.e. $G = 2$.

$$\mathbf{y}_k(\mathbf{w}) = \begin{cases} 1 & \text{for } \mathbf{w} \text{ being from class } k \\ 0 & \text{otherwise} \end{cases} \tag{4.12}$$

$$\mathbf{S}^2 = E \left\{ \|\mathbf{y} - \mathbf{A}^T \mathbf{x}(\mathbf{w})\|^2 \right\} = \min_A \quad , \text{ leading to :} \tag{4.13}$$

$$\mathbf{A} = E \{ \mathbf{x} \mathbf{x}^T \}^{-1} E \{ \mathbf{x} \mathbf{y}^T \} \tag{4.14}$$

The goal of the quadratic optimization shown \mathbf{S}^2 is to minimize the expectation of error made between the target function \mathbf{y} and the decision vector \mathbf{d} . The matrix \mathbf{A} is the one used in equation 4.12 used to compute the decision vector from the feature vector. To allow the repeated computation of the matrix \mathbf{A} with different a priori weights of the classes or subclasses it suffices to keep the moment matrices $E \{ \mathbf{x}_k \mathbf{x}_k^T \}$ and the vector $E \{ \mathbf{x}_k \mathbf{y}_k^T \}$ for each class and subclass k . This will be used for the creation of the classifiers starting from few or purely *synthetic* samples as described in the following section 4.3.

$$\text{Max criterion} \quad d_{k_{max}}(\mathbf{w}) \geq t_{max} \tag{4.15}$$

$$\text{Diff criterion} \quad d_{k_{max}}(\mathbf{w}) - d_{k_{2nd}}(\mathbf{w}) \geq t_{diff} \tag{4.16}$$

$$\text{RAD criterion} \quad RAD = \|\mathbf{y}_{k_{max}}(\mathbf{w}) - \mathbf{d}(\mathbf{w})\| \leq t_{RAD} \tag{4.17}$$

The decision for a class or for accepting the result of the classification at all is

based on the decision vector \mathbf{d} . The first rule in equation 4.15 decides by finding for the highest component d_k in the decision vector \mathbf{d} and decide for class k . This value has to be higher than the threshold t_{max} . The second rule for accepting a classification result is the minimum difference between the highest and second highest output value in the decision vector. Should this difference be too small, meaning smaller than t_{diff} , no safe decision for one class can be made. Where the first two rules observe the highest values in the decision vector only, the third rule reviews the plausibility of the complete decision vector. The idea behind the third rule is that the classifier was optimized to minimize the Euclidean distance between the decision vector \mathbf{d} and the target vector \mathbf{y} . Should the Euclidean distance between these two be too high, higher than t_{RAD} , the reasoning is that no sample resembling the current one was in the training set. Thus the decision of the classifier for this sample cannot be trusted and the classification result has to be rejected. All three rules are implemented as described in [Schuermann, 1996b].

4.1.1 Training of the Principal Component Analysis

The only free parameter in this transformation is the number of dimensions after the transformation. In our case the number of dimensions (M) was chosen to allow for an approximate 90% reconstruction rate ($1 - \overline{R}^2$). This rate was determined by using the transformation and retransformation on a multitude of normalized traffic sign *cutouts* (chapter 2.6.1), and deciding if the retransformed image was still decipherable by the human beholder. To allow efficient pipelining and for a faster computation the number of dimensions used in the final system had the additional requirement of being dividable by the factor eight. Each classifier in the hierarchical setup has its own principal component analysis and thus the dimensionality of the classifiers may differ. The input for the adaptation of the principal component analysis were taken from the positive set only, since a high number of *garbage* elements which introduce noise into the process and would thus have an adverse effect on finding principal axes in the distribution.

The importance of different classes is regarded by computing a moment matrix for each class or subclass and adding the matrices according to the selected weighting. By remembering the number of contributing elements for each matrix the moment matrices can be updated without high computational effort, thus allowing bootstrapping in acceptable adaptation time (equation 4.2).

For the training and application of the classifier the use of the *whitening* mentioned in chapter 4.1 has no effect besides numerical stability. When preparing the transform and the following classification for a use of fix point computation, as it is necessary for many digital signal processors, the *whitening* is a necessary step to assure that the input values of the polynomial classifier are of the same order of magnitude. Without this step the values might differ by some orders of magnitude, for the classifiers in our case mostly two orders for the first fifty eigenvalues.

4.1.2 Training of the Polynomial Classifier

The equations necessary for the training of the classifier have been introduced in equation 4.12-4.14. The complete set of classifiers consists of seven PCA and polynomial classifier sets. One classifier, stage I, called *cluster classifier* distributes the samples to the sub classifiers or rejects the samples as not depicting signs at all (section 2.6.2, figure 2.13). The six pictogram classifiers in stage II have to be trained in succession to the cluster classifier, since the assignment of the samples used in the training of these sub classifiers is based on the result of the cluster classifier. Should the cluster classifiers result indicate for the sample to most likely be a garbage element the sample is not propagated to the second stage.

All normalized and PCA-transformed samples serve as input for the training of the polynomial cluster classifier. Once this classifier is trained the classifier is applied upon all input samples, thus dividing the complete set into the six sub types plus rejecting a number of them by classifying them as *garbage* or rejecting them using the *RAD criterion*.

The six pictogram classes actually define just four different types of signs, active and passive limits and end of limit signs. The differentiation into *inner* and *outer* is owed to the indistinctiveness of the detector which does not discern between the inner and outer contours of a traffic sign. For the *end of limit* signs this distinction is unnecessary since the difference between finding the circle on the sign and finding the outer contour of the complete sign is small enough to be ignored when training the classifier. Samples being labelled as belonging to one of the four types, but classified by the cluster classifier as belonging to a different class or being a garbage sample, are not used in training the pictogram classifiers as they are assumed to be mislabelled or illegible. This reduces the size of the training set for the pictogram classifiers. The cluster classifier is used to divide the samples labelled as active or passive limit into *inner* or *outer* for the training of the pictogram classifier as well. *Garbage* samples being misclassified as belonging to one of the four sign types treated in the pictogram classifiers are used in training these classifiers in stage II. As for the training of the principal component analysis the samples belonging to different classes or sub classes can be weighted separately, allowing to add importance to classes having a low a priori probability or allowing to mix samples from different recording conditions, e.g. day and night, weather conditions or road types with different weighting. The reweighting of the moment matrices computed in the process of training the classifier allows to easily adapt the weight factors.

4.2 Classifier Internationalization

As introduced in section 1.5.3 one necessary step in a successful traffic sign recognition system is an algorithm for the convenient adaptation of the system to the

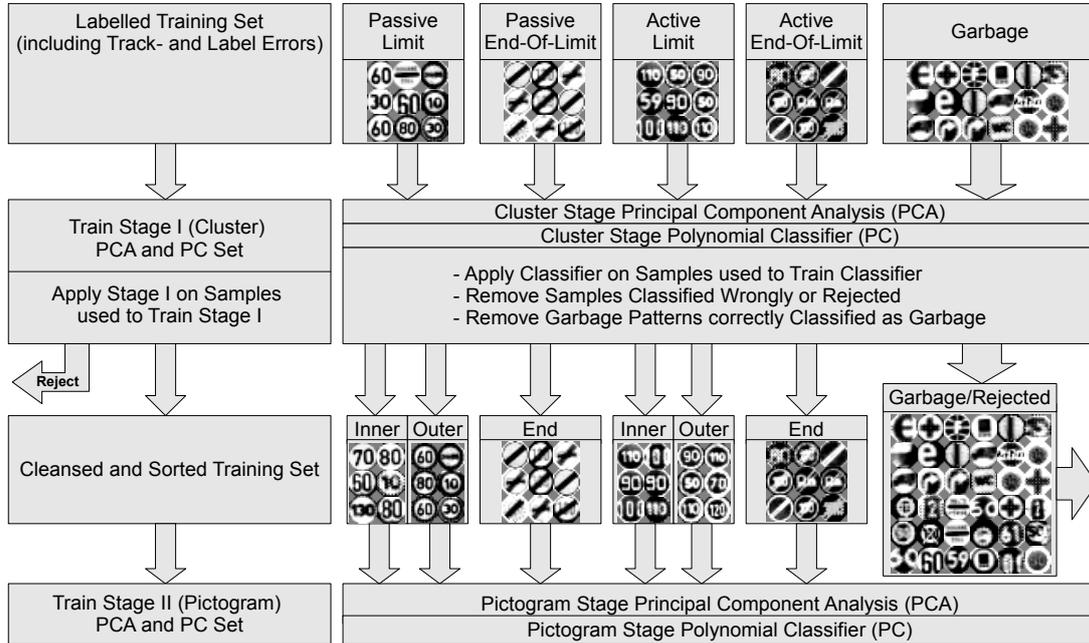


Figure 4.1: Scheme for the Flow of the Labeled Samples in the Classifier Training

differences in the appearance of traffic signs encountered in different countries. In section 4.1 the adaptation of the classifier for a single country is described.

In this thesis the system is evaluated on the signs of ten West-European states, these being Austria (AT), Belgium (BE), Denmark (DK), France (FR), Germany (DE), Italy (IT), Luxembourg (LU), Netherlands (NE), Spain (ES) and Switzerland (CH). The signs are depicted in section 1.2.

One possibility to expand the operational range to additional countries is to add normalized traffic sign samples from the countries that should be covered by the traffic sign recognition system to the process described above and keep a single classifier hierarchy for all countries. The advantage of this scheme is the low memory use and the system not depending on the information in which country it is currently applied.

Another possibility is to repeat this algorithm for each country the system should be functionable in. As explained in section 1.5.3 this leads to a comparatively large classifier system while supposedly delivering the best classification results, see section 6.7.

A third possibility is the use of several sets of classifiers, where each set of classifiers covers a number of countries in which the signs have an appearance similar enough to allow this combination without losing overly much classification performance.

The approach used is a modification of the third possibility. In the hierarchy of classifiers explained in section 2.6.2 the top level is used to reject samples

not resembling traffic signs at all and to distribute the remaining samples for further inspection by the respective sub classifiers, see figure 4.1. This top level classifier is trained with samples from all countries which should be covered by the traffic sign recognition system and thus has a lower rejection capability than a set of classifiers specialised on a single country each, but generalizes better than a classifier adapted on such a set.

The distribution of samples into the second stage of the hierarchy is not hampered by this scheme as shown in section 6.7 since the variation in appearance of the different types of signs is too large to allow frequent confusion between the classes even for a more generalizing classifier. On the pictogram level the appearances of the signs in different countries vary too much to use a single classifier for all of them without loss of performance. This is especially true when subtle details like the difference between digits closely resembling each other in the given low resolution determine the meaning of the sign. Due to this fact several classifiers were trained on the pictogram level, each of them covering a number of countries where this type of signs is similar in appearance, see figure 4.2. The number and composition of these *clusters* of countries in the pictogram classifiers depends on the differences in appearance of the respective sign type and the size of memory allowance for the classifiers limiting the number of classifiers. The scheme is explained in figure 4.3.

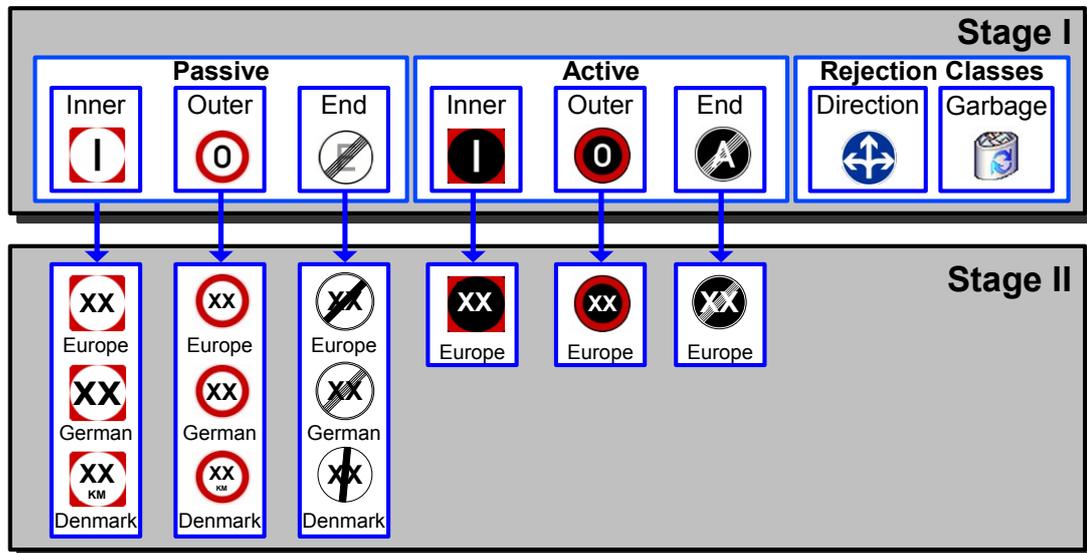


Figure 4.2: Exemplary hierarchical classifier setup as in figure 2.13, adapted for different appearance types, the respective Stage II Classifier is chosen according to the appearance of the signs in the country the vehicle is currently travelling in.

To find the first complete set of classifier clusters as described above a complete search for all possible combinations is executed on the samples of a restricted

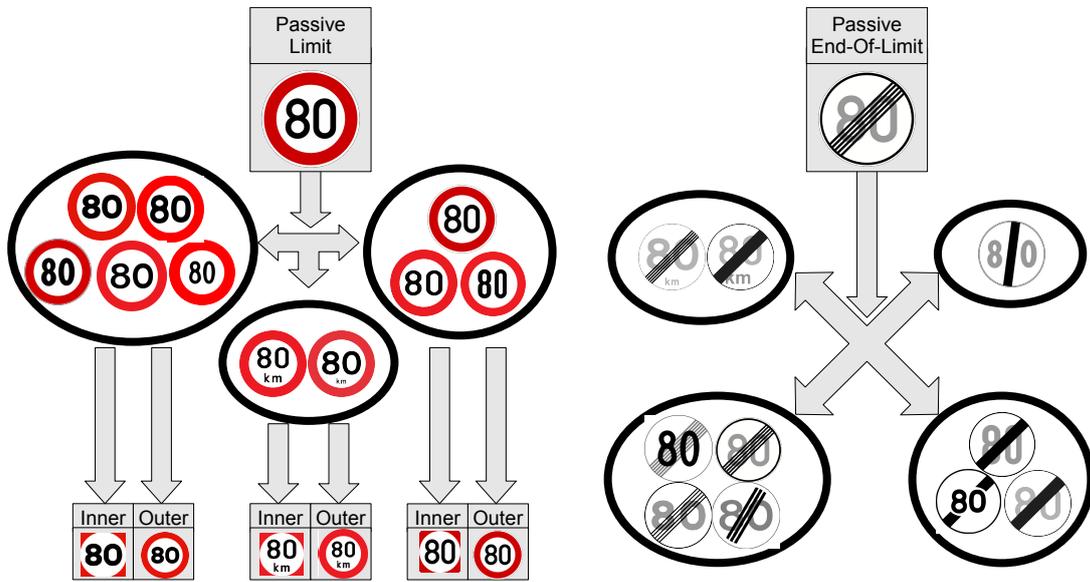


Figure 4.3: Exemplary distribution of signs from different countries to specialized pictogram classifiers

number of countries. Since the number of possible combinations would require very large tests for high numbers of countries the following after the first ten are decided by applying the existing classifiers on the samples from the country to be added and assigning the samples of the new country to the classifier performing best on this set. Should none perform sufficiently accurate the samples are inspected if they are different enough to require a new pictogram classifier, like in the set of ten countries for instance the Italian *end of limit* signs, which show a near vertical black bar which is very different to the appearance of the *end of limit* signs in the rest of the ten states reviewed. The examinations done for this goal are described in section 6.7. The results were published in [Koch, 2007], a diploma thesis which was supervised by the author of this thesis.

4.2.1 Obtaining Training Samples

Most countries have rulebooks describing the font, size and placement of the digits in the traffic sign, thus the appearance of the signs in the given country is known and can be used in the process for the creation of *synthetic* samples, see section 4.2.2, and the training of classifiers from scratch as described in section 4.3. The use of these *synthetic* samples enables the developer to have a functional set of classifiers for each country, even if there is no recorded sample set, or none of sufficient size to adapt a classifier set. If there are no rule books for a country and to check the accuracy of the rule books the internet can be a valuable source of information. Using a choice of keywords pictures taken on holidays or pictures from services showing sections of roads can be used for this purposes.

All this can be done without even having a camera. Samples from the country the developers are living in are easy to be obtained by simply taking pictures of the signs encountered every day. The farther away the countries that should be covered by the system, the harder it gets to obtain information about the appearance of the signs. For countries close by, day trips can be used for the exploration of the appearance of the signs, best recorded with the camera to be used in the final traffic sign recognition system.

There are several helpful hints on what to consider when obtaining sequences of the signs encountered during the sample gathering drives, especially when keeping in mind, that the expense in manpower and necessary hardware should be minimized.

- Simply recording of all frames is easiest to implement, but leads to huge amounts of data, approximately 40 GB per hour for the given setup and thus was done for very few sequences only.
- Record all long tracks of circular objects detected by the system, meaning tracks spanning many consecutive images. Those tracks belong to circular objects encountered in the given country and thus constitute objects for possible false positives once the classifier is adapted to the countries characteristics. These objects might be signs or advertisements typical for this country, but not existing in other countries and thus the classifier has to be trained to discern between these and the traffic signs to be recognized.
- Tracks where at least one element in the track has been classified as belonging to one of the traffic sign classes by the cluster classifier in stage I should be recorded, since objects passing this test resemble traffic signs enough to be possible false positives when the classifier is retrained and could be false negatives of the current system used to record in the given country and thus be valid *speed limit* or *end of limit* signs.
- For each of the tracks to be recorded due to one of the rules above record a few frames ahead of the frames in the track should be recorded to allow the traffic sign detection system to adapt to the conditions like contrast and brightness and set up the internal filters.
- When recording any of the tracks recorded due to the rules above record a few frames after the images in the track should be recorded to allow for *broken tracks*, meaning real world objects of which more than one track was created due to missing detections in more than two frames in the track.
- Of course there should be a manual trigger to allow the driver, or if present, a system operator, to record a sequence. This might be necessary in the rare case that none to the triggering mechanisms above did automatically record the scene, for instance if a completely new type of sign is encountered.

4.2.2 Synthetic Samples

In this section the newly developed and implemented scheme for the creation of sample sets for the adaption of classifiers is explained. The algorithm and experimental results have been published in the diploma thesis [Hoessler, 2007] which was co-supervised by the author of this dissertation and in [Hoessler et al., 2007]. This algorithm is necessary for the adaption of a classifier when no or an insufficient number of example samples for a new class of signs or appearance type of signs has been recorded and labelled. This occurs for instance when extending the usable area of the traffic sign recognition system by another country. The commonly used method is using one or a few ideal samples, for instance from the rule books for traffic sign production and placement or from a photographed sample. These few samples are replicated using geometric and photometric transforms, thus creating a larger sample set for the training of the classifier. The types of transformations used are explained in section 1.5.3.

In the literature the parameters used for the transformations are arbitrary or guessed by an experienced human operator. The main idea is the emulation of the transformations the *cutout* image was undergoing from the real world sign to the pixel raster image. Since the transformation parameters occurring are hard to guess even for an experienced user they are learned from a sample set. The algorithm presented in this thesis utilizes a large labelled set of samples belonging to a common sign class, e.g. speed limit 80. The parameters for the transformations necessary to create each of the samples from an ideal model are stored as a probability distribution and can be used to create new samples from an ideal model belonging to a different class or to a different sign appearance type. Geometric and photometric transformations were used in the algorithm. The geometric transformations considered in this thesis are translation of the sample $t_{u,v}$, the scaling $s_{u,v}$ and rotation in the image plane α . The shearing parameters $h_{u,v}$ are neglected since the samples did not show a measurable shearing component due to the camera system and the detector used restricting the possible transformations applied on the *cutouts*. The equation applied for the affine geometric transformation reads as follows:

$$M(t, A) = \begin{pmatrix} a_1 & a_2 & t_u \\ a_3 & a_4 & t_v \\ 0 & 0 & 1 \end{pmatrix} \begin{cases} a_1 = s_u \cos(\alpha) + h_u s_v \sin(\alpha) \\ a_2 = s_v \sin(\alpha) + h_v s_u \cos(\alpha) \\ a_3 = -s_u \sin(\alpha) + h_u s_v \cos(\alpha) \\ a_4 = s_v \cos(\alpha) + -h_v s_u \sin(\alpha) \end{cases} \quad (4.18)$$

This transformation is used pixel wise to construct a transformed from an original image.

$$Pix_{\text{transformed}} = M(t, A) Pix_{\text{original}} \quad (4.19)$$

The photometric transformation is set by the equation:

$$I_{photometric} = a I_{affine} + b \quad (4.20)$$

The parameters are describing the contrast on the sign and b the base luminance. Due to the small size of the samples after the normalization process the effects of the lenses point spread function (PSF) can be neglected. Other photometric effects like non-uniform illumination, shadings or specular reflections could be included in the model in case the reconstruction from equation 4.22 could not be solved satisfactory, but were not necessary for the traffic sign case. The new step is to estimate the parameter set from a labelled set of samples belonging to one class. Each sample's image is assumed to be formed by the ideal original on which the above geometric and photometric transforms are applied.

$$I_{observed} = T(\Phi) I_{ideal} = T(\mathbf{t}, \mathbf{s}, \alpha, a, b) I_{ideal} \quad (4.21)$$

with Φ being dependant on \mathbf{t} , \mathbf{s} , α , a, and b

The optimization problem to be solved for retrieving the transformation parameters is set as follows:

$$\Phi = \arg \min_{\Phi} E(\Phi) \text{ with } E(\Phi) = \|I_{observed} - T(\Phi)I_{ideal}\|^2 \quad (4.22)$$

The equation is solved by the use of an optimizer, in this thesis the Levenberg-Marquardt algorithm was used. The application of this algorithm on a large set of samples generates an equally large set of transformation parameters Φ . The set of Φ is used as probability distribution $P(\Phi)$.

$$I_{synthetic} = T(\Phi) I_{ideal} \quad (4.23)$$

with $T(\Phi)$ being drawn from the previously determined probability distribution $P(\Phi)$. An *ideal* sign is transformed with the parameter set Φ and compared with a *real* sign. The parameter set is optimized until both signs resemble each other and the optimization converges. The parameter set used in the last transformation is saved. If the transformation/comparison loop does not converge, the *real* sign either belongs to a different appearance type than the *ideal* sign or the transformation function does not cover the type of conversion that transformed the *real* sign to the one perceived in the cameras image. If a different appearance type is the cause of the inability to reconstruct a sample an additional sample set for this appearance type can be added to the *synthetic* sample set. Should the reconstruction fail due to a missing type of transformation, this type of transformation could be added to the set of transformations adapted if it seems to be relevant, meaning reappearing and not being a singular case. For each *real* sign that could be reconstructed one parameter set is saved. From the parameter sets a probability distribution of the parameters can be created, for instance by

creating a histogram of all detected transformation parameters and normalizing it by dividing through the number of all sets of transformations. Some parameters are not independent, as can be detected by correlating the value pairs for these parameters. This is true for the s_u and s_v scaling parameters for instance, due to the circular form of the signs. The correlation of those parameters has to be taken account for by using combined histograms for the computation of the probability distribution. In the recall phase the *ideal* sign template of a different class or appearance type is transformed either with one of the parameter sets Φ previously computed or with a parameter set drawn from the computed probability distribution.

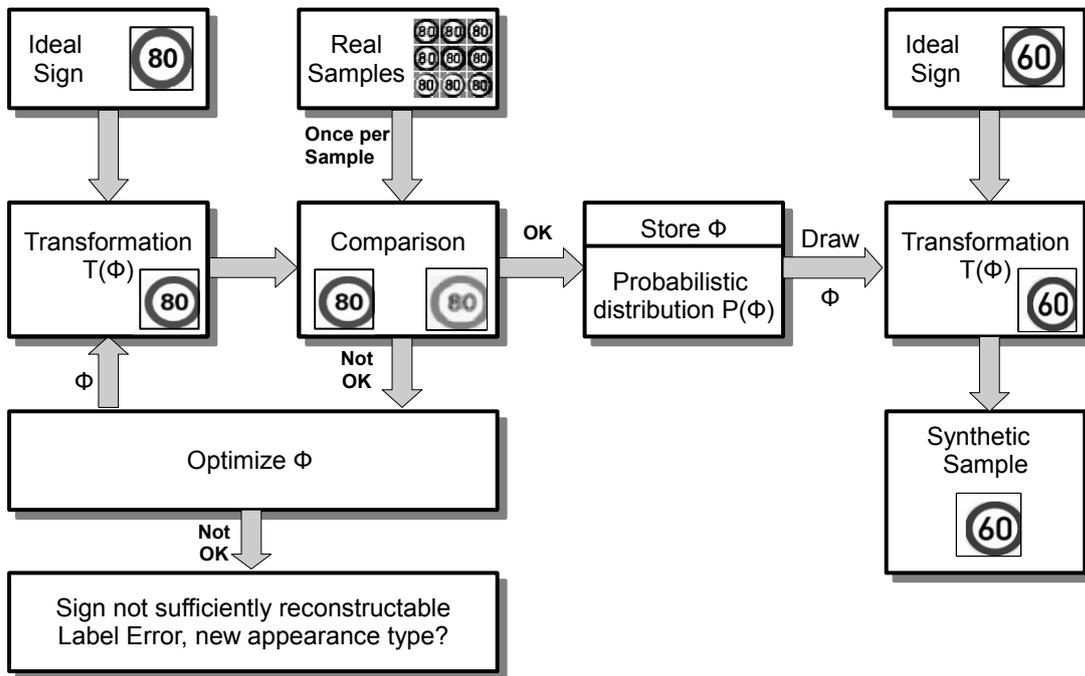


Figure 4.4: Scheme for the creation of synthetic signs. Transform an ideal template in an transformation — comparison — optimization loop until the difference in the comparison with a real sample is small enough. Keep the transformation parameter set, then compare with next real sample. If similarity in comparison could not be reached, show real sample to human operator for decision if additional appearance type or transformation type is necessary. For creating a sample set from another template loop the scheme on the right part of the diagram.

To enhance the practical use of the algorithm, the backgrounds, especially for the *cutouts* representing the outer contour of the signs, are varied as well, since when part of the transform is a translation a small part of the background can appear in the otherwise background independent circular sign area.

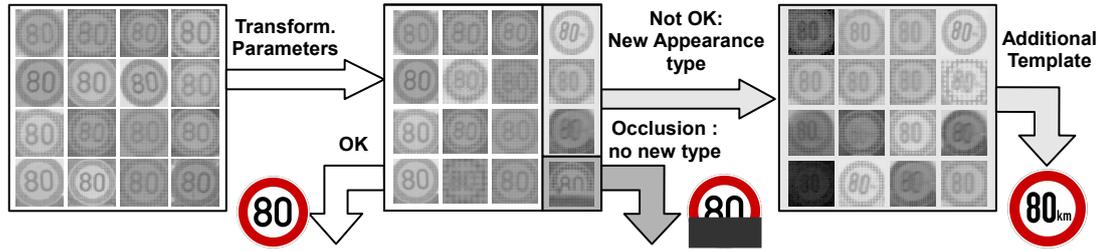


Figure 4.5: Identifying new sign appearance types for the generation of *synthetic* samples. If a fitting transformation from the ideal template to the current real sample was found as depicted in figure 4.4 the transformation parameters are saved (Ok). Otherwise a human operator decides if the sign is illegible or occluded, in which case the attempts for reconstruction are stopped. When a reappearing new appearance type is found, these real world signs are used to create a new ideal template fitting for this type and removed from the list of real samples to be reconstructed with the original ideal template.

4.2.3 Knowledge-Based Autonomous Sample Collection

One possibility of meeting the diverse goals of internationalizing traffic sign recognition elaborated in this dissertation is using driver and system feedback. The system should be able to learn from its inputs and the drivers reactions what the infrastructure in the current situation was requiring of the driver. For instance in a country where the system is still not completely adapted a speed limiting sign will still be detected due to its circular form, but the inscription might be unreadable for the system because of a different font or placing of the digits used. The attentive driver will still abide — at least partly — to the encountered sign. This - in context with other on-board sensors giving hints concerning the behaviour of other vehicles in the vicinity and road curvature, etc. — can be used to add the detected, but unrecognised, signs to a watch-list that is later on used to fine tune the classifier. This list can be refined by keeping only those elements in the list that suffice some requirements, such as length of the track, classification result of the first stage of the classifier hierarchy or behaviour of the driver when encountering the object.

Another possible step is to use classifiers trained on specific appearance types or country specific appearances from other countries to check if the newly encountered object fits into one of these appearance types and can be classified successfully by one of those classifiers. This additional computation can either be done in the running control device when there is free computation time due to fewer detected circles or when the vehicle is stopping, when there are no new object detections to be expected, or the computation is performed offline on the collected data, thus requiring more storage capability in the collecting device.

The two ways of utilising the information are firstly unsupervised adding of the pattern in question to the learning set and including it in the current classifier or secondly remembering the pattern and showing it to an expert — who might even be the driver — to be able to incorporate it to the set of patterns used to create the next version of the classifier. The first solution has the advantage of not encumbering a human with having to teach the classification system and having a higher update rate, while the second possibility has less necessity of having a system being able to cope with wrongly labelled data.

4.3 Design and Implementation of a System for the Automation of Sample Collection

As stated in section 4.2, the use of more than one classifier will be necessary to cope with the variability of the fonts and sizes used for classifying the signs. The more countries have to be covered, the more attention has to be placed upon automatizing the whole process. Using a recording device as described in 4.2.1 the recording of sequences showing traffic signs is not a strenuous task, since the operator just has to start the recorder at the begin of the tour and copy the resulting sequences to the servers database after returning. Thus holiday and business trips were used to gather data from Germany and the surrounding countries and of course vehicles could be equipped without the driver having to operate the system at all. The more work intensive part is the labelling process for the obtained data. To keep the need for a human in this part of the system as low as possible the following scheme was developed and can be followed in figure 3.1:

1. Obtain initial sample prototype images for one country set by using the ideas from section 4.2.1 like using rule books or holiday pictures of the signs. Then create a first training set by creating *synthetic* samples based on the prototypes found, using a plausible distribution of the distortion parameters used, see section 4.2.2.
2. Create classifier set based upon the first training set created in the first step.
3. Label samples of one frequent sign type, for example in this thesis the passive sign *speed limit eighty* was used, in larger quantities by recording many occurrences of this sign at different lighting conditions, different weather conditions and varying relative positions to the sign and varying velocities of the vehicle. Since the classifier is still unsatisfactory at this point of the algorithm, some signs have to be relabelled in this stage.
4. From this set of samples for one sign type, compute the set of parameters for the adaptation of the input parameters of the process for creating a

better set of *synthetic samples*. Create a complete *synthetic* sample set of all sign types using the algorithm described in section 4.2.2.

5. A complete classifier set is adapted using the enhanced *synthetic* sample set. Should there be no recordings of traffic signs beyond the ones used to create the parameter set for enhancing the parameter set for the creation of the *synthetic* samples, the algorithm ends here.
6. Use the enhanced classifier trained in the previous step to automatically label the signs in all sequences recorded on track base, meaning giving one tracked object the label the traffic sign system decided is the most likely.
7. Either a human operator checks and corrects the labels on the complete tracks by looking at the last detected circles — Or the label is accepted or discarded based on the scheme presented in section 4.2.3.
8. With the enhanced *synthetic* sample set and the real samples derived by processing the sequences recorded the final classifier is trained. An additional step to improve the classification performance is the selection and bootstrapping algorithm explained in section 4.3.1.

Once the above scheme has been executed for the set of one country instead of having to create a *synthetic* set before the enhanced *synthetic* set is unnecessary since the distortions introduced by the imaging system are not dependant on the country it is used in, thus one can start at the fourth step with the creation of the enhanced *synthetic* sample set for every following country.

As shown in section 4.2 the algorithm for creating classifier sets usable in more than one country includes the weighted merging of the samples created for, or gathered from different countries. Both the adaptation of the principal axis transform and the training of the polynomial classifier allow a simple method for this weighted merging of the samples by the weighted adding of the moment matrices created by the features of the samples as can be seen in equation 4.2 and 4.14. The additional matrix $E\{\mathbf{xy}^T\}$ necessary for the training of the classifier contributes one additional vector per class and characteristic sign type to be stored as seen in equation 4.24.

$$E\{\mathbf{xy}^T\} = \{p_{c1}\boldsymbol{\mu}(\mathbf{x}_{c1}), p_{c2}\boldsymbol{\mu}(\mathbf{x}_{c2}), \dots, p_{ck}\boldsymbol{\mu}(\mathbf{x}_{ck})\} \quad (4.24)$$

In the expectancy value matrix $E\{x, y^T\}$ the column vectors depend on the mean vector of this class, possibly a weighted mean from different appearance types and a weight factor p_{ci} accounting for the a priori probability of the respective class. Keeping the complete set of moment matrices for a single class requires $(\dim\{\mathbf{x}\}^2 + \dim\{\mathbf{x}\})$ Bytes/Value Bytes, in the case examined this amounts to approximately 3MB $((861^2 + 861) * 4 \text{ Bytes})$. The diversity of the sign appearances in some countries make the storing of a second or third set of matrices

preferable, thus allowing the control over the a priori weight the samples for this appearance type.

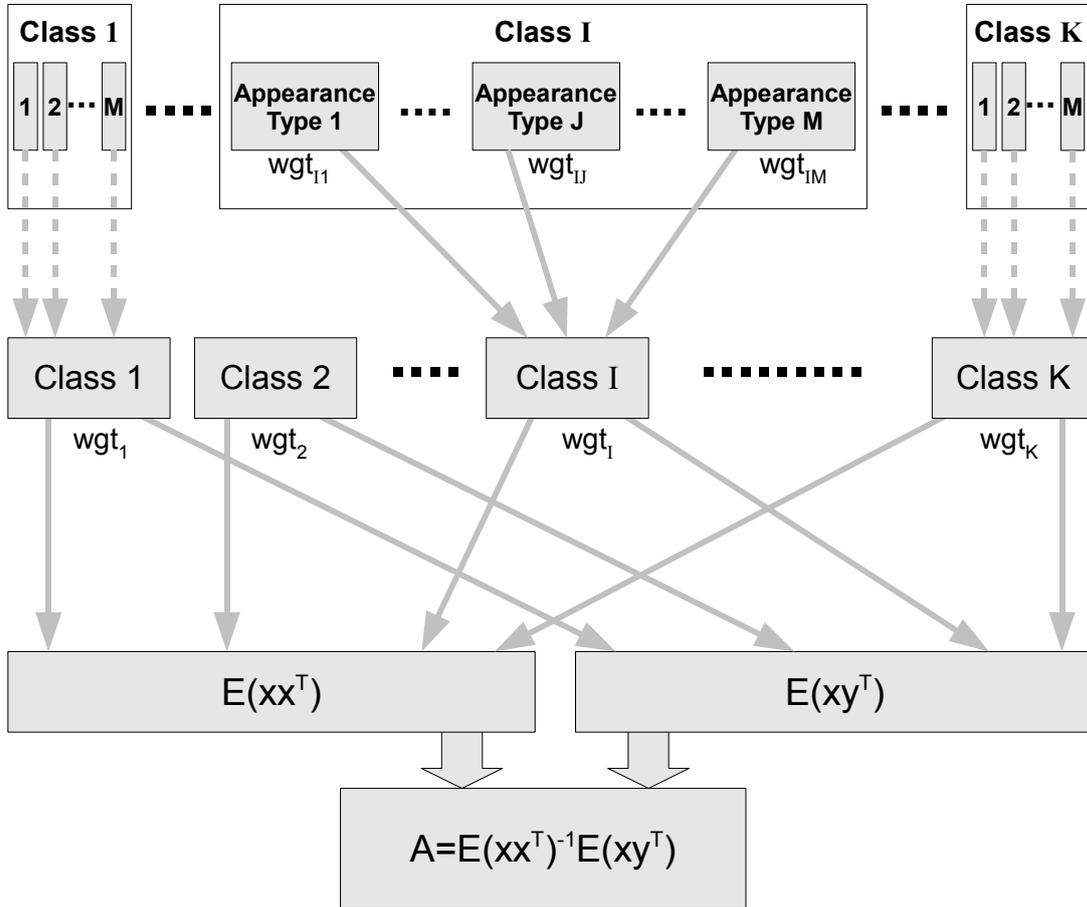


Figure 4.6: Scheme for the training of a polynomial classifier from separately stored moment matrices for K different classes and M different appearance types. In the chart the split in different appearance types is highlighted for class I. In the system the other classes are treated analogue. For the creation of new classifiers only the moment matrices have to be stored, not the complete samples sets raster images. The weights wgt_{xy} are set independent of the a priori probability for the respective class or appearance type.

4.3.1 Bootstrap Training and Classifier Refinement

This section describes how to fine tune a classifier derived by the algorithm explained in section 4.3. Once there is a stable classifier the next task is to optimize the classification results of this classifier by readaption to achieve the optimal result for the complete traffic sign recognition system. This does not necessarily

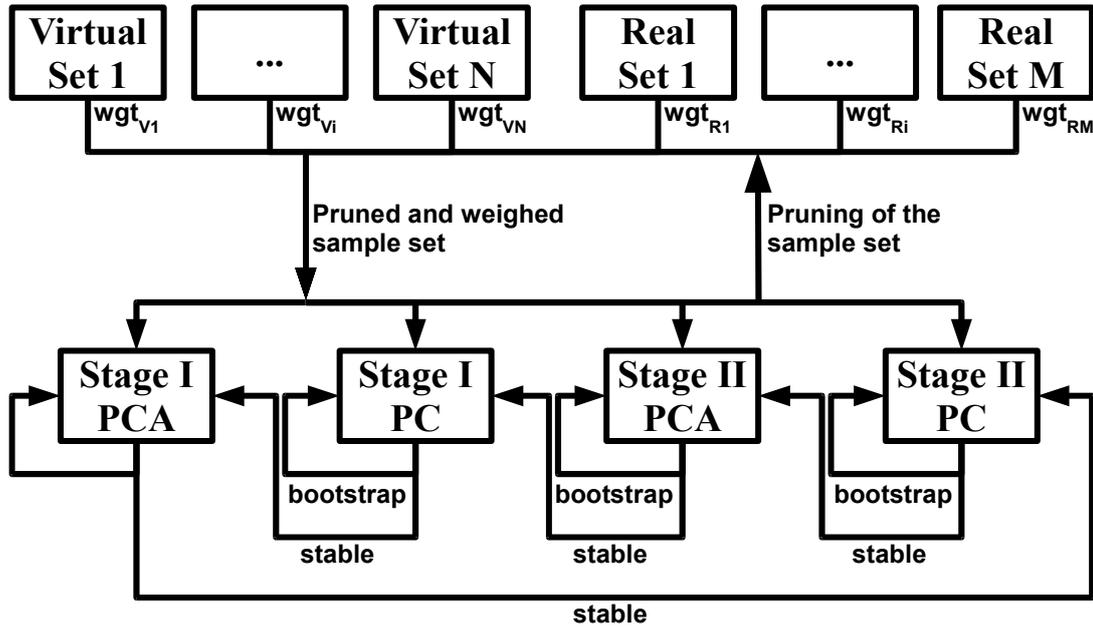


Figure 4.7: The developed system for the retrieval of large sample sets, which have to characterize in detail the problem of the recognition system. Scheme for the selection process in the bootstrapping of the classifier system. Sets of *synthetic* signs and image *cutouts* of real recordings are used in a weighted mix to adapt the classifier system which in term selects the samples to be used for the adaptation in the next iteration step.

mean to get the maximum number of samples classified correctly, since the a priori probability of the different classes and appearances for classes vary strongly and thus the more rare classes or appearances of signs might be disregarded in favour of the more frequently encountered signs or appearances. Two approaches are most often used in the literature. The first approach uses equal weights for each sample. This represents the a priori distribution of the classes in the training set. The second uses equal weights for the accumulated samples for one class or appearance class. The a priori weighting leads to the previously mentioned loss of objects of seldom encountered classes, while the equal weighting for the samples belonging to one class might lead to an over-emphasising of the rarely encountered sign classes and thus to too-high losses at the side of the most frequently encountered classes. Training the classifiers with a designed weighting of the different appearance classes can be used to optimize the classification result for the complete traffic sign recognition system. An advantage of the polynomial classifier is that for the weighting of the training set not the individual samples, but only the respective moment matrices and mean vectors have to be reweighted and used in equation 4.14 to adapt a new classifier.

The fact that a real world sign object has more than one image representation in a track of circular objects leads to a certain redundancy, which can be used to optimize the system. For a track to be accepted, at least three images holding circular image objects have to be encountered, see section 2.5, while the vast majority of tracks has many more entries. Thus the rejection of a few samples in a real world objects track does not necessarily mean that the traffic sign recognition system is not able to classify the object correctly. This redundancy can be used by altering the optimization goal of the classifier from classifying each sample correctly to classifying the majority of tracks correctly. The approach chosen for this optimization is the reweighting of the moment matrices for the different appearance types and classes. This additional set of parameters is optimized via a gradient descent on the track-wise classification outputs. The limiting values for the weights are the a priori values and the equal weights mentioned above.

Another function of the bootstrapping step is the detection of sample types not being represented in the *synthetic* sample set. Should a sign's appearance type be encountered in a recorded sequence that is not yet included in the collection of appearance types this type can be added at this point of the algorithm and the necessary samples be generated. The new type is detected by looking at labelled tracks of objects where all elements in the track either have a very low reconstruction rate in the principal axis transform of stage II of the classifier or are classified with a very low output value for the correct class in this stage.

Due to the imperfect training set this part of the algorithm acts as an additional selector for samples to be used in the training process. The three inputs used for this are the reconstruction error and the classifiers discriminant vector \mathbf{d} . From \mathbf{d} the value for the correct class has to be above a threshold and the rejection radius \mathbf{r} has to be below a threshold. The complete description of this part is in section 4.3.2.

With the existing classifier newly recorded image sequences can be processed to extract samples pre labelled by the existing classifier. Then either the scheme from section 4.2.3 or a human in the loop relabelling the traffic sign objects tracks is used to correct eventual wrong labels. In addition to adding samples that were classified correctly already the effect is both adding the *cutouts* of recognized tracks and, if a human labeller was used, adding the samples complete of wrongly classified tracks to the correct set.

4.3.2 Selection of Samples in the Bootstrapping Process

A necessary step in the classifier construction is the selection of suitable samples from the imperfectly labelled set. The adaptation of a classifier is based on a set of *synthetic* signs which are correctly labelled, even if some of the samples might be transformed in an unnatural looking way due to an unfortunate random selection of transformation parameters. For the following classifiers the sample set includes label errors since the human labeller or the automatic labelling process might have

made mistakes, or there might be errors in the track of images used to extract the *cutouts* from, such as the first or last objects in the track not belonging to the real world object labelled. Additional care has to be taken for the *cutouts* that are labelled correctly, but are illegible due to bad image quality, occlusions, an inaccurate circle location by the detector or unfavourable geometric conditions like high rotation of the object. To keep these samples from diluting the classifier they are removed by the sample selection algorithms mentioned above, RAD criterion, classification value and reconstruction rate of the principal component analysis, see section 4.3.1.

The process of pruning the sample set of unsuitable training samples is implemented in a bootstrapping loop. There are two possible approaches. First, starting with a small perfectly labelled set and adding more samples afterwards or second, starting with the complete imperfectly labelled set and removing samples from the set.

In the first approach we start with the created set of *synthetic* samples. The first classifier provides an estimate on how perfect signs look like after a set of geometric and luminance transformations. Thus the selection on atypical signs will be rather strong, removing most of these from the sample set. Training a new classifier including the real world samples accepted by the selection process allows to run a second selection process on the complete set of samples, this time adding samples that were not added in the first run, since the real world signs included some transformations not covered by the generation process of the *synthetic* signs. This procedure can be repeated until the changes in the samples sets after the selection are small enough.

The second approach to prune the sample set is to start with the complete imperfectly labelled samples set, train a classifier based on this set and remove the samples that do not satisfy the selection conditions, then retrain the classifier with the pruned set until the changes in the samples sets after the selection are small enough. This approach was used in this thesis, since the generalization capabilities of the classifiers created using this approach were superior compared to the classifier created using the first approach.

The sets added and pruned in the bootstrapping process lead to new adapted classifiers in each step. To allow for computational efficiency in the bootstrapping process the re-adapting and pruning is first executed on the stage II polynomial classifier and only when it is stable there the principal component analysis for this stage is re-adapted. When both parts of stage II for all sub classifiers are stable, stage I is re-adapted. This process is repeated until a satisfactory solution is reached or the process ends due to no or too few changes in the pruning process and thus the classifiers not changing any more. The complete scheme is depicted in figure 4.7.

4.4 Combination of Classifiers of Different Countries

The idea behind this approach is that for just a few appearance types a classifier for each of the types could be adapted and being used in the country this appearance type is used. For a multitude of countries this would lead to a huge number of classifiers and the complexity of gathering samples for training sets. Thus a way has to be found to reduce the number of classifiers necessary to a manageable number and adding a new classifier only, if in a newly added country, where the system should be functionable, the signs are too different to be correctly classified with one of the existing sets of classifiers. Otherwise one of an already existing set of classifiers should be used in this country. Which classifier should be used can be determined by testing the performance of the existing classifiers on a small set of real samples or on *synthetic* samples. Should a sample set of sufficient size be recorded in this country these samples can be used for readaptation of the respective classifier set by adding it to the sets of the already included countries. An implementation of this algorithm was published in the diploma thesis of Denis Koch [Koch, 2007], which was supervised in the course of this thesis.

When the vehicle enters a country, which is detected by using GPS and a map, the corresponding set of classifiers is loaded and used. A set of classifiers consists of the seven classifiers in stage I and stage II as explained in 2.13. The composition may differ from country to country with varying choices for inner/outer limit or end of limit classifiers.

5 Design of the Evaluation System

The system can be evaluated module by module and — of course — by function, but there are more intermediate steps necessary for system optimization and evaluation. A simple example is the sole evaluation and parameter optimization of the detector to allow the detection of signs being very far away and thus being very small in the image or being partly occluded. Optimizing the system to reliably detect even these small or occluded objects leads to an increasing rate of false positive detections, since the thresholds in the detector would have to be lowered very far to safely detect these candidates. In most cases the vehicle approaches the far away signs up to a distance of less than 20 *m* in our system, thus the small objects are just the begin of a track of objects becoming larger. Those not getting large enough in the image usually are the signs on the more distant roadside of a multi lane road. For the system the detection and recognition of the one at the closer roadside suffices, thus the non-observance of the far off sign does not impair the system, in the contrary, the recognition of signs having a very large lateral offset relative to the vehicle might complicate the decision for the currently valid speed limit, when signs on parallel lanes or on exit lanes are recognized.

5.1 Evaluation of the Camera Control

For the complete system the camera control has to be evaluated as well. Since camera control is not in the scope of this thesis there are only a few hints on as how the camera settings are evaluated.

The complete inclusion of the camera control into the evaluation concept would lead to a huge expansion of the sets to be assessed, since there would have to be a large number of frames for each camera setting to be evaluated. Due to this fact the camera control is optimized with small sets of images for each camera setting, examining the edges obtained from the images and the over all appearance of the images. Since the image sensor allows settings for high dynamic range response curve there are many parameters to be considered. The main control target is set to a rectangle on the road, optimizing the mean value and the number of extreme pixel values in the rectangle. There are additional heuristics for tunnel entrances and exits and special settings for images under low lighting conditions such as at night or in tunnels.

5.2 Creation of the Training and Testing Set

For the following evaluations a database of labelled image sequences has to be created. For the different stages of evaluation the type of labelling has to be different. The different types of labelling stages vary greatly in effort to be expended in the labelling process and thus in the size of the set for a given time spent for the labelling. The following four types of labelling were performed.

1. Exact labelling of the circles belonging to traffic signs in the image. (Set1)
2. Labelling the circles detected by the traffic sign detector. (Set2)
3. Labelling the whole tracks of circles as generated by the system by assigning the class of the last element in the track to all elements in the track. (Set3)
4. Labelling in the real world domain, assigning the permitted maximum allowed speed to sections of the road in recorded image sequences. (Set4)

The workload for the labelling process using the first scheme produces an extremely workload for the labelling person. Thus only comparatively few images were labelled in this way. These images were used in the process of adapting the detector. Some examples are depicted in figure 5.1.

After the detector was developed and implemented the labelling was conducted in the second way described above. Since the system detects nearly all signs at least in some images of the image sequence belonging to one real world traffic sign object, as shown in section 6.3, new circle objects have to be added using the first labelling scheme only seldomly. The drawback accepted in this faster labelling method is the possibly imperfect placement of the circles in the images as seen in figure 5.2.

The huge number of labelled images necessary for the use of the evaluation in multiple countries lead to the development and implementation of the third labelling scheme. Here the tracks of circles in the sequences are the base for the labelling. The tracks are used as they are created by the tracker. This means that elements assigned wrongly to a track of circles will still be labelled with the same class as the rest of the track, since reviewing every single item of the track would cause much higher workload for the labeller. Here the trade off was made allowing some erroneous labels for gaining a much higher variety and number of labelled sign cutouts. A typical case for errors in the assignment of circles to a track is a circle object found in cluttered background which is connected to a newly detected traffic sign object, who's first image circle object is detected in the next frame. The number of circle elements given a wrong label in this manner is much lower than the number of correct labels, but has to be observed when training or evaluating the classifiers based on these training sets, meaning that the type of classifier chosen has to be robust against a small number of wrongly labelled training samples, such as the polynomial classifier has proven to



Figure 5.1: Examples of labelled circles

be. This scheme of labelling produces an extremely low workload for the labeller, especially when the whole traffic sign recognition results are given as a first choice for the labelling. Thus, as the whole traffic sign recognition system gets more refined and the number of errors decrease, the number of labels to be changed by the labeller decreases as well. In figure 5.3 the rectangular regions of the image enclosing the detected circles, here called *cutouts* are depicted. Every second row is comprised of *cutouts* which are generated when two circles are detected for one image sign object. When the real world sign is still far away, on the left side of the tracks, the image objects are barely decipherable, but getting more recognizable when closing up to the object as can be seen in the right part of the tracks. The lowest two rows show a track where objects from the back of a lorry sometimes hide the sign object and thus introduce wrong labels in the sample set, when the last object in the track defines the label of the whole track. Since this kind of occlusion is extremely rare it does not influence the following analyses. In the evaluation process the set is pruned by removing wrong labels, but the problem to which extent signs with occlusions and *cutouts* imperfectly centred on the signs circle have to be recognized remains. This part of the pruning of the evaluation set is based subjectively on the appraisal of the labeller, removing too



Figure 5.2: Examples of imprecisely detected circles

occluded or acentric samples from the evaluation set.

For the complete evaluation of the detection system the traffic scene has to be labelled, thus allowing to benchmark the traffic sign recognition system against the *ground truth* of the traffic regulations. The database created in this step takes all traffic regulations of the country where the respective sequence was recorded into account and thus allows the evaluation of a complete traffic sign recognition system on the base of distance travelled with the system showing the correct, the false result or none.

For German roads more than 50.000 kilometres of driven distance were labelled for the valid maximum velocity. The sets for other European countries are considerable smaller, 10.000 kilometres in France and Italy for example. The exact numbers are given in section A.9. In all there are about 109.000 kilometres of labelled trips, resulting in 11.5 gigabytes of data. The set consists of 25,9 million images showing 124.000 real world signs and over 13.000 supplementary signs.

In figure 5.4 a labelled route of 200 km is depicted. The vehicle information yaw rate and speed are shown in dark blue and light green. The *ground truth* label is the dark blue line, where a label of 200 km/h stands for *no speed limit*. The

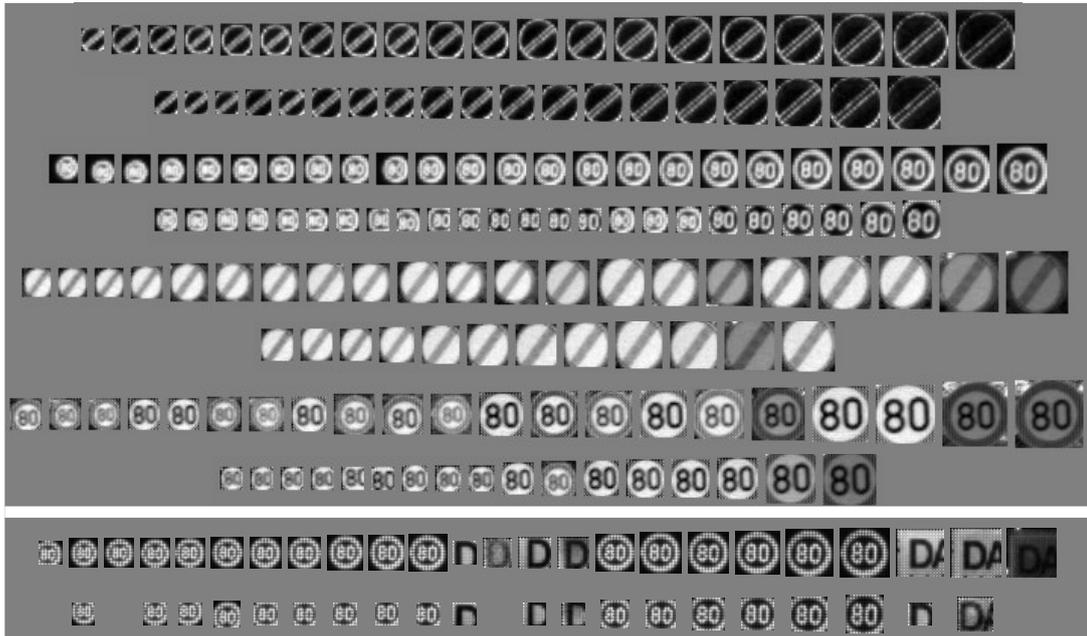


Figure 5.3: Examples of rectangular image regions forming tracks with each second row showing rectangles achieved by detecting two circles for one image object. In the two bottom rows a track including wrong circles is shown.

lines below the zero line show additional information like temporary or weather dependant speed limitations. The red dots along the *ground truth* line depict speed limit traffic signs. The red line above the zero line and the red bar below the zero line show discrepancies between the results of the speed limit recognition system and the *ground truth*. In the first 17 kilometres the error shown by the red line are generated by the fact that there was no previous traffic sign, so for the recognition system the real speed limit was unknown. In this route there are 23 kilometres of 198 kilometres shown to be incorrect. Disregarding this part of the route the recognition system produces the correct result when compared with the *ground truth* on 175 of 181 kilometres of the trip.

All the sets above were labelled by a human operator, but still there were errors in the labelled sets, which was to be expected when considering the huge number of signs labelled. These errors were corrected every time they were detected. Often the traffic sign recognition system itself did not make the same mistakes as the labellers and thus when reviewing the mistakes the recognition system made the label errors could be corrected.



Figure 5.4: Example of a labelled route of about 200 km length with labelled auxiliary information yaw rate (top blue), travelled speed (light green), allowed speed or *ground truth* (dark green), and error of speed limit system (red)

5.3 Evaluating the Detector

The first and foremost of the output of the detector to be reviewed is the detection rate and if the positioning of the circles is accurate enough. A second value being important in this context is the number of other objects being detected along with the signs. This second value should be kept as low as possible. From these two we can define a signal to noise ratio of the given detector, meaning the detection rate relative to the whole number of circles detected. These two values can be plotted in a two dimensional curve when varying a parameter. Another parameter that has to be considered is the computational load created by the detector, which is influenced by the parameter settings. Important parameters in the detector are:

- The edge threshold of the coarse detector
- The number of peaks used from accumulator

Important parameters in the detector verification are:

- The type of edge detector used in verification step
- The edge threshold
- The tolerance angle for checking if an edge pixel fits to the circle edge template

- The type of algorithm used on the edge image: the correlation detector or of the chamfer algorithm
- Spatial low pass on the template matching results on/off
- The decision threshold for the corresponding algorithm used
- final acceptance threshold

The optimization of these parameters is explained more closely in section 6.3.4.

5.4 Tracker

Since the tracker is mostly used as a blackboard and to connect the detected objects from consecutive images, the evaluation for this part of the algorithm is based on its capability of connecting the correct objects while connecting as few as possible of the objects not being signs. The connection of the correct objects is of vital importance for labelling the signs, since for the time and effort to be invested for labelling the objects it is not possible to label all single objects in the sample set. Instead the label is given to the whole track, expecting all elements of the track to belong to the same class. The labelled set for optimizing the tracker consists of 664 real world signs and thus 664 tracks consisting of 13487 image circles in 91146 frames and additionally 45740 verified circles not belonging to a traffic sign.

5.5 Single Sign Classification

In this section the evaluation of the single pattern classifier is described. The most impact on this evaluation has the selection of the test set. A set of ideally segmented traffic sign images could be used for evaluation, but this would not help in analysing the complete traffic sign recognition system, since the traffic sign pictures classified would look different from the ones detected by the system, since the variations introduced by the imperfection of the detector would not be included in the set. Thus the test set has to be created by running the traffic sign detector on a set of input images and the regions detected by the traffic sign detector have to be classified. Since the large set described in chapter 5.2 includes a comparatively high number of wrongly labelled signs due to tracking errors, misplaced detections or label errors, a selection process has to be performed on the set to reduce the number of wrongly labelled objects to allow a correct evaluation. To perform this task with a minimum of human interaction required three different schemes were implemented.

- A principal axis transform was performed on a small set of correctly labelled sign patterns. The complete set of patterns was transformed with the first

principal axes, leading to a reconstruction error \overline{R}_M^2 of about 0.1 or 10% on the small set used to train the principle axis transform. This ratio is determined via equation 4.8. After the reconstruction, meaning the inverse transform, the patterns were sorted by the error between the correlation of the reconstructed pattern and the original pattern R_M , computed as in equation 4.9. The patterns yielding the highest error being most likely candidates of having been incorrectly labelled.

$$\text{Reject sample if } R_M^2 \geq \theta_1 \overline{R}_M^2 \quad (5.1)$$

- A polynomial classifier was trained on the principal axis transformed set of correctly labelled sign patterns. The complete set was classified with this preliminary classifier. The samples were then sorted by the value of the output value of the decision vector \mathbf{d} of the class they were labelled to belong to. The samples having the lowest output neuron being the ones to be most likely wrongly labelled.

$$\text{Reject sample if } d_k \leq \theta_2 \quad (5.2)$$

- Based on the classifier trained for the second scheme, the Euclidean distance of the decision vector \mathbf{d} of each sample to the ideal decision vector \mathbf{y} , meaning one for the labelled class and zero for all others, is computed. This value is called *reject radius* or RAD criterion, as explained in equation 4.17. The samples are sorted by this value, the higher this value is, the more likely its label is incorrect.

$$\text{Reject sample if } \text{RAD}_k \geq \theta_3 \quad (5.3)$$

The three algorithms described above allowed the removal of most, if not all of the wrongly labelled patterns from the evaluation set, thus allowing to measure the recognition rates of the different classifiers.

The values examined for the single pattern classifier are the values in the decision vector \mathbf{d} as defined in equation 4.12. For each sample belonging to a traffic sign class there are five possible results. For each sample in the evaluation set the different costs are added for each type of result.

1. The sample's decision vector \mathbf{d} is accepted with respect to the RAD criterion (4.17) and the highest value in the decision vector belongs to the *correct* class of the sample.

$$\text{RAD}_k < \theta_3 \quad \bigcap \quad \text{maxidx}(\mathbf{d}) = k \quad (5.4)$$

2. The sample's decision vector \mathbf{d} is accepted with respect to the RAD criterion

(4.17) and the highest value in the decision vector belongs to a *different* class than the one the sample was labelled as, thus the decision would lead to a misclassification.

$$\text{RAD}_k < \theta_3 \quad \bigcap \quad \text{maxidx}(\mathbf{d}) \neq k \quad (5.5)$$

3. The sample's decision vector \mathbf{d} is accepted with respect to the RAD criterion (4.17) and the highest value in the decision vector belongs to the *garbage* class, thus the decision would lead to a classification as *garbage*.

$$\text{RAD}_k < \theta_3 \quad \bigcap \quad \text{maxidx}(\mathbf{d}) = \text{Idx}_{\text{garbage}} \quad (5.6)$$

4. The sample's decision vector is rejected with respect to the RAD criterion (4.17).

$$\text{RAD}_k \geq \theta_3 \quad (5.7)$$

5. A sample not belonging to a traffic sign is wrongly classified as a sign and is not rejected by the RAD criterion. This is called a *false positive*.

$$\text{RAD}_k < \theta_3 \quad \bigcap \quad \text{maxidx}(\mathbf{d}) \neq \text{Idx}_{\text{garbage}} \quad (5.8)$$

The correct classification as in type one is set to create no costs. The cost for the confusion of two traffic sign classes as in type two of the results are set to be the same, since no confusion between two classes is less unfavourable than the others. The errors of type three are thought to be of less impact and thus the costs are lower than those of type two. The rejection due to the RAD criterion is assigned an even lower cost since this type of error is the easiest to be countered in the track classification stage.

5.6 Track Classification

In this part the result based on the correct classification of one real world traffic sign object has to be evaluated. In the previous chapter the evaluation of a single image object was examined. If these were all correctly classified the track classification stage would be superfluous. For the combination of the results of the single pattern classification the reliability of the information gained by the single pattern classification has to be considered, for example using the information about the decision vector \mathbf{d} for the single image objects and the resolution of the raster images before the normalization process.

There are four different types of results:

- The number of *correctly* classified sign tracks.

- The number of *misclassifications* where two sign classes are confused.
- The number of tracks not belonging to traffic signs, but being classified as such, the *false positives*.
- The number of *false negatives*, where the track of a sign is classified as garbage or rejected.

5.7 Traffic Scene Interpretation

This part of the evaluation is based on the results of the track classification for real world signs and is moving one step further. In many cases, when there is only one real world sign present in a section of the road, the classification of the real world object as a track of image objects is the final result of the traffic sign recognition system. On most higher order roads like highways traffic signs are placed on both sides on the road or multiple signs are placed above the road on overhead gantry signs, the recognition of one of the signs is sufficient. The traffic scene is recognized correctly when one of multiple signs with the same meaning placed in one segment of the road is recognized. Thus rejecting, but not misclassifying, the additional signs does not lead to an incorrect evaluation of the traffic scene. This means that rejecting signs with a large lateral offset might actually improve the over all system performance, even when reducing the track classification performance. Another case is the rejection of signs on parallel roads or exit lanes, which again reduces the performance when evaluated via the track classification, while improving the system performance.

The evaluation of the complete system is done on a per-kilometre basis. The *ground truth* is labelled along the complete routes travelled and the allowed speed computed by the image recognition system is compared to this value. If the resulting speed allowances differ, the distance travelled while this difference exists are accumulated. The number of kilometres the system differs from the *ground truth* in relation to the complete distance travelled is the most important value evaluated.

Additional values introduced are the expiration length for the validity traffic signs, after which the restriction introduced by a traffic sign is supposed to end even without a new traffic sign overriding the previous. This is necessary since traffic signs in construction sites have no *end of limit* signs placed at the end of the construction site. Another reason for the necessity of the elapsing length is that the traffic sign recognition system has no internal information if the vehicle turned on to another road than the one the restriction was placed on. Thus the validity of a sign has to be limited. When a navigation system is used the information of the map can be used to end the area of validity of the sign.

The rules introduced for deciding for the allowed maximum speed in a *scene* are as follows:

- *Active* signs are deemed more important than *passive* signs.
- Limiting signs with supplementary signs restricting the validity to not applicable vehicle classes, e.g. lorries are ignored.
- Limiting signs with the *arrow* supplementary sign for vehicles leaving the current road are ignored, since without the assistance of a navigation system the route of the vehicle is unknown.

The following three rules have been introduced due to the imperfections of the supplementary sign recognition system.

- If there are more than one sign of the same type in the same scene and underneath one of them a supplementary sign has been recognized, the other signs are set to having the same supplementary sign, weather or not it was detected underneath those.
- In case more than one sign of similar importance is recognized in the same scene, e.g. two active limits or two passive limits, the one showing the higher value is deemed correct and the inaccuracy due to an oversight of a supplementary sign underneath the limit with the smaller value. This approach is correct in 70% of the cases in question.

6 Experimental Investigations and Evaluation

In this chapter the implementation details and results of the evaluation of the algorithms presented in the chapters 2 and 4 are presented. First the general considerations on the camera setup and settings are elaborated. In the following sections the main parts of the algorithm are inspected. Those are the detector, the tracker and the classification process. The results of the necessary extension modules for the detection and classification of supplementary signs are presented in the following section. The final section in this chapter is the over all evaluation of the traffic sign recognition system.

6.1 Examining Camera Exposure Time

As mentioned in section 2.3 the maximum exposure time exp_t is restricted by certain limits. These limits are influenced by the following factors:

- The size of the sign $s[m]$
From the rules for manufacturing and setting up traffic signs the diameters of the sign are known for each country and can be stored in the systems data base. For the use in countries where the permitted sign sizes are unknown default values can be used until the system calculates the sizes itself.
- The lateral offset of the road sign to the sensor when passing the sign $y_s[m]$
The horizontal offset where the system should work with undiminished accuracy has to include cases where the sign is set beside the neighbouring lane. The equations for vertical sign offsets are the same as for horizontal offsets. Thus only the horizontal offset is inspected in the following.
- The sensor resolution $r[\text{pixel}/\text{degree}]$ and field of view $[\text{degree}]$, e.g. focal length, or image width $w[\text{pixel}]$.
- The minimal size of a sign in an image to be recognized $s_p[\text{pixel}]$
From tests with the classifier we know that for the best recognition performance the sign should be seen at least three times at a minimum size of 20 pixels or larger in the image.
- The minimum number of images in which the sign has to appear to be accepted N
In this case the number of tracked elements N should be 3 or more.

- The number of frames per second $fps[Hz]$
Depending on the computational speed of the system on the respective hardware this can be a crucial factor.
- The maximum allowed movement of the signs representation in the image during exposure time to be still recognizable $b[\%]$
For a sign to remain recognizable the imposed motion blur cannot be allowed to exceed certain limits. The inner part of a *Speed Limit 100*, for instance, has ten black to white transitions, as shown in figure 6.1. This means if the sign moves about ten percent of it depicted size in the image during one exposure the pattern shown in the image will become an indecipherable grey bar.
- The vehicle speed $v[m/s]$
For most European countries 130 km/h or lower is the maximum speed, but considering Germany higher speeds have to be checked as well.

Since varying all these parameters would lead to confusing results some will be fixed and only a few are varied in the following inspection. This yields the worst case results in this field. These factors combined should give us the maximum vehicle velocity at which the system works unimpaired. The resulting two bounds are the number of frames in which the signs representations are large enough and the distance from which the motion blur becomes too large.

The variables are described in the list of influencing factors above and the figure 6.1.

$$d = \frac{y_s}{\tan\left(\frac{w}{2r}\right)} \quad , \quad d_N = d - \frac{Nv}{fps} \quad (6.1)$$

The following equation describes the relation between the influencing factors necessary for the signs size in the N-th image to be at least the minimum size necessary for a reliably correct classification s_p .

$$s_p \leq \alpha_s \cdot r = r \left(\arctan\left(\frac{y_s + s}{d_N}\right) - \arctan\left(\frac{d_s}{d_N}\right) \right) \quad (6.2)$$

The following equation describes the relation between the influencing factors necessary for the signs to have a motion blur smaller than the maximum allowable for a reliable classification b .

$$b \geq \frac{(\alpha_e - \alpha) r}{s_p} = \frac{r}{s_p} \left(\arctan\left(\frac{d_s}{d - exp_t \cdot v}\right) - \arctan\left(\frac{d_s}{d}\right) \right) \quad (6.3)$$

The above equation allows the computation of the camera setup according to the factors given by maximum travelling speed and geometric factors.

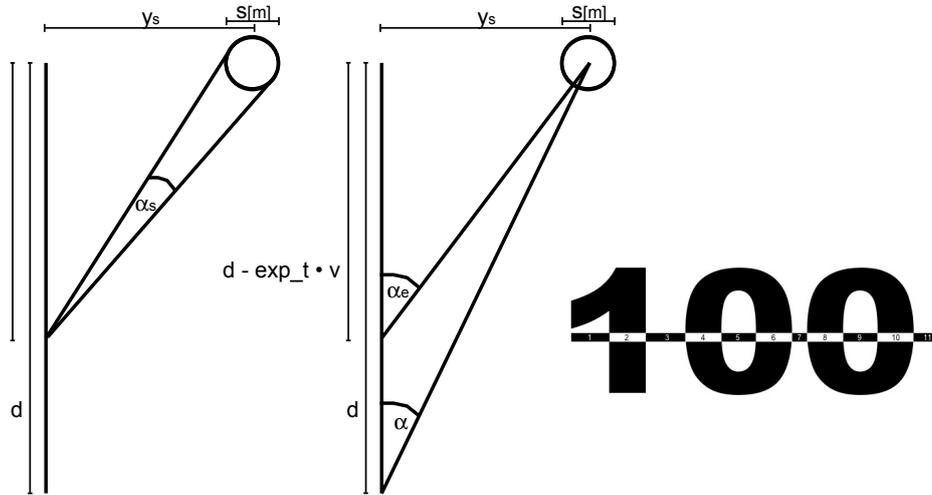


Figure 6.1: Scheme of the two limiting factors for velocity and exposure time. Firstly the minimum necessary sizes, secondly the maximum blur. The third element shows the black and white transitions on a speed limit 100. See section 6.1

6.2 Training and Testing Sets

As described in section 5.2 there are four main types of sets used as *ground truth* in the evaluations. In the set *labelled using the first scheme*, the exact circle labelling, there were 3559 image circles belonging to traffic signs in our experiments. Of each real world traffic sign in the set only one in each track of image circles was labelled to get a more diverse set of circles. The sizes of the circles varied from the minimum width of objects used in the detector, here 15 pixels, to the maximum size of 61 pixels for the evaluated system.

	Day	Night
Two Circled Signs / <i>Speed Limit</i> signs	951	621
Singly Circled Signs / <i>End of Limit</i> signs	222	193
Number of Circles	2124	1435

Table 6.1: Sample set for determining the circle verification algorithm.

The set was split in night and day images to allow different detector settings for the substantially different environment encountered at night as compared to daytime images. This includes low light characteristics of the sensor like higher image noise and artificial lighting of the scene through headlights and street lights.

The set *labelled using the second scheme*, labelling single circles as detected by the system, was created for the first adaptation of the tracking and classification

algorithm and was not used in the following evaluations since the size of the set *labelled using the third scheme* soon grew much larger and thus more significant than this one. This second labelled set consisted of about 5000 labelled traffic sign circles.

For the evaluation for the tracker the set was taken from German sequences. The sequences were picked from day and night and many different weather conditions. The size of this set is given in table 6.2.

Frame number	Real world traffic signs	Frames showing traffic signs	Circles of traffic signs	Circles not of traffic signs
91146	664	11874	3784	46146

Table 6.2: Sample set for adjusting and evaluating the tracker.

The third set, based on track labelling, has to be larger by far than the previous two sets to allow statements about the classifiers performance in different countries and using different classifier training schemes. Over one million image circles holding signs were labelled and about one million garbage objects were added. The garbage objects were chosen from objects forming tracks of circles in the image sequences only, since only these would reach the classification stage in the real system and thus are realistic inputs for the classifier. The exact composition, separated country-wise, is given in the appendix section A.8. In the following tables first the numbers for the German training and testing set are given (6.3, 6.4), then the numbers for the European sets in table 6.5.

	Active		Passive		Garbage
	End of Limit	Limit	End of Limit	Limit	
Train	4671	16119	6622	30825	345639
Test	16125	120613	37730	383572	968515

Table 6.3: Sample set for German traffic sign *cutouts* in the stage I classifier, see section 2.6.2.

For the detection and classification of supplementary signs the set used for the training and testing of European signs was filtered for signs showing supplementary signs. Those were used in the adaptation of the supplementary sign modules. The numbers are given in table 6.6.

The sample sets shown above are used to adapt and test all parts of the traffic sign detection and recognition algorithm.

	Active			Passive		
	End of Limit	Limit Inner	Limit Outer	End of Limit	Limit Inner	Limit Outer
Train Garbage	5138	20857	17575	12131	67272	59476
Train/Synthetic	1000 per class					
Train Traffic Signs	4062	10936	13852	11588	60394	66896
Test Garbage	11382	30644	10793	16567	56785	23939
Test Traffic Signs	14397	58442	49248	33991	188502	166657

Table 6.4: Sample set for German traffic sign *cutouts* in the stage II classifier, see section 2.6.2. The differentiation between active and passive signs divides signs with bright digits on dark ground from signs with dark digits on bright ground. Limit inner and outer differentiates between the outer *cutouts* created from circles including the red rim, while inner *cutouts* derive from detected inner circles of the red rim.

Country	AT	BE	CH	DE	DK	ES	FR	IT	LU	NL
Train	2462	763	1184	26878	428	4330	2679	4348	249	1601
Test	6999	2150	3338	76691	1198	12267	7618	12434	711	4538

Table 6.5: Sample set for the European traffic sign *cutouts* classifier. The table shows the number of real world signs. The number of *cutouts* is about 20 times higher since this is the mean length of an image tack.

6.3 Traffic Sign Detector

In this section the evaluation and parametrization of the main parts of the traffic sign detector are introduced, beginning with the edge detector and ending with the verification of the detection results.

The detection range in size and centre coordinates can be determined by looking at the rulebook [Bald and Giesa, 2002], but still have to be verified by experiments in case of the setup of the signs not being according to the rules. Figure 6.2 shows that the bulk of the signs have a diameter smaller than forty pixels. Since a higher radius allows easier classification and, most of the time, better three dimensional positioning, circles up to a diameter of sixty pixels are detected.

The region in which the signs are detected depends on the mounting of the camera. From the positions of the signs in the images the area to be scanned for circles has to be from ten degrees above the horizon to one and a half degrees below. To be able to detect temporary signs on tripods seldomly used by the police at accident sites, or more often in France, the detection area is expanded to five degrees below the horizon, leading to the image area from the top to the pixel row three hundred in the camera system used for the evaluation. Horizontally no

	Country	AT	BE	DE	FR	ES	LU	PT
Train	Cutouts	3881	2390	14108	8671	6265	597	1896
	Synthetic	1000 per class						
	Signs	191	118	695	427	309	29	93
Test	Cutouts	14302	8807	51995	31956	23089	2202	6989
	Signs	708	436	2574	1582	1143	109	346

Table 6.6: Sample set for the European supplementary signs. The table shows the number of *cutouts* used for the adjustment and testing of the detector, verification step and classification and the number of real world signs composing these sets. In the training of the pictogram classifier additionally *synthetic* sign *cutouts* were used for training the classifiers, since for many classes a very low number of signs were recorded.

Diameter	17	19	21	23	25	27	29	31
Number	135329	175236	89604	102940	65091	58614	53888	35443
Diameter	33	35	37	39	41	43	45	47
Number	30808	23659	19878	19788	12769	7159	6138	6098
Diameter	49	51	53	55	57	59		
Number	35271	5225	3808	2935	2012	1786		

Table 6.7: Table showing the exact numbers of circles per diameter as displayed in figure 6.2

area can be ruled out as to be seen in figure 6.3

6.3.1 Edge Detector

The edge detectors examined more closely, as stated in section 2.4.1, were Sobel filters, Prewitt algorithm and Canny filter. The Canny filter, while delivering the clearest edges for the human observer, removed too much fine structures from small circles by applying the low pass filter before applying the Sobel mask, since the system is operating at the limit of its applicable resolution anyhow. Tests with edges generated using the Canny filter led to 15% less relevant traffic sign circles on a set of about 100 000 traffic sign circle candidates. Between Prewitt and Sobel filter as initial mask there has been only a minor difference. Because of the Sobel filter having the more specified positioning of the edges this mask was used further on. In the verification step the image is used in full resolution, thus the *Bayer-Pattern* on the sensor alters the direction of the edges encountered. To prevent this the Sobel filter masks are stretched to touch pixels having the same colour filter only, resulting in two sparse 5x5 filters.

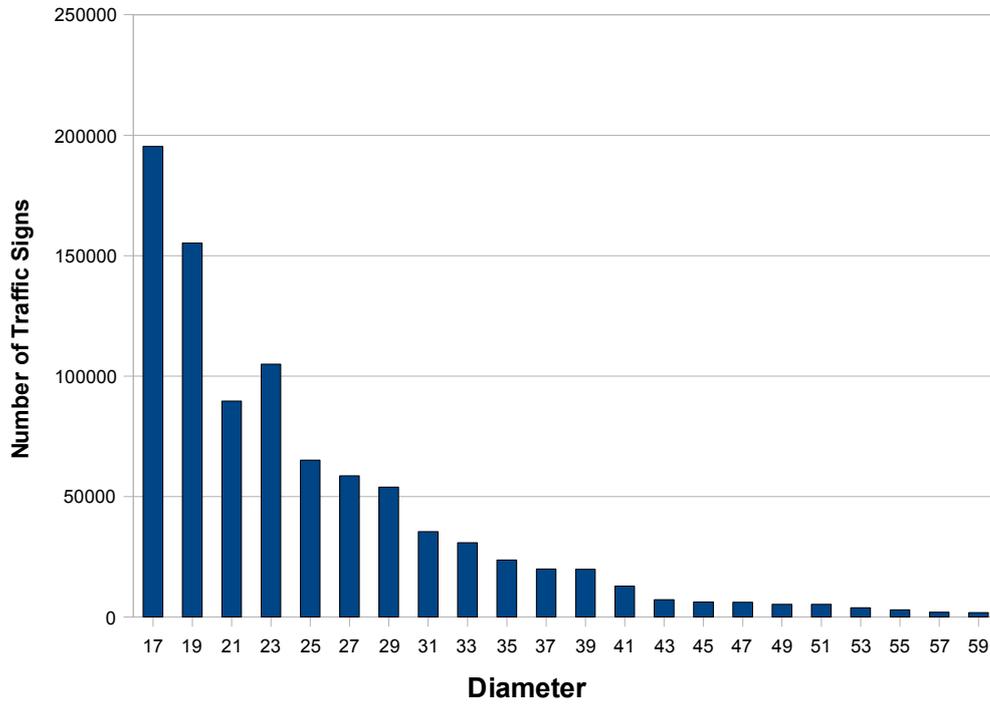


Figure 6.2: Distribution of the sizes of 905 thousand circles being the border of traffic signs in recorded images in the evaluation set

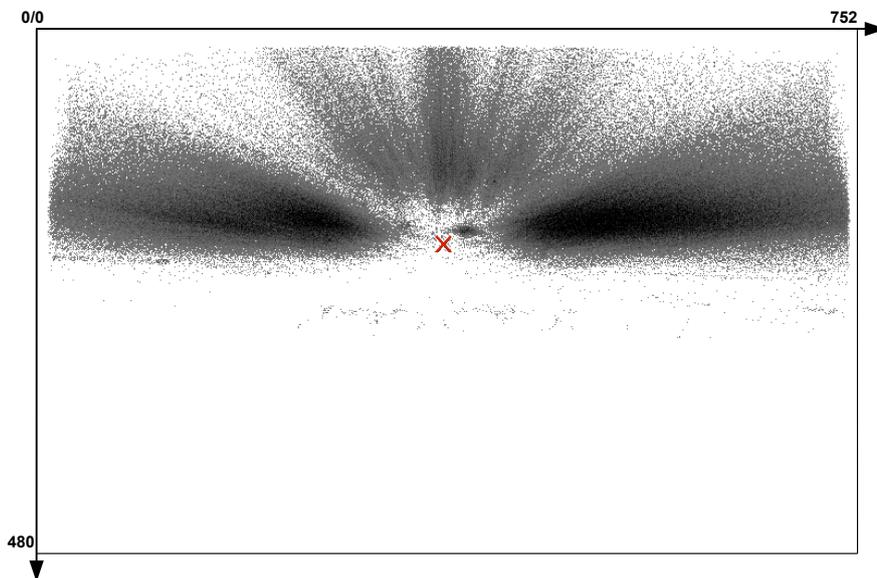


Figure 6.3: Centres of circles belonging to labelled traffic signs from about 905 thousand appearances in images. The darker the more centres were in this pixel area. The X marks the projection of the vanishing point in the image.

The non-maximum suppression scheme from the Canny algorithm was reduced to checking in two (horizontal and vertical) directions only for sake of speed with only minor adverse effects in edge quality.

The realization of the phase gradient detection has to allow for low cost in CPU time, cache friendliness and sufficient gradient accuracy. The 3x3 Sobel mask is processed line wise and the resulting X and Y Gradient processed directly, allowing to proceed without having to store 16 bit intermediate images.

The arc tangent is replaced by a pair of tables. In addition to the edge thinning as described in the Canny operator, there is a minimum threshold applied to the gradient amplitude to prevent edges generated by noise to enter the following evaluations. The threshold is comparatively low (about 3 least significant bits resulting in a threshold of 30 in the Sobel edge image) to allow for low contrast areas in the image to produce edges as well.

In addition to the absolute edge threshold there is a relative threshold comparing the centre grey value of the edge filter with the filter result. This reduces the number of edges in the bright areas of the image, edges in sky areas for instance. This additional step, though costing additional computation time in the edge detection process, reduces the number of edges considerably without removing relevant edge pixels. There is a minor impact on the mean computation time of the complete detection process, but the worst case calculation time was reduced by 40%.

To let this scheme to be acceptable in computation time the operations are realized by table lookups. There are three tables involved. Firstly the relative threshold table yielding the logarithm to the basis of two of the grey value of the pixel at the centre of the convolution mask. This value can be compared to the shift value derived by the use of the second table to get a fast relative edge threshold spaced in powers of two. The second table for determining the number of bits the gradient values have to be shifted down to fit into the third table, simultaneously thresholding the gradient by applying a high shift value on elements below the threshold. The third is a square table holding the arc tangent for the equation $\Theta = \arctan(g_u, g_v)$.

$$\begin{aligned}
 g_m(u, v) &= \max(|\delta_u(u, v)|, |\delta_v(u, v)|) \\
 v_c(u, v) &= \log_2(g(u, v)) \\
 v_g(u, v) &= \log_2(g_m(u, v)) \\
 s(u, v) &= \begin{cases} 16 & , \text{ if } g_m < \text{Edge Threshold } \epsilon \\ 16 & , \text{ if } g_m < v_c + \text{Relative Threshold } \epsilon_r \\ \max(0, v_g(u, v) - \log_2(\text{AtanTabSize}/2)) & , \text{ otherwise} \end{cases} \\
 \Theta(u, v) &= \arctan(dv \ll s, du \ll s) \\
 \ll s &: \text{ bitwise right shift, thus a division by } 2^s
 \end{aligned} \tag{6.4}$$

g_m : The maximum of the vertical δ_v and horizontal δ_h gradient value for one pixel, v_c : The \log_2 of the grey value of the centre pixel, s : the bit shift value, Θ : the resulting phase gradient

The phase values are mapped to the values of 1 to 255 in the resulting image. A value of zero indicates that there is no gradient at this position, see figure 6.4.

6.3.2 Hough Detector

The fast generalized Hough transform is based on a gradient image in which one or more objects have to be detected. The search is performed by accumulating *votes* in the Hough space, which's dimensions are the quantized targeted variables. In our case these variables are the centre point of the circle and its radius. The detection results are the peaks in the Hough space. For a complete explanation of the Hough algorithm please refer to [Duda and Hart, 1972, Illingworth and Kittler, 1987].

In the scheme of the fast *Hough transform* for circles to reduce memory consumption and computation time the radius range to be scanned is quantized into N discrete sets of radius ranges. The best trade off for N in the current case is four. The image dimensions u and v are quantized in single pixel steps on the sub-sampled image. The accumulator for the N radius-bands is stored interleaved to allow for efficient memory management.

The *Hough transformation* algorithm implemented in this thesis closely resembles the radial symmetry system used in the papers by Barnes et al. [Barnes and Zelinsky, 2004]. The main difference is the use of more than one entry per radius segment. This leads to a considerably higher signal to noise ratio, meaning the ratio of maximum entry in accumulator to the applied threshold necessary to extract a maximum number of positions with high accumulator entries. The ratio is 1.55 rather than 1.45 when using just the one entry per radius segment as proposed by Barnes et al. This new scheme detects 16% more correct sign circles on a testset when using the algorithm described in the paper. Additionally 5% less background circle candidates are detected. The exact results are shown in table 6.8. The set used for this evaluation consists of 9359 frames containing 700 real world traffic signs, resulting in 8495 traffic sign circles of minimum size 15 pixels diameter in the images. The image sequences were gathered at different times of day and night, under differing lighting conditions, weather conditions and in all seasons of the year and are a subset of the main evaluation set. Computation time was given in the paper [Barnes and Zelinsky, 2004] as 20Hz in the year 2004, when 3 Ghz Pentium processors were the common CPU for desktop computers. The algorithm presented in this thesis computes on a 500 MHz CPU in less than 30 milliseconds per frame, thus a considerable speed up was achieved as well as a higher detection rate.

6 Experimental Investigations and Evaluation

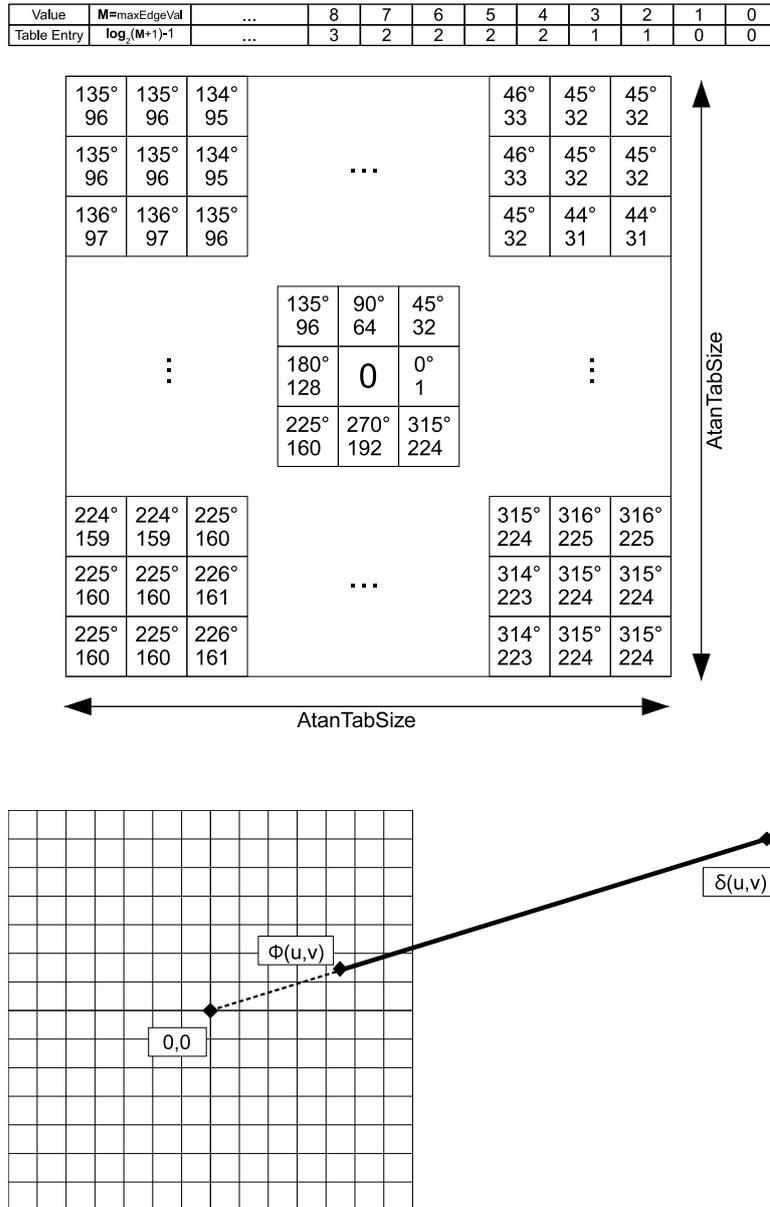


Figure 6.4: Shift table for entering the arc tangent table and the arc tangent table itself. Typical Values: MaxEdgeval for 10 Bit: 4095, for 12 Bit: 16383, AtanTabSize: 64. The size of 64 yields sufficiently low quantization noise for detector accuracy. Scheme of deriving Θ from gradient values u and v . See equation 6.4. The values in the table are the quantized gradient values (1-255) and the respective value in degrees.

Parameter	Barnes et al.	Proposed Algorithm
Signal to Noise Ratio	1.45	1.55
Correctly Detected Circles	7252	8441
Background Circle Candidates	24678	23427

Table 6.8: Comparison of the algorithm proposed in this thesis with the algorithm for circle detection proposed in [Barnes and Zelinsky, 2004]. The assessed values are the relative height of the peaks in the hough space (see above), the number of circles belonging to a traffic sign detected and the detected circles not belonging to a traffic sign.

Low Pass on Accumulator and Maximum Extraction

In this step the points with the highest number of entries in the accumulator image are extracted and entered into a list structure. Of the four accumulator entries per image coordinate the maximum value is stored in an additional image. A low pass filter is used on this maximum image and via a histogram computed on it the N coordinate points with the highest entries are selected and stored to the so called *island point list*. This list holds the value resulting from the blur of the maximum image as well as the four single radial accumulator values for this coordinate and of course the image coordinates of the point itself.

The algorithm is realized in a single pass procedure by computing the maximum value and the low pass on the maxima in a single run. The histogram is built in this pass as well. The threshold to be used on the maximum image to get the top N points is computed from the histogram and in a second pass over the image these points are extracted. This second pass is circumvented by using the threshold from the previous image on the current image and thereby extracting the points on the fly while computing the maximum image. This allows for an additional memory economy by keeping just the three lines of the maximum image which are necessary for the low pass filter. This approach is viable as long as the image scene does not change considerably, which is true for the given frame rate of 16 Hz.

The blurring step is necessary since the centre of the circle is generally not situated in the middle of a pixel and in the worst case lies on the border point of four pixels, where the contributions of all of them have to be accounted for. An additional factor enforcing the use of a low pass filter are the perspective distortions leading to the image of an ellipse where the two focal points have disjointed maxima close enough to get connected to one *island* by the use of the low pass. The imperfection in the determination of the edge direction itself leads to a spreading of the centre as well and thus calls for the blur too. Without the blurring the number of detected circles belonging to traffic signs decreases by 15%.

Primary Detector Candidate Generation

The list of points is submitted to an eight-connected components search, meaning that two pixels are considered connected when touching at the sides *or* the corners. This search is one pass since the points are ordered in the u and v image coordinates. Of the *islands* created the weighted centres are computed. This floating point image coordinate is the input for the following verification step as described in chapter 2.4.2. For the verification step an additional hint which radius the circle creating the *island* had. Since the Hough accumulator is quantized in four discrete radius sets the *islands* have quantized radius values as well. The four accumulator entries per contributing *island point* are added up in separate slots and the radius value belonging to the slot with the most entries is assigned to the *island*.

6.3.3 Detector Fine Positioning and Verification

The verification step is necessary for finding the pixel exact position of the centre and radius of the circle in the image, since the coarse detection was performed on the sub-scaled image and the radius quantised to the four values of the four discrete radius sets in the Hough transform. The five values to be optimized are:

- The number of sign circles detected relative to all signs (N_c).
- The number of signs of which both inner and outer circle were detected relative to the number of signs with two concentric circles (restrictions) (N_d).
- The accuracy of the centre point as the mean Euclidean distance (Δ_{uv}).
- The mean absolute radius error (Δ_r).
- The number of additional circles detected that do not belong to traffic signs, meaning false positives in the case of traffic sign detection relative to the number of signs in the set (N_{fpos}).

To accomplish this task two algorithms, explained in section 2.4.2, were tested and two sets of parameters adapted. The use of two parameter sets is necessary since in environments with very low lighting, e.g. night time and in tunnels, the camera sensor has a very high pixel noise. This made a second set of parameters necessary, allowing for lower height of the edges and the higher noise of the edges directions.

The first of the two algorithms is a chamfer matching approach, where a template matching is performed in a distance transformed edge image. The distance transformation results in an image showing the euclidean distance to the nearest edge pixel. The mean distance of a circle template to the nearest edge pixel is

computed by adding all values in the distance transformed edge image touched by the template and dividing by the number of pixels of the template. The second algorithm is a template matching algorithm matching edge position and direction. The parameters are adapted by modifying the free parameters of the algorithm and computing the corresponding five result values mentioned above. The optimal parameters are defined by being the ones yielding the maximum value for a linear combination of the five result values for all tested parameter sets.

6.3.4 Detection Verification Fine Tuning

There are a large number of parameters involved in the fine positioning and verification step. The verification sample set defined in section 6.2. It consists of 3559 traffic sign circles belonging to 1987 traffic signs, see table 6.1. It is used to determine the best set of parameters. Since most of the parameters are switches and not continuous values no gradient descent or similar optimization scheme was used, but a brute force grid search approach. The objective of the optimization is the best performance of the algorithm based on the five decision values (N_c , N_d , Δ_{uv} , Δ_r and N_{fpos}) explained in the previous section.

There are seven parameters used in the optimization. The first four parameters concern the generation and use of the edge image in the matching algorithms. One parameter is the type of matching used, either on the edge image directly or on the distance transformed edge image. The final two parameters are concerning the use of the template matching results.

The parameters optimized are in the following list:

- Type of edge detection operator (u-grad, v-grad accordingly):

$$\mathbf{I} : \begin{Bmatrix} -1 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{Bmatrix} \quad \mathbf{II} : \begin{Bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{Bmatrix} \quad \mathbf{III} : \begin{Bmatrix} -1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ -2 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 1 \end{Bmatrix} \quad (6.5)$$

- Edge threshold as defined in section 6.3.1.
- The tolerance angle ξ is set as the absolute difference between the phase of the gradient at the current point and that of the template, or in case of edge thinning the gradient of a circle centred at the point of the coarse detection. For edge thinning the sign of the phase gradient is disregarded allowing for black on white and white on black circles. The value for ξ is given in the metric used for the phase gradient, leading to

$$\xi' = 255 * \xi / (2\pi) + 1 \quad (6.6)$$

- Type of edge thinning:

Type I: Remove all phase gradient elements α computed as stated in 6.3.1 not fitting to a circle according to the tolerance angle.

$$\left\| \text{angle}(u, v) - \arctan\left(\frac{u}{v}\right) \right\| > \xi \quad (6.7)$$

Type II: Despeckle via morphologic operation, removal of singular pixels in the already filtered phase gradient image.

For the template detection the use of the Type I despeckle algorithm when not using Type II in addition is superfluous since the directional testing is inherent in the algorithm already. The use of the Type I despeckle is required for the Chamfer algorithm, since the clutter pixels created in the phase gradient image would reduce the maximum distance to the next set pixel strongly, thus producing a distance transformed image holding very low values only. The standard way to prevent this is the use of multiple directionally split distance transformed images and templates, but this scheme is too computationally expensive to be used here.

- Use of either the template matching or the Chamfer algorithm as explained in section 2.4.2.
- Low-pass filtering of the output of the matching algorithm, in radius dimension only to represent the possibility of the shape detected being slightly ellipsoid due to

$$\tau'(u, v, r) = \tau(u, v, r - 1) + \tau(u, v, r) + \tau(u, v, r + 1) \quad (6.8)$$

The value τ is the resulting matching value for a single template of the respective verification algorithm.

- Finally the decision threshold, which for the template matching algorithm is the percentage of matching pixels of the template, while for the chamfer algorithm it is the mean distance in pixels from the template to the next matching phase gradient pixel (θ_I, θ_{II}) .

The ranges of parameters are given in the table 6.9. The values applied for parameter ϵ are 16, 24, 32, 48, 64, 96, 128. The varying step width is used to reduce the necessary number of tests.

Due to the constraints for the use of edge thinning and low pass filtering explained above there are 11760 possible combinations of the parameters. For those parameter sets the resulting object list was computed on the set of 2124 images showing traffic signs declared above. The algorithm ran at approximately 35 Hz including image loading and result saving on the computer used, leading to a complete computation time of about eight days.

Parameter	Symbol	Range	Step	Num. of tests
Edge Type	η	I, II, III		3
Edge Threshold	ϵ	16 - 128	8 -32	7
Tolerance Angle	ξ'	10 - 30	5	5
Edge Thinning	χ	0, I, I+II		3
Algorithm Type	κ	1, 2		2
Low Pass Filter	λ	0, 1		2
Final Threshold Algorithm I	θ_I	0.15-0.90	0.05	16
Final Threshold Algorithm II	θ_{II}	0.50-2.00	0.10	16

Table 6.9: Table describing the experiments performed in the grid search for the best parameter set. The values are the range in which the respective parameter is varied, the step-width for this parameter in the grid and the number of tests that have to be performed on this parameter if all other parameters are fix. To get the whole number of experiments done in the grid search all numbers of tests have to be multiplied, in this case 322 560 experiments.

The task of the optimization is finding the maximum of the following linear combination of the factors influencing the circle position verification algorithm:

$$\begin{aligned}
D_V(N_c, N_d, \Delta_{uv}, \Delta_r, N_{\text{fpos}}) = \\
a1 N_c + a2 N_d - a3 \Delta_{uv} - a4 \Delta_r - a5 N_{\text{fpos}} = \quad (6.9) \\
f(a1, a2, a3, a4, a5, \eta, \epsilon, \chi, \kappa, \lambda, \theta)
\end{aligned}$$

The optimization problem is solved using a grid search. The positive factors are the number of detected or doubly detected signs (N_c, N_d) while the imprecisions in the detected circles position and sizes as well as the number of false positive circles are negative factors.

The detection function D is a linear combination of the five target values. Most important is the number of detected circles N_c followed by the number of signs where both inner and outer circles were detected N_d . The number of false positives N_{fpos} creates additional computational load for the following steps in the traffic sign recognition and hinders the association of circle objects in the tracker, but is deemed less an aggravating factor than the two positive factors N_c and N_d . The offset in u and v direction is more relevant than errors of the same proportion in the radial factor. Thus the factors for the optimisation were chosen from these experiences as: $(a1, a2, a3, a4, a5) = (1, 0.75, 0.5, 0.5, 0.25)$. The values for $a1 - a5$ were chosen by using the track classifier explained in 2.6.3 on a small set of traffic sign tracks. The results showed the influence of errors in the circle verification

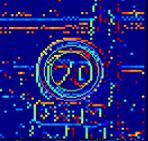
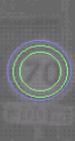
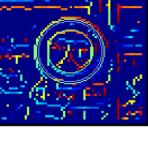
Edgetype	u-grad	v-grad	p-grad colour coded	result	result	normed cutout
I						
II						
III						

Figure 6.5: Example of the effects of the edge type on the shape verification result. The phase gradient edge images vary, producing differing positions and radii of the detected inner and outer circles. In this example the template matching algorithm was used.

step. The influence of the false positive circles could not be determined using this approach, since circles detected at random positions in the images have next to no effect, while the rejection of circular image objects is counter-productive, since it always removes traffic sign circles as well. Thus the value for a_5 was chosen as an experience value and not by a test set.

Due to high motion blur and sensor noise at night time or in tunnels a different set of parameters is used in case the camera is set to maximum exposure time or the mean grey value of the image is below the target mean value, in our case grey value 256 of 1023 maximum grey value.

As shown in table 6.10 the two different algorithms evaluated lead to similar results. The difference of in D_V of 0.74 to 0.73 at day and 0.61 to 0.60 at night are small. The Chamfer matching variant yields slightly higher detection results, while being less precise in the positioning of the circle. The modules following the detection process, namely the three dimensional positioning system and the classifier decline in performance when the coordinates and sizes of the circles is imprecise. Thus the template matching algorithm is chosen using the two parameter sets shown in table 6.11. The global evaluation of the traffic sign recognition system was performed on both the results of the template matching and the chamfer matching algorithm, producing results of comparable quality.

	Day				Night			
	Chamfer		T-Match		Chamfer		T-Match	
Opt. for max	$D_{V\ opt}$	$N_{c\ opt}$	$D_{V\ opt}$	$N_{c\ opt}$	$D_{V\ opt}$	$N_{c\ opt}$	$D_{V\ opt}$	N_c
N_c	0.90	0.91	0.89	0.90	0.95	0.95	0.90	0.95
N_d	0.79	0.83	0.79	0.83	0.71	0.86	0.59	0.78
N_{fpos}	0.48	1.18	0.89	1.71	0.69	0.97	1.00	1.58
Δ_{uv}	0.70	0.71	0.38	0.41	0.70	0.71	0.30	0.31
Δ_r	0.70	0.83	0.42	0.53	0.70	0.72	0.35	0.39
D	0.73	0.38	0.74	0.33	0.60	0.58	0.61	0.49

Table 6.10: Table of the eight best results of the parameter optimization for the detection verification. The results are split into brightness conditions (Day/Night), algorithm type (Chamfer/Template Matching) and objective of the optimization ($D_V(N_c, N_d, \Delta_{uv}, \Delta_r, N_{fpos})$ as weighted measure of the quality of the algorithm and N_c as number of correctly detected and accepted traffic sign circles). The evaluation set is shown in table 6.1. It can be seen that when optimizing for the number of detected signs N_c the value is only slightly higher than the result for N_c when optimizing for the over all measure D , while the values other measures are partly very low when compared to the results for the D optimization. Thus showing the advantage of not only taking N_c into account in the optimization process.

6.3.5 Summary of the Detector Results

The coarse detector based on a fast hough transform is capable of detecting over 98% of the traffic sign image circles of a minimum size of 15 pixels in diameter. The verification and fine positioning step is accepting 90% of the traffic signs circles and positions them to less than 2 pixels error in centre position and radius. Over 95% of the signs are detected when allowing up to 3 pixels error. The rate of non sign circles detected and verified is less than 0.5 per frame. All free parameters of the detection algorithms are adapted by optimization of the results on an evaluataion set. The detector consisting of coarse detection and verification step is algorithmically optimized to be executed in less than 30 ms per 752x320 pixel image on a 500 MHz computer without the use of hardware specific optimizations.

6.4 Tracker

The evaluation of the tracker is mainly based on the number of correctly tracked circle objects. The evaluation set consists of 664 real world traffic signs consisting of 13487 image circles to be tracked in 11874 of the 91146 frames. In addition

	η	ϵ	ξ'	χ	κ	λ	θ	D_V
Day	II	32	15	I + II	Template Matching	1	0.30	0.74
Night	III	24	15	I + II	Template Matching	1	0.25	0.61

Table 6.11: Verification step of the circle detection, table of the parameters for the best results for optimization of D_V , see table 6.10. These values will be used in the following for the verification step of the detector. For the explanation of the values, see list in section 6.3.1

there are 45740 circle image objects not belonging to traffic signs, but possibly forming non traffic sign tracks. Two types of trackers were implemented, one for the use in case no ego motion information is available and one if the information is available. For both types of tracking algorithms the values to be optimized are the same. The best case are tracks tracked completely from the first to the last circle belonging to the traffic sign object without wrongly adding objects to the start or end of the track and no non traffic sign circles are connected to tracks.

The optimization of the parameters of the tracking algorithm is based on one favourable and four negative factors

- The number of detected traffic sign objects tracked for at least 3 frames (n_0). This is the favourable factor.
- Number of *broken* tracks. This is another common error, especially when no ego motion information is available, the breaking of tracks in two or more unconnected sub-tracks (n_1).
- Number of late starts or early stops of tracks thus reducing the number of tracked circle objects (n_2).
- Number of tracks with elements added wrongly to a track (n_3).
- Number of tracks built of non traffic sign objects (n_4).

The optimization function is defined as a linear combination of the five influencing factors, with N the number of labelled traffic sign tracks :

$$D_T(n_0, n_1, n_2, n_3, n_4) = \frac{1}{N} (b_0 n_0 - b_1 n_1 - b_2 n_2 - b_3 n_3 - b_4 n_4) \quad (6.10)$$

The most influence on the system result besides the real world signs not being tracked at all, have the number of *broken* tracks since these shorten the tracks strongly, thus complicating the classification process and the three dimensional pose estimation. The reduction of the track lengths by either losing elements at the begin or end of the track influences the classification result, especially for short

tracks. For the tracker a small number of wrongly added elements at the begin or end of the track is of minor consequence due to the following classification step. The last factor is the number of tracks not belonging to traffic signs. Some of those tracks are inevitable since they belong to circular real world objects and are thus in the tracker stage indiscernible from traffic signs. Other tracks are wrongly connected circles belonging to random objects having a circular form in the image. The negative effect of these tracks is the additional computational load on the system and the additional possibilities for false positives in the classification step. The weights in the evaluation function are chosen according to their relevance as:

$$b_0 = 1, b_1 = 0.5, b_2 = 0.25, b_3 = 0.1, b_4 = 0.05 \quad (6.11)$$

The values for $b_1 - b_4$ were determined by classifying a small number of tracks, actively applying the four tracking faults and observing the errors introduced by the tracking faults.

The use of the ego motion of the vehicle and thus the motion of the camera allows a much better prediction of the position of the appearance of the circular object in the next frame and has a different set of parameters to be optimized than the tracker without the use of ego motion. Thus the optimization was split into two separate parts. One part for optimizing the necessary parameters for tracking without ego motion, as it is necessary in case of unavailability of vehicle data as it is the case for mobile devices as smart phones or if there is no connection to the vehicle's internal bus system.

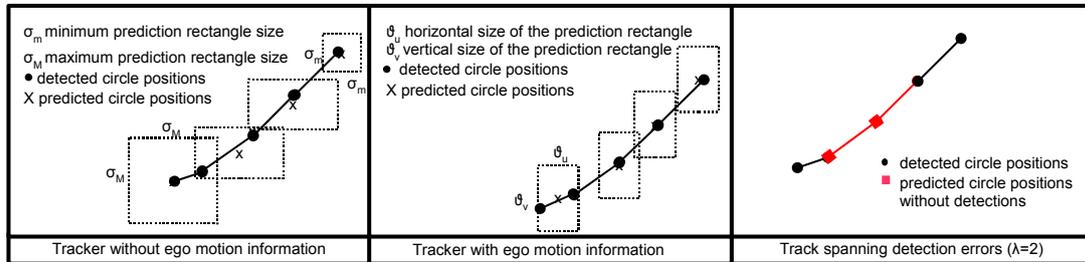


Figure 6.6: Figure explaining the parameters to be optimized in the tracker.

6.4.1 Tracker without ego motion

Should no information about the ego motion of the vehicle be available, the process predicting the position of an object in the next frame from the position detected in the current frame is based on image information alone. The $\alpha\beta\gamma$ – tracker algorithm is explained in section 2.5.2.

One of the main parameters influencing the behaviour of the tracker are the expected inaccuracy of the prediction and thus the size of the area in which the next element of a track is expected. This rectangular area starts at a defined size

for finding the second element for a track and is adapted relative to the accuracy of the previous prediction for the current circular object. The defined size at the start of the tracking process is the maximum size of the rectangle. To prevent the tracker from reducing the size of the rectangle too much, should there have been a predicted position very close to the continuing circle, the second parameter for the tracker is the minimum size for the rectangle. The next parameter is the number of frames where there are no detected circles added to a track, but the track is still labelled as not having ended, thus how many frames the tracker is allowed to skip before breaking the track. The last parameter is the α variable of the tracker, β and γ are derived from the α value, see table 2.1 for the use of constant acceleration. The main parameters to be optimized are summarized in the following list and explained in figure 6.6:

- Minimum size of the rectangular area around the predicted position of the circle to look for the next circular track element in pixels (σ_m).
- Maximum size of the rectangular area around the predicted position of the circle to look for the next circular track element in pixels (σ_M).
- The maximum number of frames to predict the position of a following object before ending a track (λ).
- The tracker constant defining the low pass property of the tracker (α).

To find the optimal parameter set for the tracker the value for D_{T1} in equation 6.12 has to be maximized on the given sampe set. This is done as for the parameters for the circle verification step in section 6.3.4 by the use of a grid search.

$$D_{T1}(n_1, n_2, n_3, n_4) = f(n_1, n_2, n_3, n_4, \sigma_m, \sigma_M, \lambda, \alpha) \quad (6.12)$$

The ranges of parameters are given in the table 6.12. This leads to 400 tests to be run for the optimization of the tracker.

The optimum was found at $\sigma_m = 15$, $\sigma_M = 45$, $\lambda = 1$ and $\alpha = 0.85$. As can be seen in the table 6.13 for the variation of σ_M the maximum is comparatively flat, allowing for minor changes of the parameters without large changes in the results. The same is true for the other parameters, allowing the retaining of the parameters even if there are minor changes in the over all traffic sign recognition system.

6.4.2 Advantage of Ego Motion Information

When the ego motion of the vehicle and thus that of the camera is known the position of the next element in a track can be determined much more accurate that without this information. This reduces the number of all the four negative factors, broken tracks, early stops, wrong continuation and non sign tracks (n_{1-4}), in the

Parameter	Symbol	Range	Step	Num. of Tests
Minimum size of prediction area	σ_m	5-25	5	5
Maximum size of prediction area	σ_M	35-65	10	4
Maximum prediction number	λ	0-3	1	4
The tracker constant	α	0.80-0.95	0.05	5

Table 6.12: Parameter ranges to be tested in the optimization of the tracker without the use of ego motion information. The *Range* gives the minimum and maximum value to be checked for this parameter. The *Step* defines the step width from the minimum value to the maximum value in the *Range*. The *Num. of Tests* is derived by counting how many *steps* have to be made from the minimum to the maximum *Range*. The number of tests necessary for the complete optimization is derived by multiplying the single numbers of tests necessary. $\Pi(\text{Num.ofTests}) = 400$.

above optimization. The base for this kind of tracking is the three dimensional position estimation as in section explained 2.5.4. The estimation is based on finding the point closest to all lines of sight from the respective camera position to the centre of the detected sign in all images where it was detected.

Since the first estimation of the three dimensional position can be made only after the second detection of the object, the first distance estimation $|x_0, y_0|$ has to be derived differently. It is based on the diameter of the first detected circle d_0 [pixel]. This diameter in pixels translates via knowledge about real world diameters of traffic signs with minimum size s [m] and maximum size S [m] and the focal length f [m/pixel], see equation 6.13.

$$|x_0, y_0| \in (sf/d_0, Sf/d_0) \quad (6.13)$$

When the current position relative to the camera (x_n, y_n) and the ego motion of the camera (d_{n+1}, α_{n+1}) is known, the following position (x_{n+1}, y_{n+1}) and deriving from that the expected pixel positions u_{n+1}, v_{n+1} can be estimated as depicted in figure 6.7 as:

$$\begin{aligned}
(x_{n+1}, y_{n+1}) &= (x_n - d_{n+1} \cos(\alpha_{n+1}), y_n - d_{n+1} \sin(\alpha_{n+1})) \\
\beta_{n+1} + \alpha_{n+1} &= \arctan\left(\frac{x_{n+1}}{y_{n+1}}\right) \\
\beta_{n+1} &= \arctan\left(\frac{x_n - d_{n+1} \cos(\alpha_{n+1})}{y_n - d_{n+1} \sin(\alpha_{n+1})}\right) - \alpha_{n+1} \\
u_{n+1} &= -f\beta_{n+1}
\end{aligned} \quad (6.14)$$

The vertical position v_{n+1} is computed analogue from the real world position in longitudinal and vertical direction (x_{n+1}, z_{n+1}) . The negative leading sign for

σ_M	D_T	n_0	n_1	n_2	n_3	n_4
35	0.644	645	25	7	4	4054
45	0.646	646	21	6	5	4079
55	0.646	644	20	6	6	4087
65	0.634	643	20	5	9	4180

Table 6.13: Exemplary table for the optimization results for the tracker without the use of ego motion information, where the maximum prediction area σ_M was varied, while the other parameters were set to minimum prediction area $\sigma_m = 15$, maximum detection misses $\lambda = 1$, and tracker constant $\alpha = 0.85$. The number of labelled tracks N is 664 tracks belonging to traffic signs, as shown in table 6.2. This is a one dimensional view of the results of the grid search.

u_{n+1} is owed to the fact that in the real world coordinate system left is positive while in image coordinates right is defined positive.

From the distance to the sign and the movement of the camera the next image position of the sign can be predicted. There are imprecisions in detection of the centre of the circle, the ego motion estimation, the real world position of the sign relative to the camera and the unknown pitch angle between the frames the sign was detected in. To allow for these imprecisions, the image region where the current track can be continued by a new circle in the next frame is a rectangle enclosing the estimated next position of the track. The parameters to be optimized involved in the ego motion tracking are explained in 6.6 and defined below:

- The base size of the rectangle enclosing the estimated position in horizontal direction in pixels ϑ_u .
- The base size of the rectangle enclosing the estimated position in vertical direction in pixels ϑ_v .
- The maximum number of frames to predict the position of a following object before ending a track (λ).

The ranges of parameters are given in the table 6.14. This leads to 80 tests to be run for the optimization of the tracker.

The result of the optimisation is the parameter set $\vartheta_u = 7$, $\vartheta_v = 10$ and $\lambda = 1$. The resulting score is slightly higher than the one for the tracker without the use of ego motion information. This mainly due to the lower number of *broken* tracks and less non-traffic-sign tracks. The result is shown in table 6.15 the row with the parameter setting 7 for ϑ_u . The 3887 tracks formed of non sign circles consist

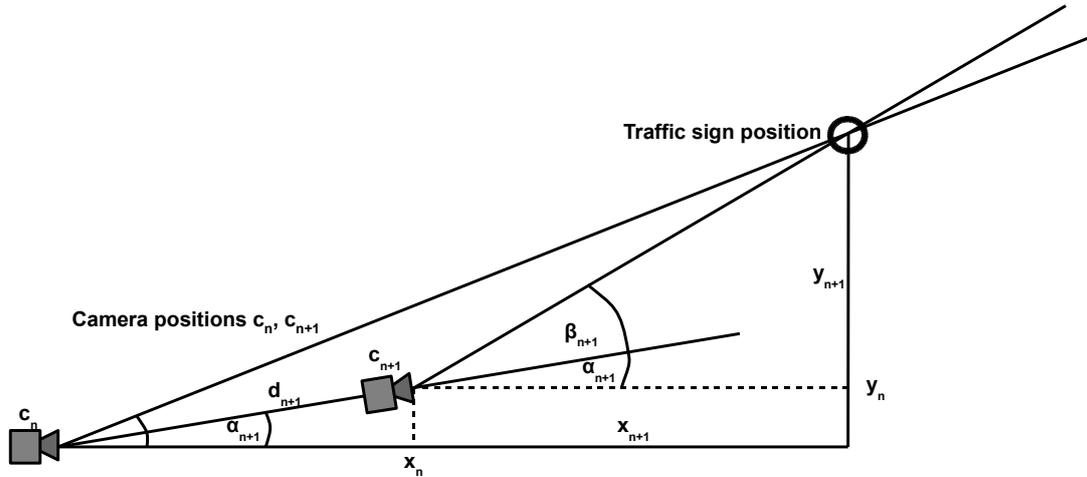


Figure 6.7: Explanation for the variables used in equation 6.14. Camera at position c_n with the vector to the sign (x_n, y_n) , camera at position c_{n+1} , being d metres from c_n in the direction α_{n+1} , with the vector to the sign (x_{n+1}, y_{n+1})

of 38953 of the non traffic sign circles, most of the circles belonging to real world circular objects. 7193 circles were singular objects not connected to tracks.

6.4.3 Three Dimensional Pose Estimation

The three dimensional pose estimation is not only used for the prediction of the next sign position in the tracking process, but is also important for the validation of the traffic sign by checking the plausibility of the size and world position of the sign relative to the vehicle. The position of the sign is also important for the determination of the valid velocity on the given lane or road if there are parallel lanes with different maximum speed allowances.

The first experiments for the evaluation of the accuracy of the three dimensional pose estimation were based on placing four signs in defined relative positions, a rectangle of 10 m length of edge at a height of 2m above ground. The vehicle was then driven between these signs in different driving manoeuvres. The reconstruction of the rectangle succeeded with an error of less than $\pm 0.4m$ in mean edge length of the rectangle and less than $0.2m$ in height.

Further test using a ground truth sensor of very high precision performed to verify the results reached by the usage of the new sensor. The results of the tests have been published in [Lindner, 2010]. The sensor used is the multi-row laser scanner from Velodyne ([Glennie and Lichti, 2010]). The scanner has 64 vertical scan lines and yields 5 pixel per degree, leading to 12.8 degrees opening angle in vertical direction. The horizontal resolution depend upon the rotation rate of the scanner. The selected speed of 10 rotations per second leads to a horizontal

Parameter	Symbol	Range	Step	Num. of Tests
Min. horiz. size of pred. area [pix.]	ϑ_u	5-11	2	4
Min. vert. size of pred. area [pix.]	ϑ_v	5-25	5	5
Max. prediction number	λ	0-3	1	4

Table 6.14: Parameter ranges to be tested in the optimization of the tracker with the use of ego motion information. The *Range* gives the minimum and maximum value to be checked for this parameter. The *Step* defines the step width from the minimum value to the maximum value in the *Range*. The *Num. of Tests* is derived by counting how many *steps* have to be made from the minimum to the maximum *Range*. The number of tests necessary for the complete optimization, 80, is derived by multiplying the single numbers of tests necessary.

resolution of 5.5 pixels per degree. The distance accuracy is given as $0.02m$, which is adequate for the evaluation of the three dimensional positioning module having shown the accuracy to be of the order of $\geq 0.2m$ in the previous tests. The position of the scanner relative to the camera position was calibrated using the algorithm presented in [Krueger et al., 2011].

As can be seen in figure 6.8 the scanner delivers ambiguous data on the retro-reflective surface of the traffic sign. To avoid this problem the position of the sign was determined by measuring the position of the pole holding the sign and adapt the vertical position by adding half the diameter of the sign, thus retrieving the centre position of the sign.

Four values are determined for the evaluation:

- Difference between the ground truth longitudinal distance and the measured distance $\Delta x = x_{GT} - x_M$.
- The condition of the determining matrix k as shown in equation 2.25, respectively the inverse $k' = 1/k$
- The mean Euclidean distance between the lines of sight from the camera to the measured three dimensional position of the sign, the residuum Δd .
- The standard deviation of the measured three dimensional position when applying noise on the ego motion data and the detected sign positions in the image σ .

The lateral and vertical errors are dependant of the longitudinal error and smaller than the Δx value due to the camera setup and thus not further evaluated.

As shown in figure 6.9 there are 20 ground truth sequences. In the real world scenario quite often the traffic sign track ends before reaching the border of the

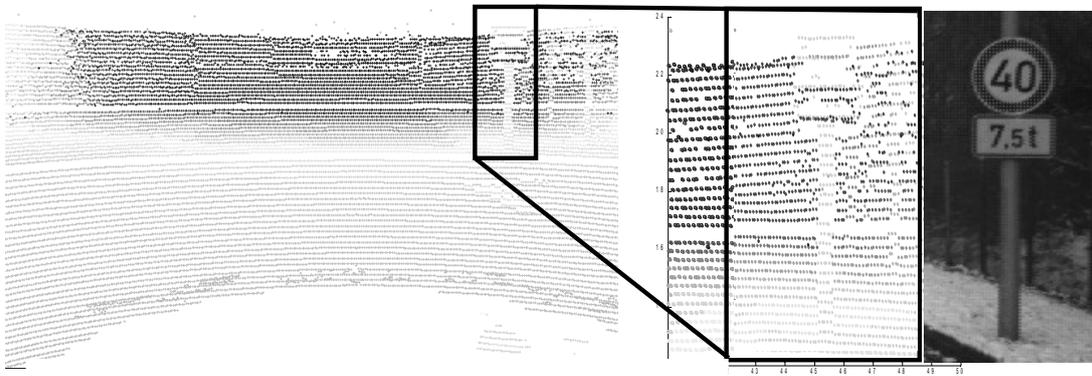


Figure 6.8: Image showing the laser scanner result, depth coded in grey value, and the according video image of the sign in question. On the left is the depth image taken by the laser scanner. In the middle is the zoomed section of the scanner image showing the traffic sign. On the right is the according section of the camera image.

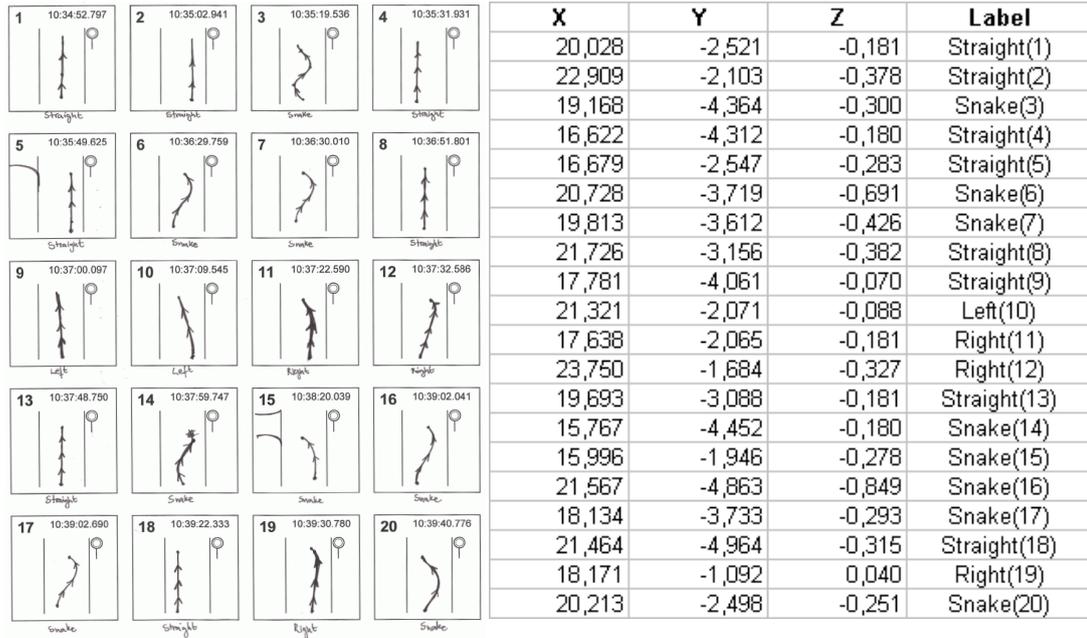


Figure 6.9: The graph shows the ground truth set used for the three dimensional positioning system. On the left are sketches of the 20 manoeuvres driven in the ground truth sequences. In the table on the right are the according sign positions relative to the camera position at the first positive classification of the traffic sign (X,Y,Z). The labels straight, left, right and snake are rough categorizations of the trajectories driven, with snake being a curve with a double bend. The velocity during the manoeuvres is between 25 and 40 km/h and the direction change during one trajectory is up to ± 15 degrees. Of each manoeuvre the last 50 image frames were analysed, resulting in 1000 analysed frames showing 20 traffic signs. 1306 circles were detected and tracked belonging to 807 frame sign objects, the additional circle being the inner contour of the detected signs.

$\vartheta_{\mathbf{u}}$ [pixel]	D_T	\mathbf{n}_0	\mathbf{n}_1	\mathbf{n}_2	\mathbf{n}_3	\mathbf{n}_4
5	0.676	645	4	7	1	3850
7	0.676	646	2	6	2	3887
9	0.677	646	2	6	2	3905
11	0.668	645	1	5	4	3978

Table 6.15: Exemplary table for the optimization results for the tracker with the use of ego motion information, where the horizontal prediction error expectancy ϑ_u was varied, while the parameter for the vertical prediction error expectancy was set to $\vartheta_v = 10$ and the number of detection misses to be accepted was set to $\lambda = 1$. The number of labelled tracks N is 664 tracks belonging to traffic signs, as shown in table 6.2. This is a one dimensional view of the results of the grid search.

image due to occlusions by infrastructure or other traffic participants. To simulate this behaviour the results on the ground truth tracks were computed on the full tracks as well as on truncated tracks, removing the nearest or farthest elements of the track. Thus the number of evaluated tracks is higher than the number of ground truth tracks.

To compute the value for σ as introduced in the above list, the computation of the three dimensional position of the sign is executed for the complete and the truncated tracks with a Gaussian noise applied on the measuring data of the signs position in the image, the vehicle yaw rate and velocity. The standard deviation for the noise on detected position was chosen as 10% of the detected circle diameter, 10% of the yaw rate and 5% of the velocity.

The mean error over the 8 ground truth sequences where the vehicle was going straight ahead is $\Delta x < 0.01m$ with a standard deviation of 0.20m. The mean error for all 20 sequences is 0.16m with a standard deviation of 0.32m. The inverse condition for all the complete tracks is in the order of 10^{-2} as can be seen in figure 6.10.

In figure 6.11 the connection between the condition and the accuracy of the three dimensional measurement can be seen. The squares are the median values of the errors $|\Delta x|$ and the triangles the 90% quantiles, meaning 90% of the error values were beneath this value. The line is derived by taking the logarithmic value of $|\Delta x|$ and k' and fitting a straight line into these values. The line is then transformed back into the linear plot. In the traffic sign recognition system the median and the 90% quantile value for a given k' can be used as expected error and upper bound error. This correlation between the accuracy of the algorithm and a value k' allows the indication of the expected position error for result three dimensional positioning system for a traffic sign track without a ground truth

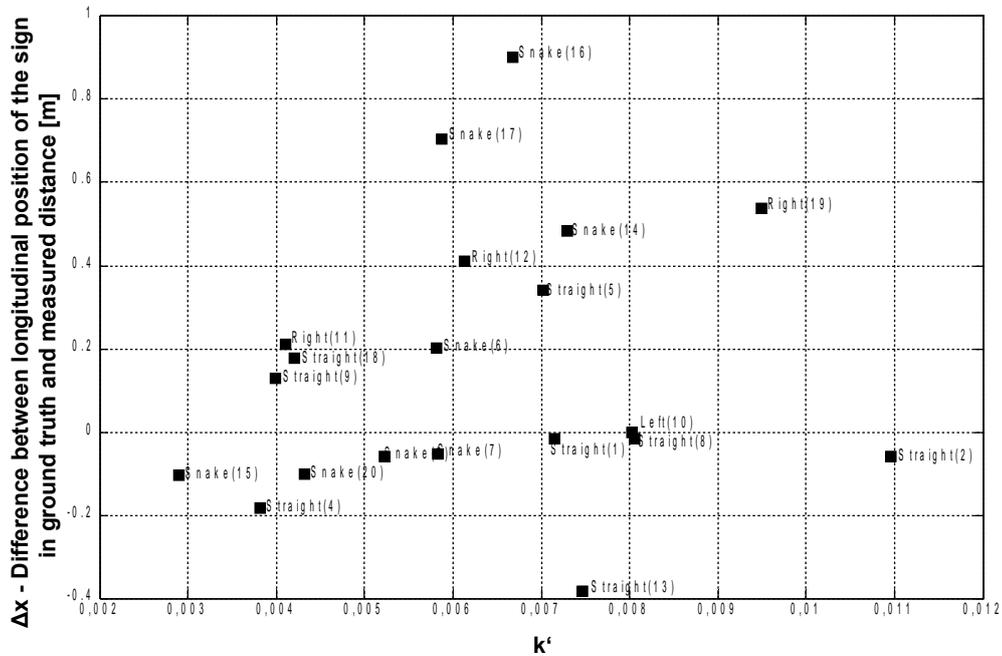


Figure 6.10: This figure shows the three dimensional positioning systems error for each of the 20 manoeuvres of the ground truth set introduced in figure 6.9. The value of the longitudinal distance error Δx of the measured sign position compared to ground truth position obtained by the use of the laser scanner is plotted over the inverse condition of the determining matrix k' , see equation 2.25, for the complete tracks. k' gives a rough estimate for the expected accuracy of the positioning systems result since it measures the *circularity* of the problem. The higher the condition k , thus the lower the inverse k' , the more elliptical the problem, thus the more inaccurate the result given noisy input data. When for each of the 20 sequences all detected sign circles are used in the computation the condition is generally *good* and the error below 1 metre in longitudinal direction, which is more than adequate for the traffic sign recognition systems requirements.

sensor being installed.

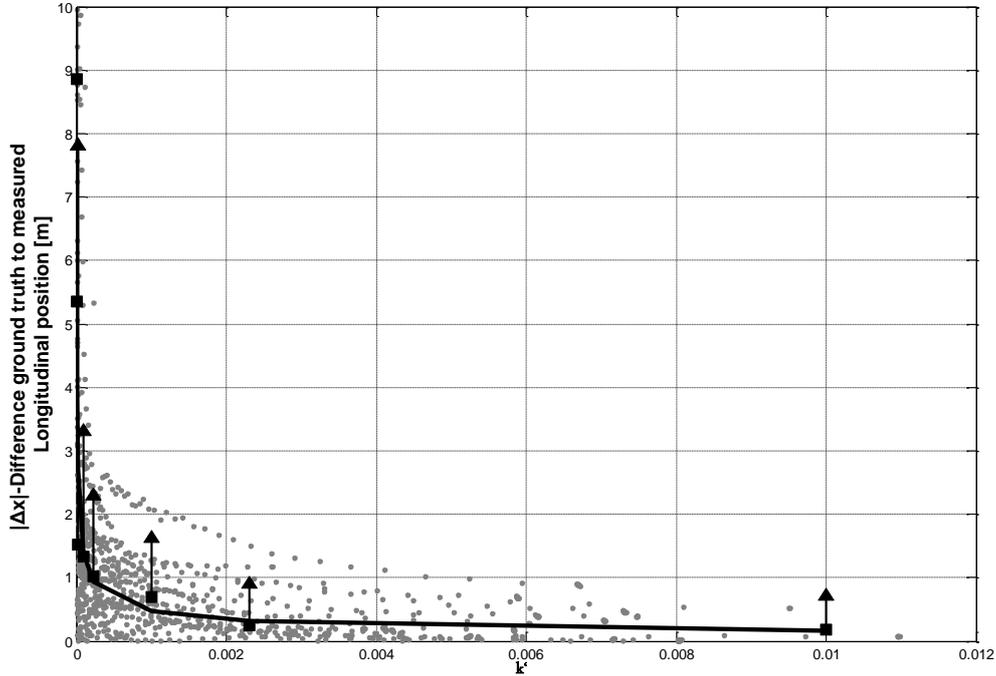


Figure 6.11: The value of the absolute longitudinal distance error $|\Delta x|$ over the inverse condition of the determining matrix k' for complete and truncated tracks. The distribution over the condition value k' shows a correlation between the distance error and the inverse condition k' , with k' being defined in equation 2.25 and giving a measure for the *circularity* of the problem and thus the expected accuracy of the result. The twenty tracks recorded with ground truth information had a mean number of 40 tracked circles. The results when using the complete tracks are shown in figure 6.10. When not using all elements of the track, but truncating them by removing the last 1..N elements shorter tracks with worse condition values were simulated. As can be seen in the graph the error rises exponentially with rising condition value k , thus decreasing k' . The black line shows the median error, the triangles show the 90% quantile.

The correctness and accuracy of the three dimensional positioning algorithm was proven by the use of the laser scanner sensor. To achieve a fast and inexpensive sample set a second way of evaluating larger sets of sequences was developed. The application of Gaussian noise on the input data of the algorithm as stated above could be used to produce a similar behaviour in the standard deviation of the result over k' as the absolute longitudinal distance error, showing that the error model of 10% error on the circle centre positions, relative to the signs diameter, and yaw rate as well as 5% on the vehicles velocity value are adequate

assumptions. With these assumptions sequences recorded without the laser scanner can be used to evaluate the accuracy of the sign position and the correlation of k' and the expected error, too.

Self Calibration Using the Three Dimensional Positioning System

The camera should have been installed horizontally with the point of expansion u_0, v_0 being in the centre of the image and the position relative to the turning centre of the vehicle according to the Ackermann model l_x, l_y being known. Of course these values are not perfect. For re-adjusting these four values in the system the three dimensional positioning algorithm is used. The vertical position of the camera relative to the vehicle can not be observed in this system due to the unknown pitch rate of the vehicle. The self calibration of the system can be solved by using the mean distance of the lines of sight to the estimated sign position $|\Delta d|$. When the system is perfectly calibrated and all measured input values for a track are correct all lines will intersect in a point. When the values for the point of expansion are incorrect the lines will not intersect, but be skewed and have a distance to the closest point for the signs that grows with the error of the point of expansion.

For the tracks to be used in the self calibration the resulting position has to be trustworthy, inducing a high value of k' . Tracks showing a low reliability of the three dimensional position due to being short or belonging to far off signs are not considered. For tracks driven straight ahead, meaning a low yaw rate ($|\text{yaw} - \text{rate}| < 0.04\text{rad/s}$) while the sign is detected, the input values for the three dimensional positioning system are recorded. For the calibration the point of expansion u_0 is varied and the $|\Delta d|$ is stored for the different values of u_0 . This is done for a number of sequences, here for all eight *straight* tracks. The mean $|\Delta d|$ can be plotted over the point of expansion, the minimum showing the correct value. The plot is shown in figure 6.13. In addition the curve showing the mean absolute error over the point of expansion is shown and has the same minimum, thus proving the algorithm to be viable. The same algorithm is used for the vertical position of the point of expansion v_0 .

For the measurement of the longitudinal distance of the camera to the turning centre of the vehicle (l_x), as explained in section 2.5.5, the sequences with a high mean yaw-rate and a high inverse condition $E(k) > 0.1$ are used. Once a stable point of expansion is established by the use of the above algorithm, the value for l_x is computed by varying l_x and finding the point of lowest mean value for the according residual Δd . The plot for the determination of l_x is shown in figure 6.14.

The experiments above prove the viability of the three dimensional positioning algorithm, showing an adequate standard deviation of the absolute error of far less than 0.5 m in the lateral direction and thus an even smaller error in the lateral direction due to the geometry of the sensor. The use of the condition

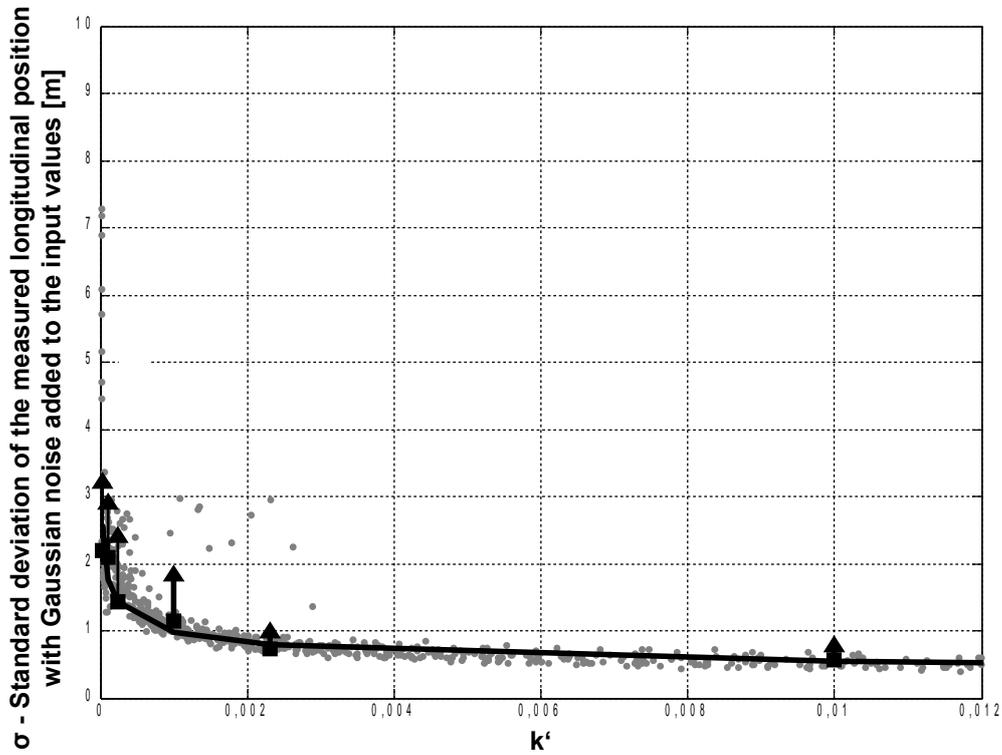


Figure 6.12: Results when Gaussian noise is added to the input values of circle centre position, yaw rate and position of the twenty ground truth tracks. 100 random tests were made and the value of the longitudinal distance standard deviation σ over the inverse condition of the determining matrix k' for complete and truncated tracks. Truncated track means a track of which the last 1..N Elements were removed to simulate a shorter track and thus a track with worse condition k . The errors made by varying the input variables of the algorithm produce a similar error expectancy as shown in figure 6.11 for the error of the algorithms measurement result distance against the ground truth distance. Thus should the camera geometry, like pixel per degree or opening angle, be changed, the distance error measured to be expected can be predicted without the use of the ground truth laser scanner.

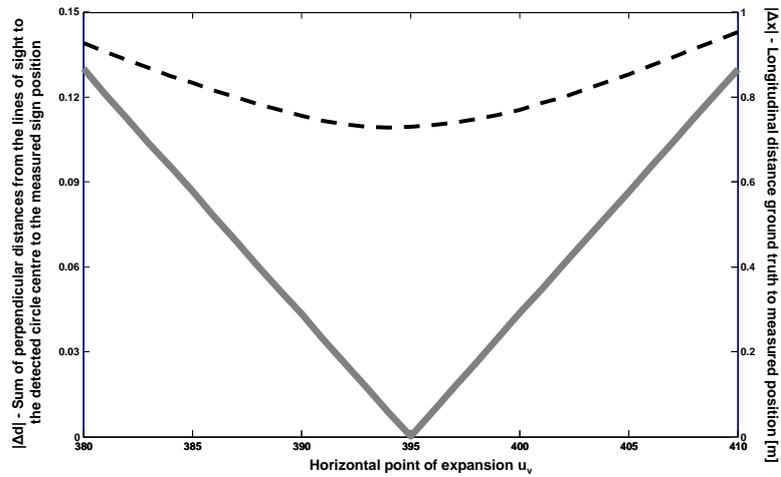


Figure 6.13: Determination of the point of expansion via minimization of the mean Δd residual value. The optimization curve is the dotted black line (Distance to ground truth in grey). The figure shows the validity of the self calibration algorithm, since the distance error produced by the three dimensional positioning system is lowest at the point of the best point of expansion determined by the self calibration.

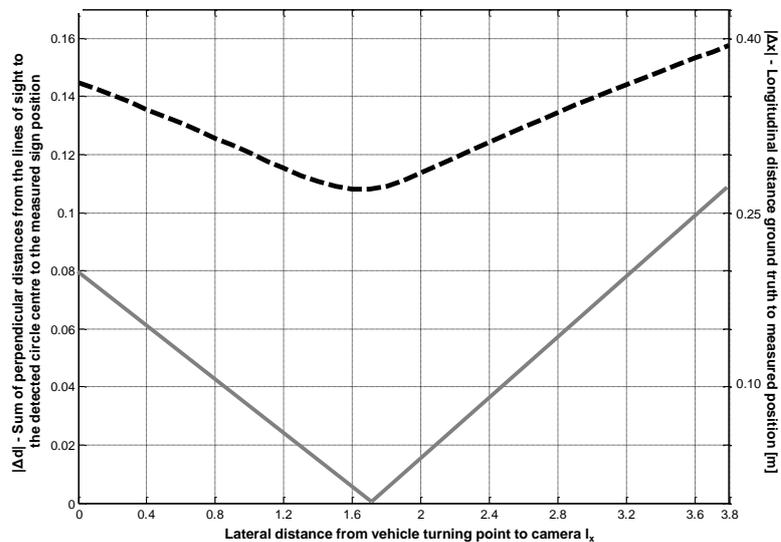


Figure 6.14: Determination of the lateral distance from the camera to the vehicle turning centre l_x via minimization of the mean residual Δd value. The optimization curve is the dotted black line (Distance to ground truth in grey). The figure shows the validity of the self calibration algorithm, since the distance error produced by the three dimensional positioning system is lowest at the point of the best distance to the vehicles turning point determined by the self calibration.

of the determining matrix as parameter indicating the expected error has been shown. The feasibility of the self calibration of the important parameters for the three dimensional measurement algorithm has been demonstrated.

The horizontal position of the point of expansion u_0 can be measured with an accuracy of ± 5 pixels, the vertical position v_0 with an accuracy of ± 10 pixels and the longitudinal distance to the turning centre of the vehicle l_x with an accuracy of ± 0.20 m. The lateral position relative to the turning point l_y has a very small influence on the distance measurement and is thus not important for the three dimensional positioning system. The accuracy for l_y is very low with ± 3 m and thus far lower than simple visual judgement when installing the camera. The vertical camera position is not observable with the system at hand.

Results and Use of the Three Dimensional Position

In table 6.16 the typical longitudinal measurement errors for a traffic sign on rural roads, meaning a lateral offset of about 3 meters is shown. Without occlusions the sign will leave the visible area at about 10 meters longitudinal distance. On highways the signs have a lateral distance of up to 3 lanes width amounting to about 12 meters and the sign leaving the image at about 30 meters longitudinal distance.

Longitudinal distance at last detection [m]	50	40	30	20	10
Longitudinal measurement error [m]	> 10	6	2.5	0.8	0.3

Table 6.16: Table for the mean longitudinal measurement error of tracks with a lateral distance of 3m measured, using a set of 100 sequences specially recorded and distance measured from this purpose.

The three dimensional position is used for the understanding of the traffic scene. In figure 6.15 an example where the three dimensional positioning system helps rejecting a sign is shown. The sign showing a 70 is mounted on the back on a lorry. By producing a very high distance and diameter value as well as a high residual Δd value it points to an object that does not fit to the model of being motionless, besides being in the same scene as two signs having plausible values for sizes, distances and the residual value Δd . Especially for signs mounted on the back of other vehicles and for signs situated on parallel roads the three dimensional positioning system is valuable for the interpretation of the traffic scene.

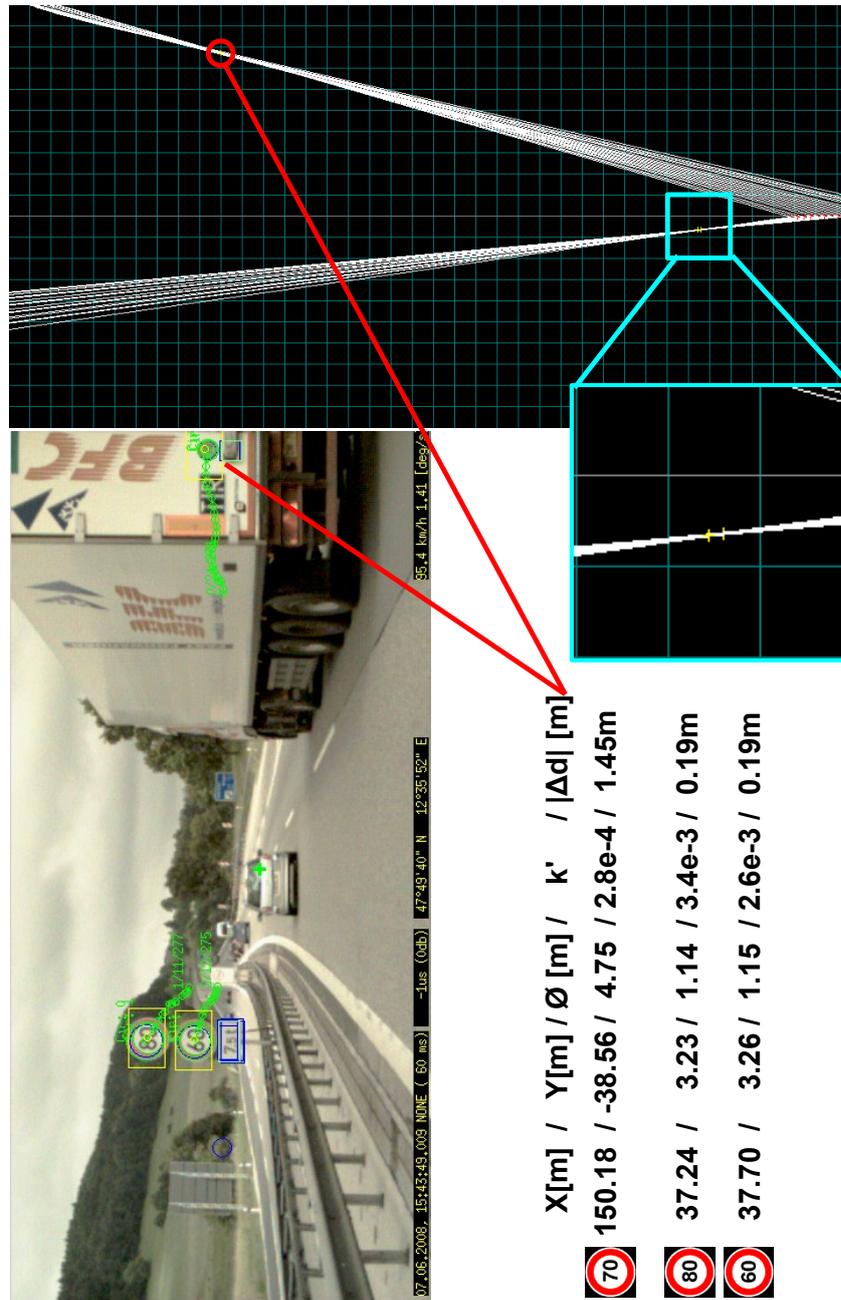


Figure 6.15: Example where the three dimensional position of the signs helps to reject a 70km/h sign attached to the back of a lorry. The values in the table are the measured relative position to the sign when first detected (X,Y), the measured diameter (\emptyset), the inverse condition of the track (k') and the mean distance of the lines of sight to the measured sign position ($|\Delta d|$). Due to the very high computed sign diameter and the comparatively high $|\Delta d|$ value the 70 km/h sign is rejected.

6.4.4 Summary of the Three Dimensional Positioning Systems Evaluation

The three dimensional positioning algorithm has sufficient accuracy for reconstructing a traffic sign scene on a strip of road, allowing the assessment of the currently valid speed limit. It also allows the rejection of most signs mounted on the back of lorries. In addition it can be used for self calibration of the parameters necessary for its operation like position of the camera relative to the vehicle turning centre and its rotation relative to the longitudinal axis of the vehicle.

6.5 Over all Performance of the Detector and Tracker

In this section statistics of the detector and tracker are presented. The table 6.17 shows the selection capabilities of the detector and tracker necessary for the traffic sign recognition system. The sequence is the same used for the tracker verification evaluation, see section 5.4.

	Number of image objects	Mean number of image objects per frame
Possible circle objects	$4.10 \cdot 10^{11}$	$4.49 \cdot 10^6$
Detected coarse circles	483229	5.31
Verified circles	59227	0.64
Tracks of circles	4533	0.05
Non sign circles tracked	38953	0.43
Traffic sign cir. tracked	13081	0.14
Verified circles not tracked	7193	0.08

Table 6.17: Table showing the reduction of sign candidates from the possible circle objects in the recorded frames to the number of tracks further inspected by the classifier. The number of tracked circles shows the processing load on the traffic sign classifier system. The base set consists of 91146 frames showing 13487 traffic sign circles in 664 traffic sign tracks. The left column showing the sum of objects in all frames, thus the number in the right column multiplied by the number of frames. The number of possible circle objects equals the number of tests necessary if a brute force correlation for circular shapes was used for the radius values from 15 to 59 for every second image pixel, thus the proposed coarse detector reduces the number by roughly 6 orders of magnitude and the verification by another order of magnitude, thus reducing the computational load for the classification process to three non-sign circle objects for every sign circle object.

	Traffic signs	rel.	Non sign tracks	rel. to signs
Tracked	646	97.3 %	3887	585.4 %
Mean track length	20.2	100.0 %	10.0	49.5 %

Table 6.18: Table showing the tracking results of sign and non sign objects side by side in absolute and relative values. The number of all sign tracks is 664, the evaluation set is given in table 6.2.

From the 664 real world traffic signs 646 (97.3%) were detected, tracked and handed for recognition to the classification system. The remaining 2.7% were not detected or detected just once and thus not tracked. The tracker did not loose any sign object of which at least two circles were detected. From the 59277 verified circles 7193 were singular objects not building a track, since no continuing circle object could be found in the following frames. The mean number of circle objects in tracks not belonging to a traffic sign is 10.0, while the mean number in a track formed by a traffic sign object is 20.2 circle objects.

The two tables 6.17 and 6.18 show that over 97% of the traffic signs are detected and tracked. The numbers also show, that even with the huge reduction of possible circles in the images possibly showing traffic signs by a factor of about 10^7 , the main workload for the following classification system consists of *cutouts* and tracks not belonging to traffic signs. This means six times more tracks detected belong to non-sign objects than to traffic sign tracks. For the *cutouts* the relative workload is only three to one higher for non-sign objects since the tracks of traffic signs are twice as long as the ones belonging to non-sign objects in the mean.

6.6 Classifier

In the following the experiments and evaluations concerning the classifier are elaborated. The first step in the classification process is the normalization of the *cutouts*, meaning the extracted raster image regions, see section 2.6.1. Then the performance of the Stage I classifier and the Stage II classifiers, as explained in section 2.6.2, are presented. Then the combination of the classification results gained from the single *cutouts* to the result for a complete track are evaluated.

The following sections show the results on the efforts to *internationalize* the traffic sign recognition system. Parts of this system being the creation of *synthetic* classification samples as explained in section 4.2.2 and the combination of classifiers and their assignment to different country classifiers.

The final part of the experiments on the classification system is the presentation of the over all systems classification results.

6.6.1 Normalization

The normalization algorithm was optimized with the goal of producing images of a uniform size and brightness of the *cutouts*, thus simplifying the classification task. For the evaluation of the quality of the normalization process the first step was the readability for a human observer.

When the size of the output image is too small, even a human is no longer able to classify the depicted sign. Should the size be chosen too large, the computation time and requirements for the number of necessary training samples rise. The minimum size where humans were able to decipher the numbers on the speed limits was 15 pixels diameter, thus this was used as the minimum size reviewed. For signs with a diameter of 19 pixels the number of necessary parameters for the PCA is already 60% higher than for 15 pixels, thus this size was selected as maximum size to be reviewed. By the use of exemplary classifiers the final size was set to 17 pixels diameter since it produced a better recognition rate than the classifier for 15 pixel diameter while being close to equal to the 19 pixel diameter classifiers and less computational intensive.

The second part of the normalization process has to remove the differences in appearance introduced by different lighting conditions and has to remove eventual sensor artefacts, like those introduced by the *Bayer-Pattern*, as shown in section A.6. Since the classifier should be usable for different types of cameras and to allow the use of synthetic samples as introduced in section 4.2.2 for classifier training, the goal is to reach as uniform as possible an appearance of the normalized *cutout*, as independent as possible of the sensor or lighting.

The approach for optimizing the normalization is as follows. Using different types of cameras, e.g. high dynamic range CMOS, CCD, Webcam, mobile phone cameras images from different distances from a traffic sign, thus different sizes in the image, were taken under different lighting conditions. The traffic sign *cutouts* were extracted and different normalization algorithms and parameter settings were tried. To evaluate the effectiveness of the used algorithm the resulting normalized cutouts were correlated with the size normalized image of an ideal traffic sign pattern digitized from the rule book. The signs from the rule book are ideal patterns drawn in a vector graphics program using the font and relative sizes as described in the rule book and rechecking against scanned images from the book itself and refining the vector graphic if necessary. The higher the correlation coefficient of the normalized images of real world signs against the rastered vector, the better the normalization scheme was rated. The algorithm performing best is explained in section 2.6.1.

6.6.2 Single Pattern Classifier

The single pattern classifier consists, as described in section 2.6, of a pattern cluster classifier (stage I, see figure 2.13) and a pictogram classifier layer consisting

of six sub classifiers in the stage II. In both stages the classifiers are pairs of a principal component analyses and a following polynomial classifier, adapted in the mean square sense, of the second degree on the dimensionally reduced feature space. Thus for one complete set of classifiers seven principal component analyses and seven polynomial classifiers have to be trained.

Principal Component Analysis

The free parameter to be adapted in the principal component analysis is the number of output dimensions M . For the determination of the number of output dimensions in the principal component analysis, the number necessary for a mean reconstruction error $\overline{R}^2 \approx 0.1$ (10%) is selected as stated in section 4.1.1. The result for one of the stage II classifiers, see figure 2.13, is shown in figure 6.16. The *synthetic* signs, see section 4.2.2, are reconstructed less adequate than the *real cutouts* in the test sample set. The mean error for all speed limit classes for $M = 48$ is close to the target value of 10%.

Stage I Containing the Cluster Classifier

In figure 6.17 the rejection capabilities of the stage I classifier are presented. The stage I classifier has two main tasks. The first is to reject the bulk of the garbage patterns presented to the classifier. The second task is the distribution of the presented *cutout* samples to the specific stage II sub classifier. The six, see figure 2.13, different appearance types are called *cluster* of sign types. Due to this separation capability the stage I classifier is further on also called *Cluster Classifier*.

The rejection capabilities are shown in table 6.19. Of the 968 515 garbage patterns 150 110 and thus just 15.5% are not rejected and thus are passed to the stage II classifier as false positives. The distribution of the remaining garbage patterns on the stage II classifiers is shown in table 6.19. The table shows that the number of garbage signs accepted as *inner* part of a traffic sign, meaning the digit pattern of a speed limit without the red rim, is two to three times higher than the number of patterns accepted as end of limit or an *outer* cutout including the red ring. This behaviour is due to *end* and *outer* signs having the clear rejection features of the diagonal stripe or the broad outer ring respectively. The *inner cutouts* have just the digit patterns which are different for all the 15 subclasses to be separated by the *inner* classifiers, these being the digits 10 to 130 plus the two types of non overtaking signs.

Even after the rejection of roughly five sixths of the *garbage* patterns the rate of garbage patterns presented to the respective stage II classifier is high. Still one quarter of the patterns are *garbage* patterns. For active end of speed limits the rate is even 45%, due to the low a priori probability of those type of signs.

The traffic sign patterns are accepted with probability of 90%. There are

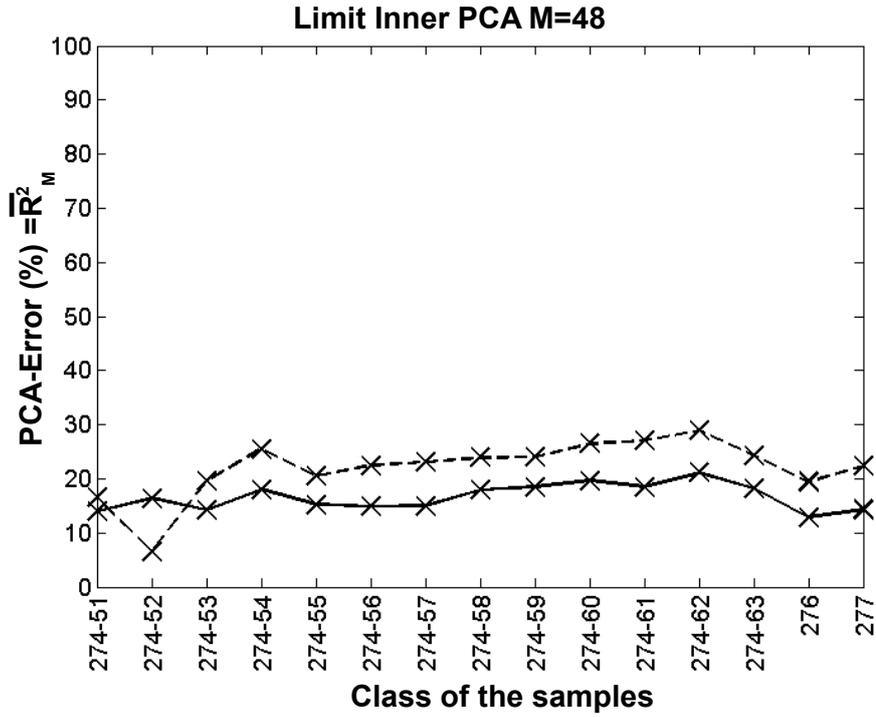


Figure 6.16: Exemplary mean reconstruction error \bar{R}_M^2 , with $M = 48$, as explained in equation 4.9 over a sample set. The test sample set in this example is the set for testing the pictogram classifier for the inner circles of passive speed limits in Germany, see table 6.4. The results for training the PCA with 1000 samples of *synthetic* signs are shown as dotted line, the curve for training the PCA with the *real cutouts* from table 6.4 is the solid line. The x-axis is labelled with the classes of the reconstructed samples, see HAV labels in section A.1. It can be seen, that the target of 10% reconstruction error \bar{R}_M^2 is nearly reached for the real signs and slightly worse reconstruction occurs when training with *synthetic* signs only.

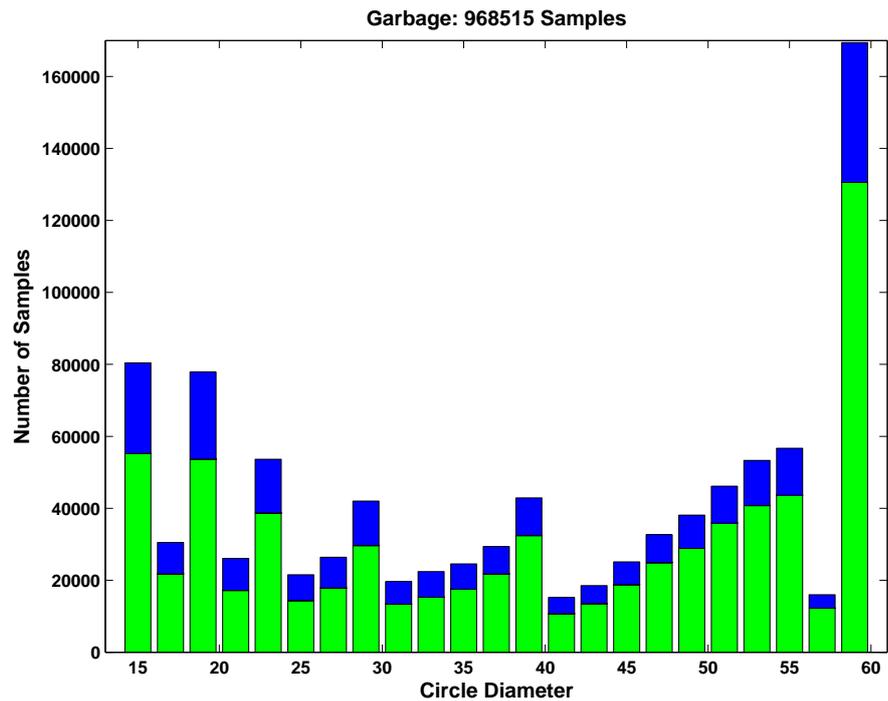


Figure 6.17: Rejection capability of the stage I classifier applied on *garbage* samples. Green shows the patterns classified as *garbage*, blue are the number of false positives handed over to the stage II. The high number of samples for the largest diameter of circles 59 derives from even larger circles, diameters up to 63 pixels, being detected and obtaining the largest detection diameter. The classifier is the PCA/PC combination explained in section 2.6.2, the training and testing set are given in table 6.3

extremely few, less than 0.01%, errors in the assignment between *active* and *passive* or *speed limit* or *end of limit*. Those errors occur only in cases of inaccurate detections of the signs centre position or occlusions on the signs surface. The differentiation between *inner* and *outer* speed limit *cutouts* is difficult to evaluate. The transition between a circle still being labelled as *inner* and a circle deemed to be *outer* is fluent and thus the decision for either *inner* or *outer* is not counted as an error at this point of the algorithm.

	Active			Passive			all
False Positive Res.	End	Inner	Outer	End	Inner	Outer	968515
Num. of elements	11382	30644	10793	16567	56785	23939	150110
Fpos/sign sample	0.442	0.344	0.180	0.328	0.232	0.126	0.227

Table 6.19: Table showing the distribution of the false positives, meaning garbage patterns being classified as sign candidates, on the stage II classifiers. The second row shows the relation of garbage patterns to sign patterns at the input of the stage II classifiers. The majority of the 968515 garbage patterns classified was classified as garbage or rejected, thus 818504 or 84.5%. These rejected patterns are not part of the above table. The classifier used is the PCA/PC combination explained in section 2.6.2, the training and testing set are given in table 6.3

Stage II Containing the Pictogram Classifier

In this stage of the single pattern classifier, the decision for the valid speed limit or end of speed limit is realized. The stage II classifier is further on also called *pictogram classifier*. This stage consists of six classifiers, each consisting of a coupled pair of a principal component analysis and a polynomial classifier. The six types are the three types *end of limit*, *inner* and *outer* speed limit for both *active* and *passive* signs, see figure 1.2. Inner and outer refer to the to the two concentric circles of a speed limits rim. The classification results for the German classifier set applied on the German sign patterns are shown in table 6.20 and depicted in the figures 6.18 and 6.19 in different views of the results.

The table 6.20 shows that the precision, meaning the probability of not rejected classification results being correct, is higher than 90% for all six subclasses, see figure 2.13. The number of traffic sign *cutouts* being classified correctly is higher than 85% for all subclasses. The *outer cutouts* of active signs show comparatively high rejection rates, especially in small *cutouts*, see figure 6.19(c). This is due to the blurring of the digits in the image and thus the lessened discriminability of the different sign classes. The high rejection rate for passive end of limit signs derives from the comparatively high a priori probability of a pattern presented

to this classifier being a *garbage cutout*. Since a typical traffic sign track consists of about 20 *cutouts* as shown in section 6.18 this classification result for single patterns allows for an even better classification rate when the whole track is used as decision base for the type of traffic sign encountered as is shown in the following section 6.6.3.

	Active			Passive			all
	End	Inner	Outer	End	Inner	Outer	
Correctly class.	13701	54884	43561	31043	183361	157005	483555
Rejected signs	574	2787	3555	1662	2206	4938	15722
Misclassified	122	771	2132	1286	2935	4714	11960
False Positives	668	2286	992	1887	8133	3625	17591
Correctly class.	0.909	0.904	0.867	0.865	0.932	0.922	0.914
Rejected signs	0.038	0.046	0.071	0.046	0.011	0.029	0.030
Misclassified	0.008	0.013	0.042	0.036	0.015	0.028	0.023
False Positive	0.044	0.038	0.020	0.053	0.041	0.021	0.033
Precision	0.945	0.947	0.933	0.907	0.943	0.950	0.942

Table 6.20: Result table of the German stage II classifiers. First part the sample numbers, second part the relative numbers. The column *all* shows the accumulated result for all classes. The third part shows the precision of the results computed by dividing the number of correct classifications by the number of samples not being rejected, thus the reliability of the result. The classifier used is the PCA/PC combination explained in section 2.6.2, the training and testing set are given in table 6.4

6.6.3 Track Classifier

For the classification of a real world traffic sign the results of the single pattern classifier belonging to the track have to be combined to a final decision. For this objective the results of the single pattern classifier are added to a weighted sum. The weights are chosen according to the precision, thus the reliability, of the classification result in relation to the signs diameter in the respective *cutout*. These precisions or reliabilities can be derived from the values in the six sub figures in 6.19 by dividing the number of correctly classified samples by the sum of all samples not being rejected. The results for *cutouts* being rejected via the RAD criterion, see equation 4.17, are not accumulated in the weighted sum, since they are not trustworthy. The polynomial output values are truncated to the values between 0 and 1. After the cropping the values are weighted with

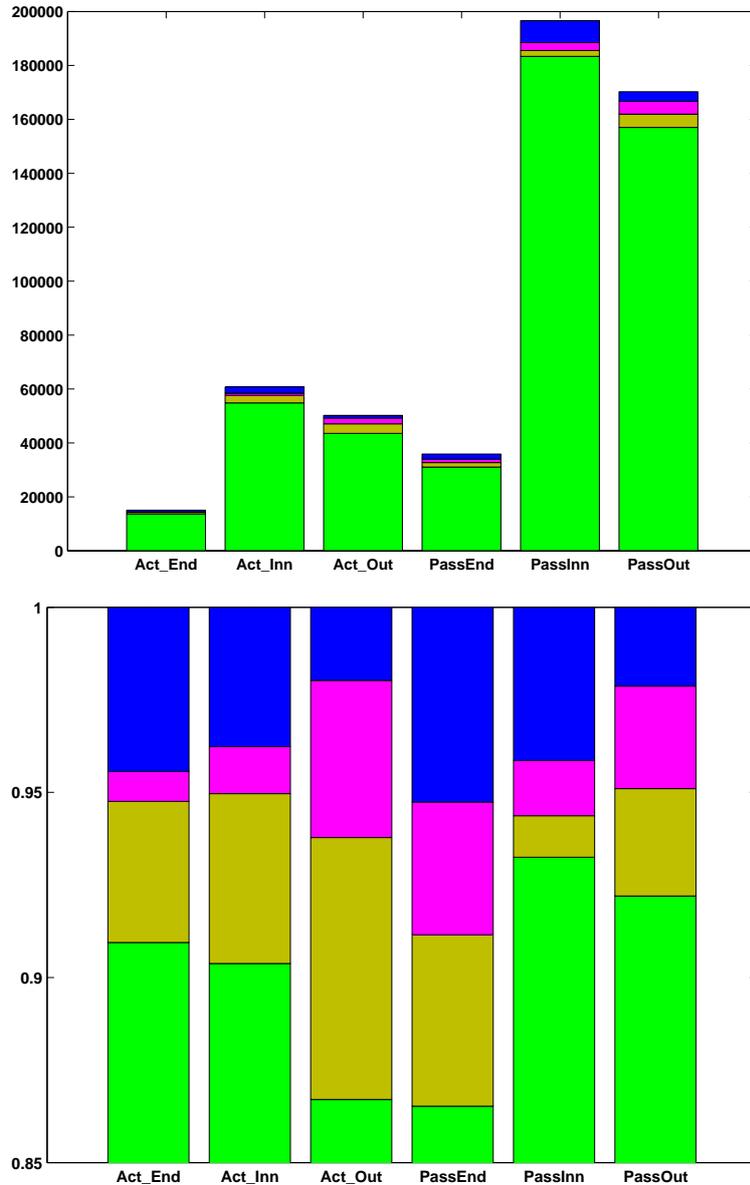


Figure 6.18: Figures showing the performance of the German stage II classifiers. The bright green on the bottom of the columns denoting the correctly classified samples, dark green, second from the bottom, the rejected samples, the magenta showing the misclassifications (confusion between two classes) and blue on top the false positives. On top showing the absolute numbers, the lower graph the relative percentages. The seemingly different percentages derive from the lower graphic starting at 85% level for visibility purposes. The classifier used is the PCA/PC combination explained in section 2.6.2, the training and testing set are given in table 6.4

6 Experimental Investigations and Evaluation

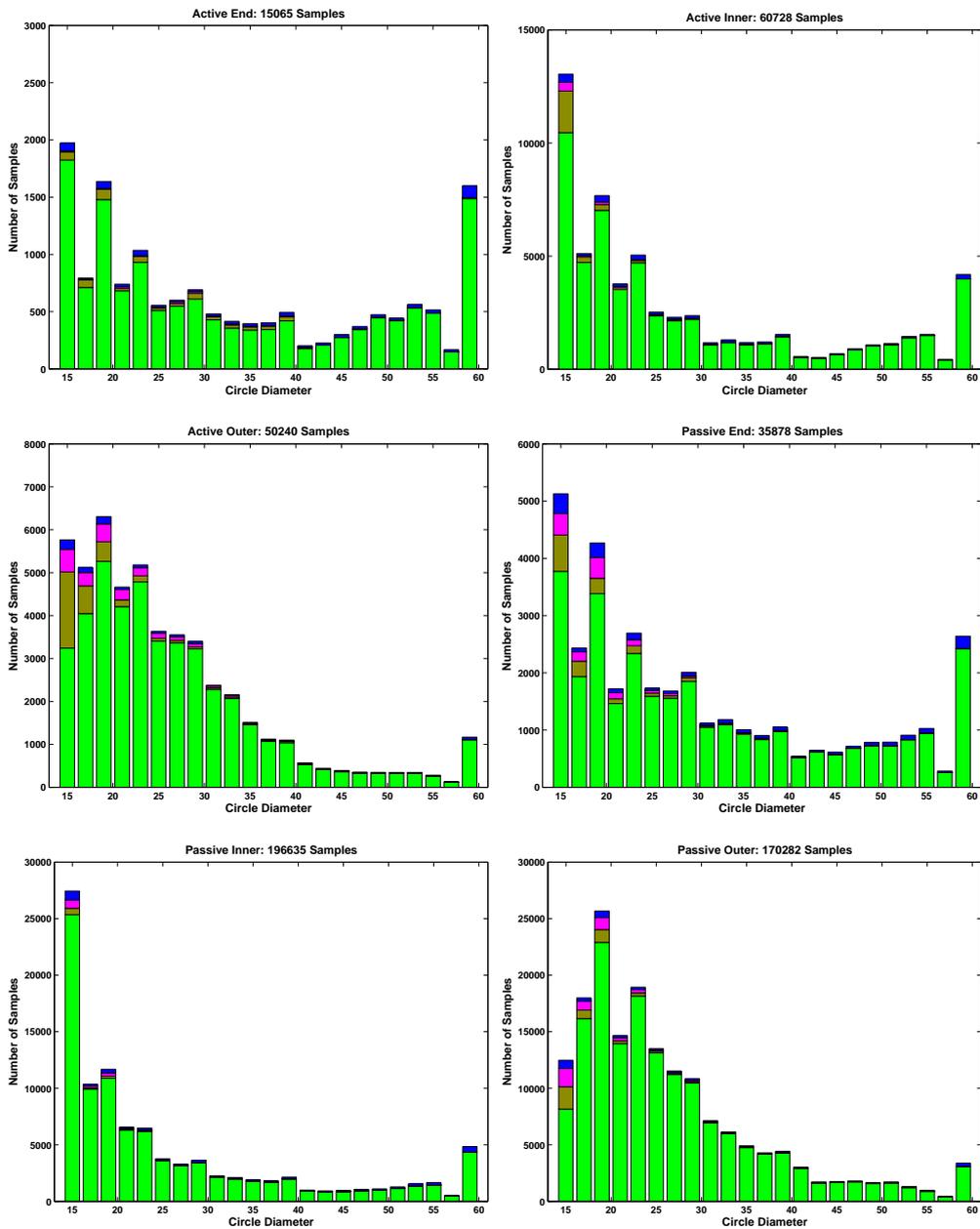


Figure 6.19: Figures showing the performance of the German stage II classifiers. The bright green on the bottom of the columns denoting the correctly classified samples, dark green, second from the bottom, the rejected samples, the magenta showing the misclassifications (confusion between two classes) and blue on top the false positives. The classifier used is the PCA/PC combination explained in section 2.6.2, the training and testing set are given in table 6.4

the determined precision value according to the size of the *cutout* and then the values added to a class wise result histogram. The results of *inner* and *outer* speed limits are accumulated in the same class slots in the histogram. When the track is completed the histogram with one entry per traffic sign class is analysed. The entry with the highest resulting value is selected as the winning class. For this result to be accepted the value in the histogram has to exceed an absolute and a relative threshold. The relative value is computed by dividing the histogram value for the winning class by the sum of all histogram entries, see equation 6.15.

h : Histogram of the accumulated classifier output neurons for all classes.
 v_{abs}, v_{rel} : Result values to be matched against a decision threshold.

$$\begin{aligned} v_{abs} &= h_{\text{winner}} \\ v_{rel} &= v_{abs} / \sum_1^{\text{NumClasses}} h_i \end{aligned} \quad \text{accept if } \begin{cases} v_{abs} > \rho \\ v_{rel} > \vartheta \end{cases} \quad (6.15)$$

The criterion validating the minimum absolute histogram value is used to reject results from very short tracks, since without this threshold a single accepted result for a *cutout* could result in a positive track result. Since the precision of the classifiers lies between 90% and 95%, see table 6.20, this would unduly increase the number of false positive track results. The thresholds ρ and ϑ were adapted using the above sample set.

Absolute track classifier threshold	ρ	1.10
Relative track classifier threshold	ϑ	0.70

Table 6.21: Thresholds for the track classifier.

6.6.4 Summary of the Classifier Evaluation

The classifier for single *cutouts* as described in section 2.6 consists of a hierarchical two stage approach each stage consisting of principal component analysis / polynomial classifier pairs on normalized grey value pixel features. The results as shown in table 6.20 show the high classification performance of the algorithm. The precision being over 95% for most classes and the recall being above 90%. The usually high number of single *cutout* elements in one track belonging to a real world traffic sign lead to even better classification results.

6.7 Classifier Internationalization

One of the main purposes of this thesis is the development and demonstration of ways to adapt a complex classification system to the variations encountered when the system is used in differing environments from the one it was first developed

in. In the case of the traffic sign recognition system this means generalizing the classification system from a purely German traffic sign classification system to a system operating in a variety of countries. The algorithms used to approach this challenge are explained in section 4.2. The results of the experiments performed on these algorithms are presented in the following sections. The first section 6.7.1 shows the advanced generation of samples for countries for which no or insufficient training sets exist for training a classifier. The next section 6.7.2 shows the results of the attempt to automatically label samples by observing the reaction of the driver on the traffic signs. The third section concerned with the *internationalization* task 6.7.3 shows the results of the algorithm determining the necessary number and composition of classifiers to cover the classification task in ten European countries with adequate performance. The fourth section 6.7.4 shows the resulting performance of the traffic sign classifier in several European countries.

6.7.1 Synthetic Generated Sign Samples

The introduction of *synthetic signs* is necessary to allow the start of a bootstrapping process for classifiers for a country, class or appearance type of signs, where no sufficient number of samples have been recorded. The process of creating a sample set consisting of *synthetic signs* begins with the learning of the transformation parameters to be used to transform an ideal sign candidate into realistically modulated *cutouts*. These transformation parameters have to be learned only once, since they represent the transformations undergone by the images of real world signs when recorded by the camera system from a moving vehicle. The distributions of parameters are stored as histograms of quantized transformation parameters, where the magnitudes of the histogram entries indicate the probability for the respective histogram slot. When the distribution of transformation parameters is learned, large numbers of *synthetic signs* can be produced by drawing parameter sets from the probability distributions learned and modifying the ideal sample *cutout* accordingly. The ideal sample images are drawn in a vector graphics program using, if possible, the font and relative sizes as described in the rule book and rechecking against scanned images from the book itself or pictures taken of the signs and refining the vector graphic if necessary. The algorithm was presented in [Hoessler et al., 2007].

In this section the validation of the parameter learning and the results of the transformation parameter learning is explained. In addition the superiority of the presented algorithm compared to the conventional random choice of transformation parameters is shown. The transformations consist of a geometric part, the rotation of the pattern in the image plane, the translation horizontally and vertically and the scaling horizontally and vertically. As well as the transformations consist of a photometric parameter part, the brightness representing the mean grey value and the gain representing the contrast. The parameters are explained

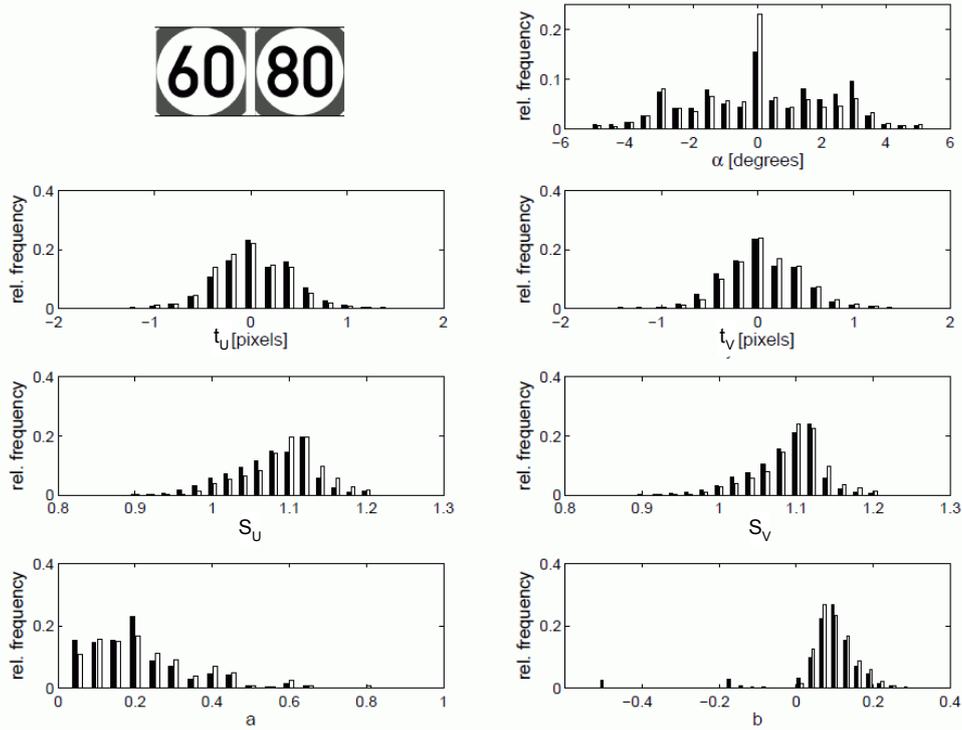


Figure 6.20: Distributions of the transformation parameters for the synthetic sign types 274-56 (black) and 274-58 (white). As shown by a Kolmogorov Smirnov test both distributions are similar. This shows the correctness of the assumption of the generalizability of the transformation for different sign classes and thus allows the use of the parameter sets for the creation of sample sets for newly encountered sign types and appearance types of traffic signs. The transformation parameters are α the centred rotation of the image in the image plane. t_u, t_v are the translations in u and v direction, s_u, s_v , are the scaling values, a is the mean brightness and b the gain, see section 4.2.2.

at length in section 4.2.2.

In figure 6.20 the trained distributions of transformation parameters for the signs *speed limit 60 (274-56)* and *speed limit 80 (274-58)* are shown. The distributions are derived by learning the transformations of 5489 samples of class 274-56 and 4277 samples of type 274-58 taken from the stage II pictogram classifiers training set. The alikeness of the distributions for the two classes has been tested with a Kolmogorov Smirnov test. The test revealed the distributions to be similar with a probability of 60% to 84%. This shows that the distribution of the transformation parameters is independent from the class of the sign and thus signs of other classes and appearance types can be created using the distributions of parameters gained by computing the parameters for just one class and appearance type. Using the Kolmogorov Smirnov test it can also be shown with a probability of over 90% that the distributions are not Gaussian.

Using correlation tests on the parameter histograms plotted in figure 6.20 yields the result that the two scaling parameters are interdependent as well as the parameters for brightness and gain. The other parameters yield correlation values of less than 0.01 and are thus considered independent. These dependencies have to be regarded when randomly drawing a parameter set from the learned distributions, see [Hoessler et al., 2007].

As a test proving the advantages of the new algorithm for creating *synthetic* samples three classifiers of stage II in the traffic sign classifier hierarchy are trained twice. The *ideal* samples are of the type most frequently used in Germany. The test set is German as well.

For the first set of three classifiers 1000 samples were created from one *ideal* sample per class, drawing the transformation parameters from a uniform distribution between the minimum and the maximum values of the distributions learned by the use of the new algorithm. The correlations between the scaling factors and the brightness and gain parameters was respected as well. The classification results are shown in table 6.22.

For the training of the second set of three classifiers 1000 samples per class are created using the proposed algorithm. The result is displayed in table 6.23.

The classifiers trained with the samples created using the new algorithm show a much better performance. The rejection rate for both sets of classifiers is comparatively high. This is due to some *older* types of German signs not being covered by the use of just one *ideal* sample. For a better performance of the classifiers further templates for *ideal* samples should be added to allow for different appearance types of German traffic signs. For the comparison of the two types of sample generation this first step is sufficient. The classifiers trained with the benefit of the use of the new algorithm especially exceed the others in the numbers of false positives occurring and thus have a much higher precision, thus proving the superiority of the newly developed scheme.

Univariate Distr.	Correct	Rejected	Misclass.	False Pos.	Precision
Passive End	17292	14273	2426	2405	
Passive Inner	136933	46456	82997	3499	
Passive Outer	80130	82997	3530	2008	
Passive End	50.9%	42.0%	7.1%	15.7%	78.2%
Passive Inner	72.6%	24.6%	2.7%	10.3%	94.1%
Passive Outer	48.1%	49.8%	2.1%	9.8%	93.5%

Table 6.22: Table showing the results for three stage II polynomial classifiers, as explained in section 2.6, trained with samples generated from a thousand samples per class, generated from one ideal sample per class and transformed using the equations 4.18, 4.20 and parameters drawn from an uniform distribution between the minimum and maximum value of the distributions in figure 6.20 tested on *real* German samples. Table 6.4 shows the composition of the testing set.

6.7.2 Sample collection by use of the driver behaviour

This section shows the results of the experiments performed on the sample generation from driver behaviour. The idea behind this scheme is to use an imperfect classifier and the driver behaviour to generate preliminary labels for sign detected and classified with a high uncertainty. For example if the classification system detects a traffic sign, verifies the existence of the *sign like* real world object via three dimensional measuring size and position and the stage I classifier accepts the tracked patterns as possibly being traffic signs, this object is more likely than not a traffic sign. This is true, even if the stage II classifier is not able to discern the exact type of sign encountered. Currently the images showing these objects are recorded and the tracks of the objects in the image are presented to a human labeller to decide to which class the track of samples belongs. The new algorithm supports the labeller or even labels the samples autonomously by evaluating the behaviour of the driver while passing the sign, see section 4.2.3.

The figures in 6.21 show the allowed maximum speed over the velocity of the vehicle. The upper figure shows a swift driver in dense traffic, while the lower figure shows a moderately travelling driver in light traffic. The swift driver moves long parts of the route uninfluenced by the speed limits, sometimes the driver exceeds the allowed maximum speed by up to 80 km/h. The moderate driver abides to the speed limits most of the time and, if not, exceeds the limit by no more than 20 km/h. The second information necessary for autonomously labelling a traffic sign is the advance or delay time or distance with which the driver reacts to a change in the allowed maximum speed. This number is highly variable. The reaction occurs from 50 meters before the sign to up to 500 meters after the limiting sign. This is true for both types of drivers. Especially a very

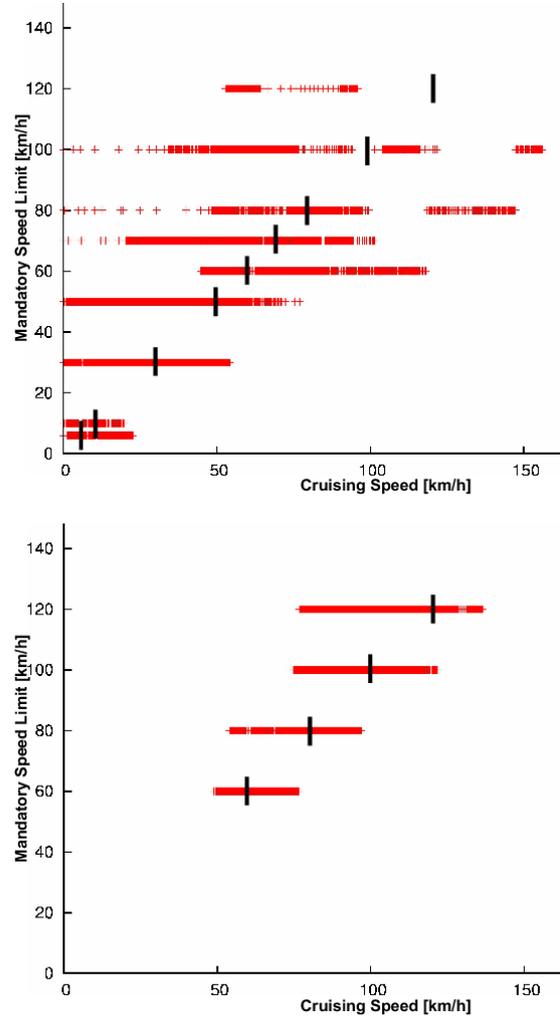


Figure 6.21: Both figures above show the allowed maximum speed over the velocity of the vehicle. The upper figure shows a sequence driven in partly heavy traffic by a speeding driver, who drives 40% of the time faster than the allowed speed limit. The lower figure shows the velocity of a driver keeping the speed limits in over 90% of the time and not speeding over 20 km/h over the allowed limit at any time. The data is sampled once a second, the higher the density of crosses, the more often the car was driving this speed. The black bar marks the mandatory speed.

Proposed Algo.	Correct	Rejected	Misclass.	False Pos.	Precision
Passive End	17890	14878	1223	683	
Passive Inner	149762	36162	2578	1508	
Passive Outer	102865	60102	3690	1009	
Passive End	52.6%	43.8%	3.6%	4.5%	90.4%
Passive Inner	79.4%	19.2%	1.4%	4.4%	97.3%
Passive Outer	61.7%	36.1%	2.2%	4.9%	95.6%

Table 6.23: Table showing the results for three stage II polynomial classifiers, as explained in section 2.6, trained with samples generated from a thousand samples per class, generated from one ideal sample per class and transformed using the equations 4.18, 4.20 and parameters drawn from the distributions shown in figure 6.20 tested on *real* German samples. The number of false positives being more than halved and the precision being enhanced compared to the classifier trained using samples generated without the use of the new algorithm, see table 6.22. Table 6.4 shows the composition of the testing set, the same set was used from the data in table 6.22.

delayed reaction makes the assignment of the reaction to a sign difficult, since sometimes the reaction takes place after the encounter of the next sign showing the same or a different limit.

The figures show that the type of driver has a strong influence on his behaviour when a speed limit is encountered. The second factor is the traffic density. In dense traffic the driver most of the time is reacting on the behaviour of other traffic participants and not on speed limits.

The samples extracted by the use of this scheme were only 10% correct, even when respecting the mean reaction time and mean speeding offset of the respective driver. The unknown traffic density and high error rate made the yield of this scheme too meagre to be followed further without additional sensors or algorithms giving additional information on the driving situation like leading vehicles or weather conditions influencing the driver.

6.7.3 Management of Multiple Sub-Classifiers for Internationalization

The results presented in this section were published in the diploma thesis of Mr Denis Koch [Koch, 2007]. As explained in section 4.4 instead of adapting one classifier per country in which the system should be operable, the training sets of countries having signs with similar appearances are joined and the generated classifiers be used in all of these countries.

The evaluation was made for *ten European countries*, see A.1. The number of countries was chosen for being large enough for a meaningful evaluation while

being manageable from the number of samples to be accumulated and labelled. For the creation of the classifiers for the *ten European countries*, the first step is to determine the classifiers for which countries can be combined and what the loss of classification capabilities is. For ten countries one could argue to keep ten classifier sets for maximum classification capabilities, When expanding the number of countries, for instance to the 27 states in the European Union or the 46 European states, this leads to a very high number of classifiers and memory use. Thus a balance between reduction of the number of classifiers and loss of classifier performance has to be found.

To find the best classifier setup, all possible combinations of samples from different countries were tried for training classifier sets. The evaluation results were compared, see figures 6.22. There are only nine figures since there were only few samples recorded in Luxembourg and those are very similar to signs from France. Thus the combination of French and was predefined. The combination of samples from countries having similar appearances leads to better classification results than when samples from countries with very different appearances were mixed. For the example of Belgian samples the classification results when mixing in the samples from all countries in the training process yields better results than mixing just two very different ones, here Belgian and Swiss. The solution of having one dedicated classifier set per country performs best, as was to be expected. When adding samples from other countries to the training set the error rate rises, depending on the *compatibility* of the two countries sets.

As an example the curve where the German test set (DE) was used is examined. When using samples from three different countries the error rate, meaning 1 – precision can be as low as 1% or as high a 2.5% for a unfavourable choice of countries used for samples in the training set. In the nine figures by means of the choice of the acceptable error rate the minimum number of necessary classifiers can be determined.

In figure 6.23 the result for passive limits is displayed. There are two possible proceedings. Either the maximum error is given and the minimum number of necessary classifiers in the set is determined, or the maximum number of classifier sets is given and the best possible error rate is searched for. For example at a maximum allowed error rate of 2.5% per country three classifiers are necessary. The combination of samples in the training sets is as follows: BE,DK,FR,IT / AT,CH,DE / NL,ES. Should we accept an error rate of 4% per country two classifiers suffice and are composed of samples from BE,DK,ES,FR,NL and AT,CH,DE,IT respectively.

Reviewing the over all result of the system given the precision values reachable with the different combinations of classifiers the classifier sets were composed as shown in table 6.24. The *active* signs are very similar in the different countries, so only one classifier for the whole of the ten countries was created. For the *passive* signs the combinations shown in the table 6.24 are used. The precision for the selected combinations for *passive limits* shows a slightly higher error rate

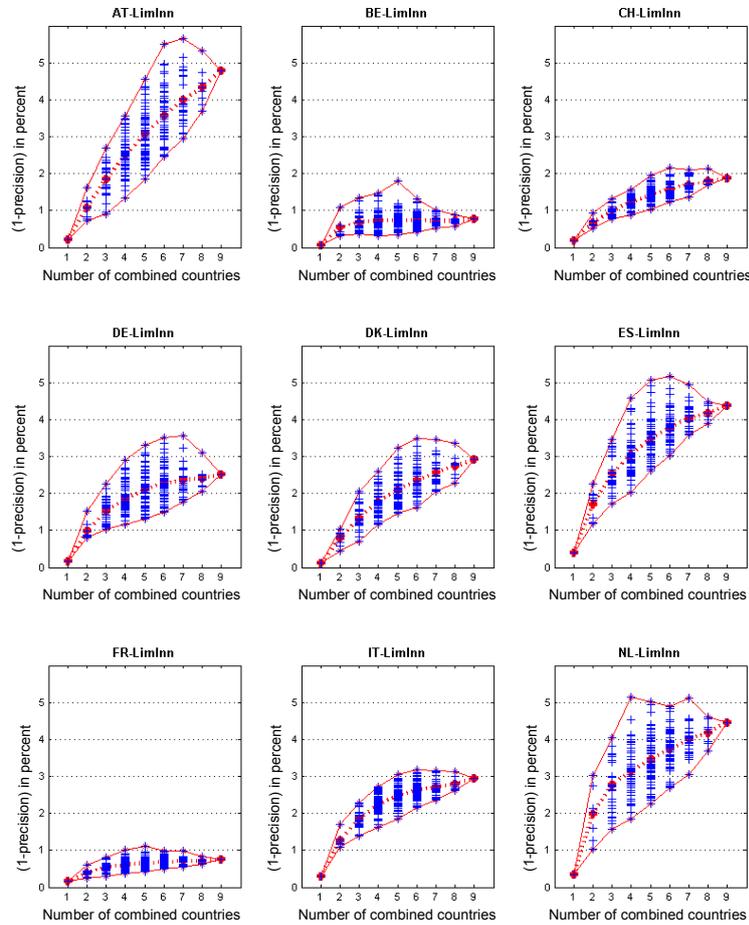


Figure 6.22: The nine figures show the error rate for samples from a given country when the classifier is generated from the samples from N different countries. On top of each graph is the country of origin of the test set. The ordinate of each graph shows the error rate, $1 - \text{precision}$. The abscissa shows the number of country sample sets that were used in the training of the classifiers. The classification results for the different training set combinations are shown as blue crosses and the enveloping curve showing the best and worst combinations in red. The lowest error rate is measured when the training set consists of samples of the country only of which the test set is used, thus number of countries 1, as was to be expected. The classification algorithm is the PCA/PC pair explained in section 2.6.2, the training and testing set is shown in section A.8. For further explanations see section 6.7.3.

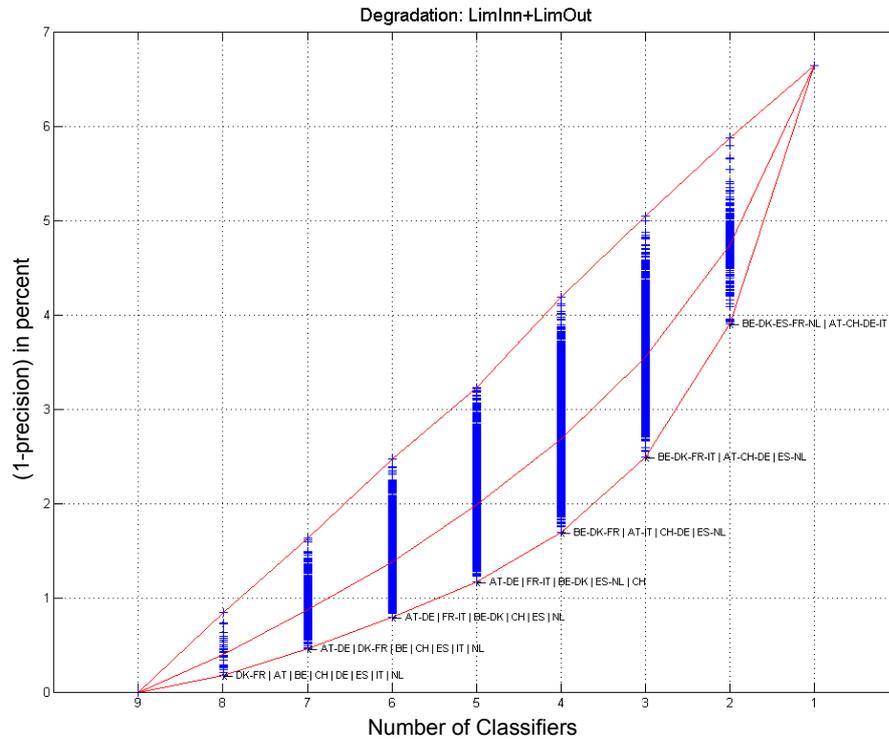


Figure 6.23: The figure shows the mean error rate, $1 - \text{precision}$, over all samples from all countries given a maximum number of classifiers used, see section 2.6.2. The classification algorithm is the PCA/PC pair explained in section 2.6.2, the training and testing set is shown in section A.8. For further explanations see section 6.7.3. The classification results for different combinations of country sample sets used in the generation of the classifiers are shown as blue crosses. The enveloping curve shows the best and worst combinations in red. By using this figure the number of classifiers necessary to stay below a maximum error can be determined, as well as the minimum error rate reachable for a given number of classifiers. The countries written at the best performance line show the combinations necessary to reach this lowest error rate.

than the best combination for maximum three classifiers, but the rejection rate is considerably lower, thus the *recall* is higher for the selected combination of countries. The *passive end of limits* have the particularity, that the Italian signs have a nearly vertical stripe instead of the approximately 45° stripe in the other nine European countries regarded, thus the classifier for Italian end of limit signs was trained with samples from Italy alone, see figure 1.7.

Stage II Classifier	Classifier I	Class. II	Class. III
Act. End/Inn./Out.	All		
Passive End	AT BE DK FR(LU) ES NL	IT	CH DE
Passive Inner	AT FR(LU) ES IT NL	BE DK	CH DE
Passive Outer	AT FR(LU) ES IT NL	BE DK	CH DE

Table 6.24: Table showing the final composition for the stage II classifiers, see figure 2.13. Here three classifiers are trained for the ten European countries considered. The samples for training were taken from the corresponding countries sample sets.

6.7.4 Results of the complete classification system

This section presents the results of the classification process on the base of traffic sign tracks for the ten selected European countries on the base of the classifiers trained with the sample combinations presented in table 6.24. The results are shown in table 6.25. The precision of the classifier is well above 90% for all countries in the set. The recall, with the exception of Luxembourg, is 85% or higher. The signs in Luxembourg are a very special case, since in the already small country only very few samples could be gathered for a meaningful sample set. In addition only about 80% of the signs in Luxembourg are of the *French* type. Close to the borders of Belgium or Germany sign appearances resemble the ones of the respective country so that the switching of classifier to the one used in France fails for these signs.

Many of the errors of the classifier derive from systematic errors. The most prominent of these errors is called *on Vehicle* and describes the maximum speed allowances of lorries that are mounted on the back of the vehicles. These signs are comparatively small and thus often not detected by the system, but they look exactly like traffic signs and are thus, when detected, often classified as false positive traffic signs.

To reduce the impact of these systematic errors the three dimensional positioning system is used. As shown in section 6.4.3 the characteristics of signs mounted on moving vehicles can be separated from signs mounted at the side of the road. This can be done via rejecting signs if the measured real world size is higher or lower than the sizes of signs installed in the given country according to the rule books. Another feature allowing the rejection of signs mounted on a lorry is the value of Δd , as explained in section 6.4.3. This residuum value is close to zero if the algorithm succeeded. If this value is high this means that the condition that the object tracked, here the sign, was not moving is broken. This is the case for signs *on Vehicle*, thus a high Δd is used as a rejection criterion.

Some systematic errors are country specific. One example is the class of minimum speed sign and the exit signs in Germany as explained in section 2.8.2. These signs look exactly like speed limits with only the colours differing. These signs have a white rim, blue inlay and white digits. In the grey value image used these signs are very hard to separate from the *active* signs, especially after the brightness normalization explained in section 2.6.1, as shown in figure 2.18. To reduce the number of errors introduced by these signs an additional algorithm is run on the un-normalized *cutouts* of tracks classified as *active* speed limits. The results of the algorithm are explained in section 6.9.

By the use of the three dimensional positioning system for the rejection of signs being *n vehicle* the number of false positives could be reduced by about 82%. In Germany the additional use of the rejection algorithm for minimum speed and exit signs together reduced the number of false positives by 88%. Of course the additional rejection steps themselves have false positives, thus rejecting some

	CC	FC	MC	FP	Rej.	Miss	CC %	FP %	Prc.%	Rec.%
AT	6159	35	73	377	632	100	97.73	5.98	92.70	88.00
BE	1889	6	25	116	218	12	98.08	6.02	92.78	87.86
CH	2910	6	23	100	379	20	98.81	3.40	95.76	87.18
DE	69408	537	422	4971	4948	1376	97.89	7.01	92.13	90.50
DK	1125	3	8	32	62	0	98.77	2.81	96.32	93.91
ES	10655	79	170	320	1228	135	97.01	2.91	94.93	86.86
FR	6897	39	67	511	505	110	97.94	7.26	91.79	90.54
IT	10847	197	98	689	1041	251	95.66	6.08	91.68	87.24
LU	562	0	3	12	133	13	99.47	2.12	97.40	79.04
NL	3926	21	52	156	486	53	97.66	3.88	94.49	86.51
All	114378	923	941	7284	9632	2070	97.62	6.22	92.59	89.40

Table 6.25: Table showing the results of the detection and classification system based on single real world signs. The column labels are as follows: CC: correctly classified / FC: false category in stage I, mostly active signs rejected as *directional arrow* / MC: misclassified, meaning wrong speed / FP: false positive / Rej: detected, but rejected sign / Miss: sign not detected / CC%: $CC/(CC+FC+MC)$ / FP%: $FP/(CC+FC+MC)$ / Prc: Precision $CC/(CC+FC+MC+FP)$ / Rec.: Recall $CC/(CC+RC+MC+Rej+Miss)$. The classifiers used were trained with samples assembled as shown in table 6.24, thus only three classifiers per passive pictogram type (end, inner, outer) were used and one per active type. The samples themselves were composed as given in table 6.5. The classification scheme is the hierarchical PCA/PC algorithm explained in chapter 2.6

valid signs as well. This leads to the number of correctly classified signs to be sinking by 0.7%, 0.8% in Germany, and the number of rejected signs rising by the same amount of signs. The results are displayed in table 6.26.

6.7.5 Summary of the Internationalized Classification System

The results in table 6.26 show that a very high precision of well over 97% is reached for the over all detection and classification system, thus the number of misclassifications or false positives is very low. On the other hand the recall is comparatively lower at about 83%. This number is explained by the typical placement on both sides of the street on higher order roads like highways. Thus one of the signs always has a high lateral offset to the camera, leaves the cameras field of view at a high longitudinal distance and therefore while still being small in the image. For the understanding of the scene the correct detection and classification of the other sign, being passed at a smaller lateral offset, suffices, thus the lower value in recall is acceptable.

	Correct Classified %	False Positives %	Precision %	Recall %
All	97.53 %	1.09 %	97.26 %	83.30 %
Germany	97.83 %	0.81 %	97.80 %	83.07 %

Table 6.26: Table showing the results of the detection and classification system based on single real world signs, after rejecting signs using the results of the three dimensional position and the exit sign suppression module. The columns are labelled according the the columns in table 6.25, training sets, testing sets and composition are the same as in this previous table.

6.8 Supplementary Sign Detection and Recognition

In this section the experiments and evaluations done on behalf of the detection and classification of supplementary signs as explained in section 2.7.1, are presented. The algorithm consists of three parts, first the detector, second the verification, third the classification.

The following experiments concerning the supplementary sign detection and classification are presented:

- Determination of the cluster positions in the detection step.
- Verification of the detection and ranking of multiple possible *cutout* candidates based on histogram of oriented gradient (HoG) features.
- Classification of the verified candidates for supplementary signs and combination of the results for single *cutouts* to the final classification decision for all detected supplementary signs of a traffic sign.

As training set for the determination of the necessary parameter settings of the algorithms a set consisting of 2441 German supplementary signs is used.

For the clustering of the sign positions the first step is to determine which distance measure to use. The parameters to be clustered are the top left and bottom right corner of the supplementary sign relative to the speed limits sign centre position and normed by the radius of the speed limit sign. The following metrics were considered, as explained in section 2.7.1: Euklidean, Chebyshev, Manhattan City Block. For checking which metric to use, classifiers were trained and tested. The samples for training and testing were created from the *correct* samples by varying the top left and bottom right corner coordinates for the creation of the sample *cutouts*. The metric correlating best with the classification results is the Chebyshev metric. The errors of the classifiers rise with the variations introduced by the translations. The classifier performs without visible losses up to box differing 0.1 in Chebyshev norm to ideal box, relative to the size of the corresponding speed limit. Minor losses occur when the corner coordinates are

changed up to 0.15 times the radius. Above this value the error rises, see table 6.27.

Translation values:	0.00	0.05	0.10	0.15	0.20	0.25
Correctly classified	85.6%	85.4%	84.6%	82.5%	75.4%	65.3%
Misclassified	3.7%	3.7%	4.1%	4.3%	5.9%	8.4%
Rejected	10.7%	10.9%	11.3%	13.2%	18.7%	26.3%

Table 6.27: Classification results for the supplementary sign pictogram classifier on German samples given samples translated in position in learn and test set, the sets being given in table 6.6 and the PCA/PC combination described in section 2.7 being used as classifier. The translation is given relative to the radius of the corresponding speed limit sign and is applied to the upper left and/or lower right corner coordinates $(\Delta u_0, \Delta v_0, \Delta u_1, \Delta v_1)$. Thus the position and scaling of the box is affected by the variations. The translation value corresponds with the Chebyshev distance of the original box to the translated box normed by the speed limits radius. It can be seen, that translations up to 0.15 result in minor reductions in classifier performance, only. Thus a positioning error of 0.15 is acceptable. This value is used for determining the parameters for the position clustering process.

For the choice of one of the clustering algorithms introduced in section 2.7.1 the number of clusters necessary is to be optimized. The number is defined as the one necessary to reach the targeted 90% of the samples being closer than the 0.15 radii from a cluster centre position in Chebyshev metric, as introduced above and in table 6.27. The agglomerative scheme performs slightly better than the divisive algorithm in this respect. The agglomerative algorithm needs 16 cluster centres to reach this target, while the divisive algorithm used 18 clusters for the same result. In the following the agglomerative algorithm is used.

In figure 6.24 the results of the clustering process are shown. The colours of the elements in the top left corner graphic correspond to the different classes of supplementary signs, see figure A.10, the black circles are the cluster centres. The bi-modality of the distribution is due to the two types of supplementary signs, half height and full height. Certain classes of supplementary signs are present in certain groups of clusters, only. This leads to a rule introduced in the supplementary classifier which rejects classification results for signs where this type of sign does not fit to the detected cluster.

After the candidate boxes are determined via the clustering process those box regions are passed to a verification and ranking classifier for each detected and positively classified speed limit sign. The verification classifier is based on histogram of oriented gradient (HoG) features. Due to the high variability of the pictograms shown on supplementary signs the regions where the HoG features

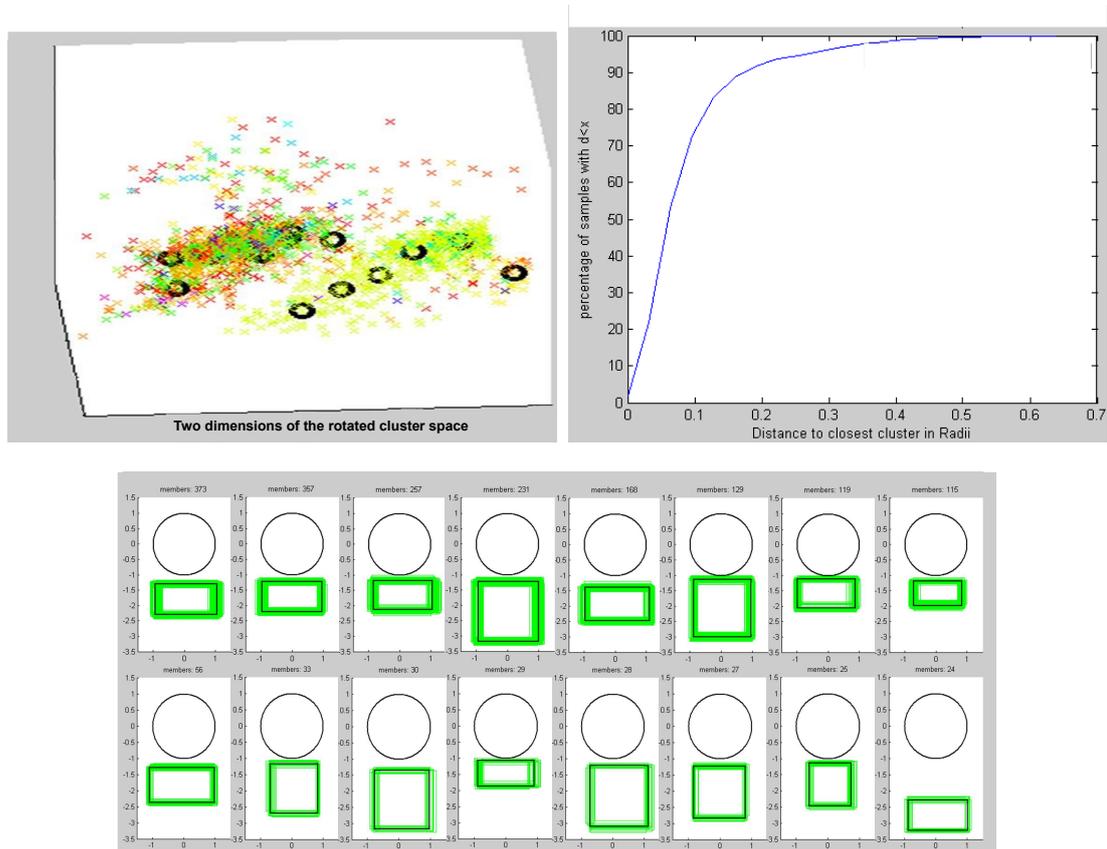


Figure 6.24: Diagrams showing the generation of the position clusters for the detection of supplementary signs.

Top left: two dimensional rotated (PCA) view of the four dimensional feature space ($\Delta u_0, \Delta v_0, \Delta u_1, \Delta v_1$) for clustering. Top right: Number of samples closer to a cluster centre than a certain distance, the example is for 16 clusters, agglomerative clustering. Bottom: Cluster centres of possible supplementary sign positions sorted by number of elements in the cluster. The sample set is taken from German supplementary sign positions only and is given in table 6.6.

are extracted were pre defined as depicted in figure 6.25 and thus concentrate on the border edges of the expected supplementary sign as well as the centre region to use the in-existence of the vertical edges belonging to the pole on which the speed limit is mounted. The classifier is a complete polynomial classifier of second degree. The features are the number of vertical and horizontal edges in the five feature boxes, see figure 6.25.

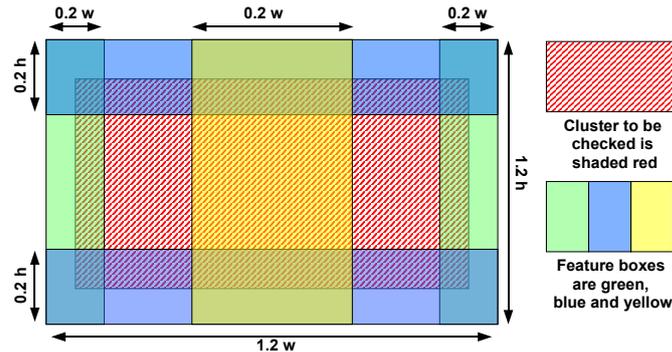


Figure 6.25: Regions of feature extraction for the HoG-Polynomial verification and ranking classifier for supplementary signs.

The classification result of the ranking classifier is shown in the confusion matrix in table 6.28. The test set consists of the rectangles belonging to the clusters being closest to the real supplementary sign position as positive samples. The negative samples are taken from speed limit signs without supplementary signs and are taken from all 16 clusters. The high number of negative samples respects the much higher a-priori probability for a sign without supplementary sign multiplied by the number of clusters.

	Verified as supplementary sign	Rejected
Has supplementary sign	2157	284
No supplementary sign	2235	225413
Precision	49.1%	
Recall	88.4%	

Table 6.28: Confusion table of supplementary sign verification classifier for Germany. The algorithm is a complete polynomial classifier of the 2nd degree based on the 20 HOG features introduced above. The verification classifier reduces the number of *cutouts* passed to the following pictogram classifier by two orders of magnitude. The number of false candidates is that high, since for each sign many possible supplementary sign positions, in this case 16, have to be tested. The training and testing sets are given in table 6.6

The boxes detected and verified in the previous two steps are normalized with the same algorithm used for the speed limit signs, see section 2.6.1. The normalized *cutouts* are classified using the same type of classification consisting of PCA and polynomial classifier as the speed limits. If there are more than one detection for a supplementary sign corresponding to one speed limit sign, all candidates are classified. Of all candidates' classification results only the one having the highest output value for a supplementary sign class is accumulated in a histogram. The reasoning is, that the candidate having the highest classification result is most likely the one fitting best to the real sign position. The highest accumulated result for a track has to be higher than a threshold (η) and the highest accumulated value relative to the sum of accumulated values has to exceed a second threshold (λ), see table 6.29.

Absolute track classifier threshold	η	1.30
Relative track classifier threshold	λ	0.50

Table 6.29: Thresholds for the supplementary sign track classifier.

The final classification results for supplementary signs are shown in table 6.30.

Conclusion: The results for the detection and classification of supplementary signs in table 6.30 show the high precision, with the exception of the signs in Belgium. The Belgian signs are comparatively small and the only ones in western Europe being printed white on blue, yielding a lower contrast for the pictogram. The recall for the detection is close to 90% as shown in table 6.28. Combined with the recall rate of about 90% for the pictogram classifier this yields a summarized recall rate of about 80%. This suffices since at higher order roads the signs are placed at both sides of the road allowing the correct interpretation of the traffic scene if just one of the supplementary signs is detected and correctly classified.

6.9 Separation of Active and Exit Signs

As an example for necessary extension modules the classification module for the separation of *active* speed limit signs from minimum speed and *exit* signs in Germany is presented. The necessity of this module is motivated in section 2.8.2 and is based on the inseparability of the two classes after the *cutout* normalization.

The algorithm is used for each *cutout* classified as being an *active* speed limit in the stage I of the speed limit classifier. As features the mean grey value and the standard deviation of the signs in the centre region of the *cutouts* are used.

In the ROC curve in figure 6.26, the lower right in the set, the acceptable performance of the polynomial classifier trained with these features can be perceived. In case the exposure time of the camera for the current frame is known this additional feature boosts the performance of the classifier even more. The polynomial

Country	Correct	Miscl.	False Pos.	Rejected	Precision
AT	524	30	2	154	
BE	379	11	46	46	
DE	2332	25	19	217	
FR	1439	8	6	135	
ES	1110	1	0	32	
LU	97	1	7	11	
PT	316	0	2	30	
AT	74.01%	4.24%	0.28%	21.75%	94.24%
BE	86.93%	2.52%	10.55%	10.55%	86.93%
DE	90.60%	0.97%	0.74%	8.43%	98.15%
FR	90.96%	0.51%	0.38%	8.53%	99.04%
ES	97.11%	0.09%	0.00%	2.80%	99.91%
LU	88.99%	0.92%	6.42%	10.09%	92.38%
PT	91.33%	0.00%	0.58%	8.67%	99.37%

Table 6.30: Classification results for the supplementary sign classifier. First the number of samples and below the percentages are shown. For each country one classifier is trained and tested on samples of this country only, the sample sets being given in table 6.6. The different types of results are correctly classified supplementary signs, supplementary signs mistaken for belonging to another supplementary sign class, false positives being non-sign areas being classified as a supplementary sign and rejected being supplementary signs classified as garbage or the classification result of which were rejected, see equation 4.17.

classifier is of order three on the two features introduced above, which is acceptable for just having two input features and two output classes. The classifier was trained with the features generated from the set shown in table 6.31.

	Active	Exit
Training Set	5000	1000
Test Set	209448	33277

Table 6.31: Training and testing sample set number of *cutouts* for the active versus exit sign classifier.

In figure 6.26 the testing set in the two dimensional feature space is displayed. In addition the black decision border between *minimum speed/exit* signs and *active* signs at the operating point, as selected in the ROC curve in the lower right part of the figure is depicted.

The errors introduced by this additional classification module are compara-

tively minor, since in this case the results are accumulated over the whole track of a sign as well, thus the wrong classification of one *active* limits *cutout* as *exit* sign does not mean that the real world sign associated with the track is rejected.

A geometric hint as for the validity of an *active* sign is the usual placement of more than one speed limit on one sign gantry. Should one or more of those be rejected, the consideration of the traffic scene will overrule this classification result. On the other hand many *exit* signs can be rejected due to their small size compared to an *active* sign.

The combination of this information leads to the *scene* wise result as shown in table 6.32.

	Class. as Active	Class. as Exit	Error in %	Sample N.
Active <i>cutouts</i>	201049	8399	4.01%	209448
Exit <i>cutouts</i>	3497	29780	10.51%	33277
Active <i>tracks</i>	19748	289	1.44%	20037
Exit <i>tracks</i>	181	4036	4.29%	4217
Active <i>scene</i>	8793	21	0.24%	8814
Exit <i>scene</i>	98	4119	2.32%	4217

Table 6.32: Classification results for the classifier *Active - Exit* on the base of *cutouts* and based on *tracks*, the accumulated results of tracked real world signs. The final two rows show the result on *scene* base. The results on scene base show the very low number of a quarter percent of scenes showing active traffic signs being rejected due to this module, while over 97% of the exit signs could be classified as such and thus be rejected.

Conclusion: The separation of active speed limits from exit signs shows a very good performance. Over 97% of the exit signs could be classified as such and thus be rejected. Based on the evaluation of traffic sign scenes the rejection of actually valid active signs, leading to wrong scene interpretations is lower than 1/4% and thus deemed acceptable given the high rejection rate of the exit signs. The input features just being the mean grey value and the standard deviation in the *cutouts* makes the retraining of this classifier easy, should the sensor characteristic be changed.

6.10 Over all Results of the Traffic Sign Recognition System

This section reveals the over all classification results when taking all results presented in the previous chapters into account. For Germany the values are broken down in more detail, see table 6.33. The interpretation the traffic *scenes* is done

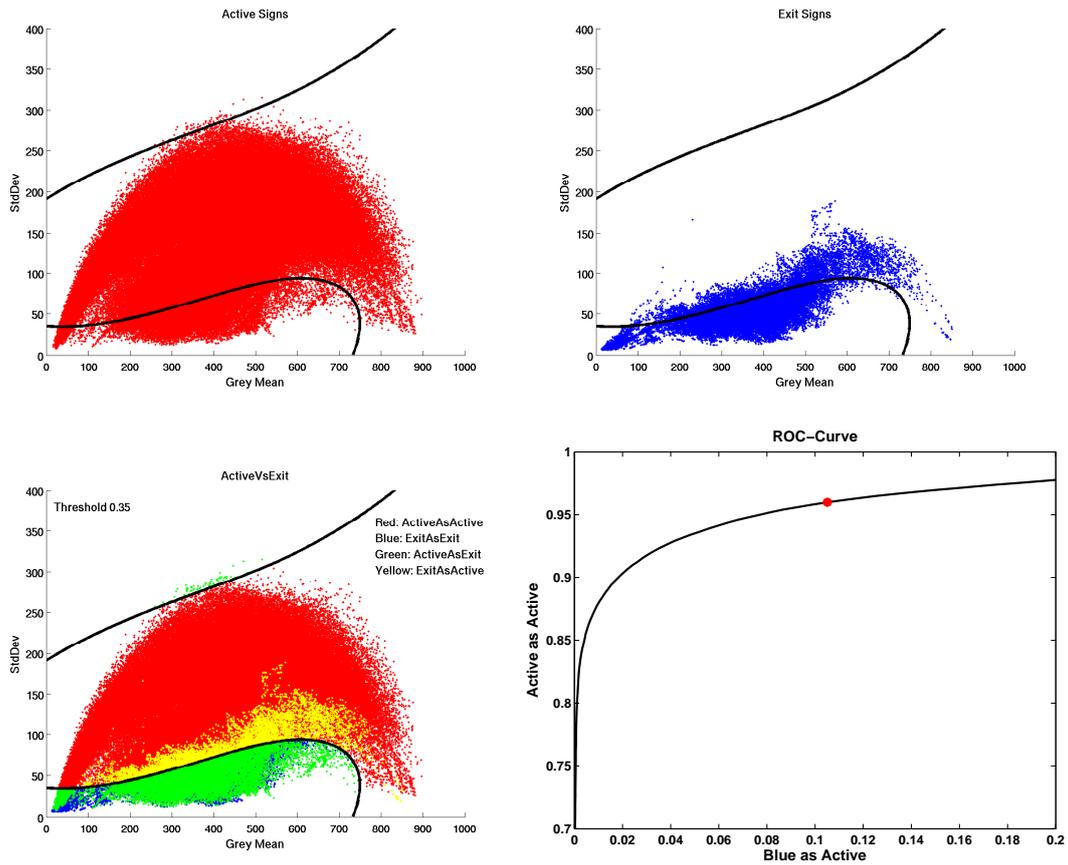


Figure 6.26: The top two graphs show the active, respectively the exit signs in the feature space used: standard deviation of the grey values over the mean grey value of the *cutout*. The lower left graph shows the classification errors at the chosen operating point and the line showing the separation done by the classifier. The lower right graph is the *Receiver Operator Curve (ROC)* of the classifier with a marker at the point of the chosen operating point. The solid line in the two top and the lower left graph is the decision border of the classifier.

using the rules stated in section 5.7. This means taking the results for all signs encountered on a certain strip of road into account when deciding for the correct speed limit, as well as regarding the results of the auxiliary algorithms like the three dimensional positioning system when reconstructing the current traffic scene.

Distance travelled [km]	12230
Amount of data [GB]	1587
Number of frames	3441982
Number of limit signs	17753
Number of end of and speed limit signs	14654
Number of speed limits with suppl. signs	2043
Number of scenes	8186
Number of <i>complex</i> scenes	3
Number of detected/recognized scenes	7972
Percentage of detected/recognized scenes	97.4%
Number of <i>complex</i> detected scenes	132
Distance travelled with correct scene interpretation [km]	11859
Percentage of correct <i>distance travelled</i>	97%

Table 6.33: Table showing the results for the traffic sign recognition system in Germany. The higher number of complex scenes in the actual detection system compared with the *ground truth* is explained by classification errors making an otherwise uncomplicated scene complex, for an example see figure 6.15.

The evaluation of the traffic sign recognition system including the scene interpretation is based on a per kilometre base, since it represents the felt performance for a driver better than an evaluation based on classification events. Table 6.34 shows the results on a distance base for the ten European countries. In addition Portugal was evaluated, where the Spanish classifiers were used, since the signs resemble one another in those two countries.

The table 6.34 shows that in most countries the traffic sign recognition system performs correctly in 93% to 95% of the distance travelled. The exceptions are Austria, Italy, Luxembourg and France. In Austria and Italy the comparatively high number of errors derives from the higher number of signs mounted on the back of lorries. Even after the reduction of this problem via the three dimensional positioning system these signs produce a high number of wrongly decided stretches of road due to the long stretch of road travelled until a correctly recognized traffic sign cancels the wrong speed limit. Using additional algorithms for the detection of other vehicles on the road or additional sensors like lidar or radar would solve this problem more conveniently.

In Luxembourg the variety of traffic sign appearances leads to a higher number

Cntr.	Dist. [km]	Wrong res. [km]	%	Wrong res. [km]	%
AT	3928	581.158	14.80%	426.088	10.85%
BE	1756	262.633	14.96%	123.372	7.03%
CH	2085	233.107	11.18%	126.891	6.09%
DE	12230	571.872	4.68%	353.447	2.89%
DK	1106	34.419	3.11%	30.611	2.77%
ES	6087	1006.196	16.53%	213.607	3.51%
FR	6412	1272.234	19.84%	467.048	7.28%
IT	7014	1852.126	26.41%	1041.011	14.84%
LU	498	83.265	16.72%	54.3674	10.92%
NL	6140	605.994	9.87%	462.180	7.53%
PT	2103	332.868	15.83%	107.678	5.12%
All	49359	6835.872	13.85%	3406.304	6.90%

Table 6.34: Result of the traffic sign recognition system. Country-wise and distance based. On the left the results without the use of the three dimensional information and without the use of the active vs. exit sign module, on the right the results with the use of these systems.

of errors. The use of all stage II classifier types instead of just the one used for France could help. This scheme would slow down processing time, but would boost the classification results. This scheme was not tested up to now.

In France a special effect of the realization of the traffic sign rule book leads to the comparatively high error rate. In France the exits of higher level roads like highways have a certain type of signposting. First a 110 sign with a supplementary sign showing that the sign is valid for the exit only, then a 90 speed limit having the same supplementary sign and then a 70 sign without a supplementary sign. Using the simple logic for scene interpretation introduced in section 5.7 this leads to the system showing 70 as valid velocity on the highway, since the final 70 sign has no supplementary sign and thus is deemed valid for the current road. With an additional rule for French traffic sign scenes this problem can be solved by rejecting the 70 sign if the other signs belonging to this assembly of traffic sign scenes have been recognized. This rule cuts the distance travelled with the wrong speed limit to one third, thus about 5%, since these situations lead to errors lasting long stretches of travelled distance.

The system presented in this dissertation relies on monocular images, ego motion information and the knowledge of the country currently travelled in alone. The use of other sources of information can help to reduce the number of errors even more. Examples for such information sources are:

- Navigation systems having knowledge of speed limits, city limits and road categories as well as the information when the vehicle left the current road

for invalidation of the speed limit last encountered.

- The detection of the number and geometry of lanes of the road. This helps validating the real world positions of the recognized signs and allows a better interpretation of traffic sign scenes.
- The detection of other road users via lidar, radar, optical flow, stereo vision or other means. This would help rejecting signs on the back of lorries.

7 Summary and Outlook

The objective of this thesis is to present a feasible chain of algorithms to realize a vehicle mounted traffic sign recognition system for the detection and verification of speed limit and end of speed limit signs. The system is based on the sensor input of a monocular grey value video camera. If available the system uses the ego motion of the vehicle, and thus the camera, as an additional input. For the determination of the country the vehicle is currently travelling in a GPS sensor is used.

To allow the use of the system in a multitude of countries algorithms and tools were developed, allowing the adaption of the system to the peculiarities of different countries with a minimum of human effort necessary.

In this thesis a new hough-like detection algorithm is presented. It is optimized for computational speed and detection capabilities as presented in section 6.3.2, allowing a detection rate of traffic signs being equal and higher than the ones presented in the literature. Compare to the literature a considerable speed up was achieved, allowing a computation time of less than 30 milliseconds on a 500 MHz CPU, DSP and iPhone 4. Since the detection is by far, over 90%, the most computational intensive part of the traffic sign detection and classification system, the complete detection and recognition system runs on the mentioned hardwares in about 30ms per frame for a 752x320 pixel image.

All free parameters in the detection and tracking system were adapted by optimizing the result on evaluation sets. The system was optimized to operate under all weather and lighting conditions. The evaluation sets reflect this requirement. For the classification module large sets of traffic sign images were gathered and labelled by a human operator using the powerful support of newly developed labelling tools. The number of labelled images showing traffic signs exceeds 3 million samples, roughly one quarter used as training set and three quarters for testing purposes. The samples were gathered in ten European countries, again respecting all weather and lighting conditions. The large number of samples is necessary for developing, training and evaluating the complex classification system.

The detector performance is high, 90% of the traffic signs circles, in an evaluation set consisting of 5000 circles, are detected and positioned to less than 2 pixels error in centre position and radius. Over 95% of the circles are found when allowing for 3 pixels position error. Only 0.5 non-sign circles per image, thus one circle in two frames, are detected in an evaluation set of 91146 images, reducing the number of possible circle position candidates of the sizes searched for in an image by a factor of 10^7 , thus reducing the computational load of the following

tracking and classification steps.

The tracking algorithm can be used with and without knowledge about the ego motion of the vehicle. If the ego motion is known the performance is slightly higher, but the main advantage is the capability of measuring the three dimensional position of the sign relative to the vehicle. The measurement of the position allows a more accurate interpretation of the road scenery. Additionally the algorithm used for measuring the position allows the rejection of traffic signs mounted on the back of lorries. The algorithm is capable of self calibrating the important angles and position values of the camera relative to the vehicle.

In this thesis emphasis is laid on the process of adapting the hierarchical classification system used for categorising the detected circular real world objects into non-sign objects and traffic signs and for the signs the separation into the different speed limits. The chosen classification algorithm is a complete quadratic polynomial classifier optimized in the mean square sense. The features are normalized grey values with a feature reduction via a principal component analysis.

Special attention was paid to the *internationalisation* of the system. This means the adaptation of the system to the local appearances of the traffic signs in different countries. The classification system is designed to be adaptable to these differences in traffic sign appearances with a minimum of human intervention being required. One important module is the acquisition and selection of training samples. A novel approach for the creation of realistic *synthetic* sample images of traffic signs was developed and evaluated. A framework for the fast labelling of large sample sets via bootstrapping and classifiers in the loop was developed and applied for the labelling of millions of samples from recordings of about 50 thousand kilometres of road, allowing the adaptation of the multiple classifiers in the hierarchical classification system.

To reach the classification performance necessary for implementing an European traffic sign recognition system covering all different sign appearances one classifier is not sufficient. Having one class or classifier for each country is not feasible when considering the large number of countries. For the EU alone 27 classifiers would be necessary, or 46 for the whole of Europe, when expecting one appearance type per country. This leads to the task of clustering the classifiers for different countries by appearance types of the signs used in these countries. In this thesis this process was developed and shown for ten European countries. The results show, that the number of classifiers can be reduced to three, still allowing for adequate classifier performance in each country. The developed algorithm allows the extension of the functionality to further countries with a minimum of human work required.

A detail disregarded in all but a very few publications is the recognition of supplementary signs. A large sample set from Germany showed that about 10% of the speed limit signs are explained or constrained by a supplementary sign. To allow for an adequate functionality of the traffic sign recognition system these signs have to be observed. The algorithm developed consists a detector, a ver-

ification and ranking step as well as a classifier for the decision which type of supplementary sign was detected. The detection step creates hypotheses for the possible position of supplementary signs relative to the speed limit sign positions based on the clustering of previously observed positions. The verification and ranking step classifies the selected positions using *histograms of oriented gradients* features and sorts the results by their plausibility. At the location of best results, if indicating the existence of a supplementary sign, a classifier based on grey value features discerns which type of sign was detected. The type of classifier used is the same as the one used for the speed limit signs as explained above, a complete quadratic polynomial classifier optimized in the mean square sense.

For optimal performance some specialities in different countries have to be considered. One example elaborated in this thesis is the rejection of exit indications in Germany. These signs are exact lookalikes to active speed limits in brightness normalized images. An additional classifier working on alternative features allows the rejection of the exit signs with a minimal loss of falsely rejected active speed limits.

For the realistic measurement of the over all system performance a traffic scene interpretation based on all the clues gathered by the system was developed. In this algorithm the decision for the currently valid speed limit based the detected traffic signs, supplementary signs and their positions relative to the vehicle along a stretch of road, their probability to be lookalikes or being mounted on the back of lorries is done. This functionality allows the assessment of the complete traffic sign recognition system on the base of kilometres driven with the correct speed limit being displayed. The resulting tables show that in Germany, where the traffic scene interpretation was based on a very large number of scenes, the system displays the correct speed limit for 97% of the driven kilometres, 11650 km of 12200 km. In some of the other ten evaluated European countries the quality of the scene interpretation is less well adapted, but still the mean distance driven with a correct speed limit display is above 93%.

7.1 Future Work and Outlook

Future work will be the improvement of the scene logic and the inclusion of more countries. A helpful step will be the enhancement of the sensor resolution allowing the better interpretation of the supplementary signs, maybe even using OCR for reading the writing on them. Further work will be done for the detection of signs in countries not following the Vienna convention from 1968, namely the USA and Canada. The tracking and classification system will work in these countries similar as in all other countries.

Further improvements of the system can be achieved by the use of additional environment perception modules. Especially the detection of Lorries, allowing the better rejection of the signs mounted on the back of these vehicles is helpful. The detection of lorries can be solved by using radar, lidar or optical stereo sensors. Another helpful module would be the complete integration of a lane detection system allowing a better scene interpretation. The use of a navigation system would be helpful as well, especially the knowledge about multiple lanes, crossings, exit roads and parallel roads as well as city limits.

A Appendix

A.1 Definitions and Acronyms

AdaBoost	Adaptive Boosting Classifier
Bayer-Pattern	See section A.6
CCD	C harge C oupled D evice
Chamfer Matching	Image matching algorithm, see A.3
CMOS	C omplementary M etal O xide S emiconductor
Cutout	Rectangular part of the input raster image holding A detected object
Ground Truth	Elements labelled by a human operator. The elements are circles in the image, Image <i>cutouts</i> labelled for their sign class, Image objects tracked in an image sequence belonging to a single real world object and Traffic scenes holding one or more real world signs and
HAV Labels	The class labels for the classified signs are taken from the HAV [Bald and Giesa, 2002]. The class prefix 274 stands for a speed limit, 276 is a non-overtaking sign for passenger cars, 277 is a non-overtaking sign for lorries. 278 is an end of speed limit, 280 is an end of non-overtaking for passenger cars, 281 is an end of non-overtaking for lorries and a 282 is a general end of limits. The number after the dash for speed limits and end of speed limits refers to the speed digits. 51 to 59 are the digits 10 to 90. 60 to 63 are the digits 100 to 130. An additional “i” at the end is a flag showing that it is an <i>inverse</i> sign with white digits on a black background, usually an <i>active</i> sign. See figure A.3 .
HOG	Histogram of Oriented Gradients
Internationalization	Adaptation to the differences between the signs used in different countries which has to be addressed.

LDA	L inear D iscriminant A nalysis [Fisher, 1936]
LVDS	L ow V oltage D ifferential S ignaling
Manhattan metrics	$d(\mathbf{a}, \mathbf{b}) = \sum a_i - b_i $
On Vehicle	This describes signs affixed on the back of other vehicles, mainly lorries and their trailers. These signs indicate the maximum speed this vehicle is allowed to drive. Since these signs look exactly like <i>normal</i> signs they often lead to mistakes in the traffic sign recognition system.
PC	Polynomial Classifier
PCA	Principal Axis Transform
PROMETHEUS: Precision	P rogra M me for a E uropean T raffic S ystem of R ate of <i>positive</i> samples classified correctly relative to the number of all <i>positive</i> samples. H ighest E fficiency and U nprecedented S afety
Reliability	Rate of <i>positive</i> samples classified correctly relative to all samples classified positively.
RGB	Colour scheme R ed G reen B lue
Scene	Road scenario consisting of all traffic signs situated locally on a short way of road
Ten European Countries	Austria (AT), Belgium (BE), Denmark (DK), France (FR), Germany (DE), Italy (IT), Luxembourg (LU), Netherlands (NE), Portugal (PT), Spain (ES), Switzerland (CH)
USB	U niversal S erial B us
YIQ	Colour scheme with Y being the luminance, I and Q being colour difference signals.

A.2 AdaBoost

AdaBoost is the abbreviation of *Adaptive Boosting*. It is a machine learning algorithm proposed in [Freund and Schapire, 1996] by Yoav Freund and Robert Schapire. The boosting itself is the cascaded call of weak classifiers to create a strong classifier. The adaptive part of the boosting algorithm is the reweighting of input sample based on their being successfully recognised in a previous classification step. In the AdaBoost algorithm a weak classifier, usually is a simple thresholding on a single feature, is used to part the samples into *object* and *non-object*. Based on the success of this first classifier the weight of the training samples is adapted to give the samples classified wrongly a higher weight. This is repeated until a maximum number of weak classifiers is reached or the classification results are sufficing. This type of classifier is often used in a cascade, meaning that the following steps in the cascade see only the samples the previous classifier labelled as not belonging to the *garbage* class, thus focussing on the

more subtle differences than the previous steps.

In the classifier introduced by Viola and Jones [Viola and Jones, 2001] the algorithm was used on Haar wavelet [Haar, 1910] features, using thresholding as weak learner and linear mappings of those as strong learner. The main effect is finding the right wavelets from a huge number of possible candidates, thus allowing a fast computation of a single classification.

A.3 Chamfer Matching

A well known method for object detection in images is the chamfer matching algorithm. The algorithm was first proposed 1977 in the paper [Barrow et al., 1977]. For a speed-up of the algorithm in the year 1988 the use of a detection hierarchy was proposed in [Borgefors, 1988]. The use for the detection of speed limit signs based on this algorithm was proposed by Gavrilu in his papers [Gavrila, 1998], [Gavrila and Philomin, 1998] and [Gavrila, 1999]. The three main steps as depicted in figure A.1 are the following:

1. The computation of an edge image which is binarized using an appropriate threshold.
2. The computation of the chamfer distance transform image. Ideally in this transformed image each pixel value corresponds to the Euclidean distance of this pixel to the next set value in the binarized edge image. With respect to computational speed usually the Manhattan city block metrics is used instead of the Euclidean, allowing the pixel values to stay in the integer domain and allowing the computation of the distance transform image in just two passes over the image.
3. The template matching of the binarized edge image of the template in the distance transformed chamfer image. The lower the score of the correlation of the two images at a given point, the closer the edges of the two input images are to each other, thus the better the localization of the object.

To add robustness to the system the edge direction of the template can be used as well. Instead of using one binarized edge image the edge image is split into multiple images, each holding edges with similar direction only from these images, usually four to eight, the distance transformed chamfer images are computed. The templates edge image is split in the same way and the matching result is the sum of the single template matching steps in the separated directional images. To speed up the algorithm hierarchical approaches are used. Templates having a small chamfer distance between themselves are combined in the first detection steps, thus decreasing the number of passes over the chamfer images to be made. If there were low return values for the match, meaning the template was fitting well, the original templates are used to check which of them fits best at the

detected position. Another speed-up is reached by using a coarse to fine search. Since the chamfer matching returns a value correlated to the distance of the template to its position in the image, the image can be scanned in multiple pixel steps first and only at positions not too far away from a possible detection the image is scanned again in smaller steps.

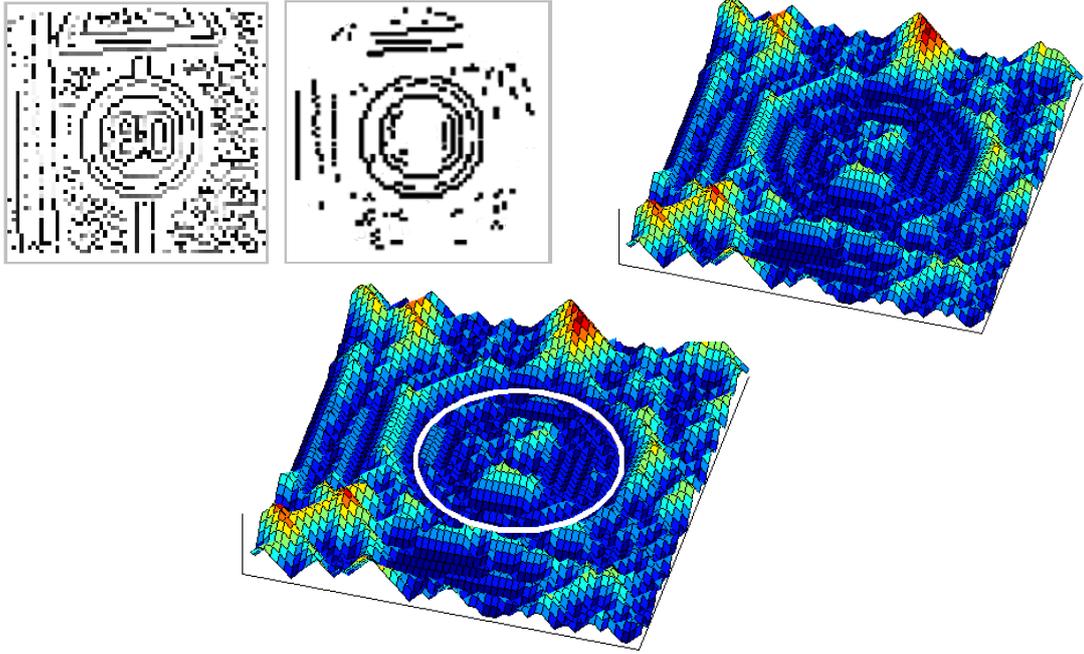


Figure A.1: The three chamfer matching steps: Edge image/filtered edge image, chamfer image, circle detection result

A.4 Camera Model

The camera model used is the Tsai model as proposed by R.Y. Tsai in [Tsai, 1986]. This model takes into account eleven parameters. The first is the focal length f of the camera as derived from the pinhole camera model where this parameter and the two coordinates of the principal point (p_u, p_v) describe the whole model. In addition to these three parameters there are another 2 intrinsic parameters, namely κ_1 , the 1st order radial lens distortion coefficient and s the scaling factor of the pixels in the digital sensor. The extrinsic parameters are the three translational parameters (t_x, t_y, t_z) and the three rotational parameters (r_x, r_y, r_z)

$$\begin{aligned}
\text{Extrinsic} \quad \begin{bmatrix} x \\ y \\ z \end{bmatrix} &= \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = TR_z R_y R_x \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \\
\text{using} \quad \begin{bmatrix} X_u \\ Y_v \\ 1 \end{bmatrix} &= \frac{f}{z} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \\
\text{Distortions} \quad \begin{bmatrix} X_d \\ Y_d \\ 1 \end{bmatrix} &= \begin{bmatrix} 1 & 0 & D_x \\ 0 & 1 & D_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_u \\ Y_v \\ 1 \end{bmatrix} \tag{A.1}
\end{aligned}$$

$$\begin{aligned}
\text{using} \quad D_x &= X_d \kappa_1 r^2 \quad D_y = Y_d \kappa_1 r^2 \quad r = \sqrt{X_d^2 + Y_d^2} \\
\text{Centring} \quad \begin{bmatrix} X_f \\ Y_f \\ 1 \end{bmatrix} &= \begin{bmatrix} s_x & 0 & C_x \\ 0 & 1 & C_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_d \\ Y_d \\ 1 \end{bmatrix} \\
\text{Yielding} \quad \begin{bmatrix} X_f \\ Y_f \\ NA \\ NA \end{bmatrix} &= D \frac{1}{z} TR_z R_y R_x \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \tag{A.2}
\end{aligned}$$

The reduced model used in this thesis considers the focal length and the position of the principal point only, since the cameras were adjusted to have a rotation relative to the outer camera coordinates low enough to be neglected and the lens had a very low radial distortion. This leads to the projection function as in equation [A.3](#).

$$\begin{aligned}
\begin{bmatrix} X_f \\ Y_f \\ NA \end{bmatrix} &\approx \frac{f}{z} \begin{bmatrix} 1 & 0 & C_x \\ 0 & 1 & C_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \\
\text{with } C_x \text{ and } C_y &\text{ in pixel coordinates as } C_U \text{ and } C_V \\
\begin{bmatrix} U \\ V \end{bmatrix} &\approx \frac{f}{Z_w} \begin{bmatrix} X_w \\ Y_w \end{bmatrix} + \begin{bmatrix} C_U \\ C_V \end{bmatrix} \tag{A.3}
\end{aligned}$$

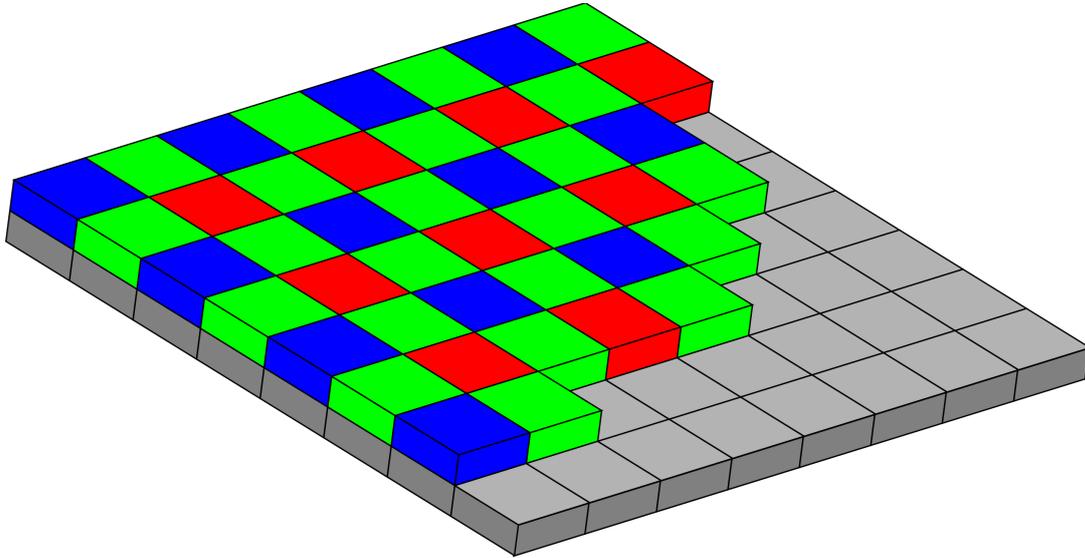
A.5 List of References For Colour as Primary Detector

Before and during the PROMETHEUS project all research and development groups working in the subject of traffic sign detection and recognition were using colour as primary detection step due to the limitations of computational power available and the indisputably important information to be gained from the exploitation of colour information. Traffic sign recognition papers connected with the PROMETHEUS project are:

[Akatsuka and Imai, 1987], [Ritter, 1992], [Kehtarnavaz et al., 1993], [Piccioli et al., 1994], [Wei, 1994], [Zheng et al., 1994], [Ritter et al., 1995a], [Ritter et al., 1995b], [Murino et al., 1995]. Numerous authors and groups are using colour as a primary detector even after more powerful computers were available, disregarding the drawbacks stated in the section 1.4. In the following the main publications with the subject of colour detectors for traffic signs following the PROMETHEUS project are cited: [Zadeh et al., 1997, de la Escalera and Moreno, 1997, Somol et al., 1999] [Miura et al., 2000, Fang et al., 2001, Lauziere et al., 2001, Vitabile et al., 2001] [Miura et al., 2002, Shaposhnikov et al., 2002, Sekanina and Torresen, 2002] [Fang et al., 2003, Fang et al., 2004, Paclik et al., 2004, Torresen et al., 2004] [de la Escalera et al., 2003, Lombardi et al., 2005, Wu and Tsai, 2005] [Silapachote et al., 2005, Shneier, 2005, Reina et al., 2006, Paclik et al., 2006a] [Paclik et al., 2006b, Gao et al., 2006, Torresen et al., 2006, Zhu and Liu, 2006] [Lopez and Fuentes, 2007, Ruta et al., 2007, Ruta et al., 2007, Tsai et al., 2007] [Liu et al., 2007, Bascon et al., 2007, Gao et al., 2008, Zhang et al., 2008].

A.6 Bayer-Pattern

The *Bayer-Pattern* is an additional coating on the sensitive part of the imager. To each pixel one of three colour filters is applied, either red or green or blue. Via demosaicing [Lukin and Kubasov, 2004] a colour image with a slightly wider *point spread function* can be computed. In a two by two pixel sensor area two green, one red and one blue filter is used. There are two green filters since green is closest to the maximum of the human *grey perception*. Figure A.2 depicts the general setup. For displaying the input images a simple nearest neighbour demosaicing was applied, e.g. composing the colour value from the mean of the respectively coloured pixels in the 3x3 vicinity of the output pixel. This algorithm uses the mean of all pixels touching the current pixel for each colour channel to retrieve the R G and B values of the pixel. For the normalization of the region of interest this scheme is not feasible due to the implicit smoothing effect of the demosaicing.

Figure A.2: The *Bayer-Pattern* on a sensor chip

A.7 Camera and Sensor Specifics

The sensor used, unless otherwise noted, is a MT9V029 CMOS sensor from *Micron Technology*, today renamed to *Aptina Imaging*. This sensor has 752x480 pixels, each of $6\mu\text{m}$ square size. The maximum frame rate is 60 Hz, but the used frame rate is 16 Hz. The camera has a global shutter and allows shutter times from $10\mu\text{s}$ up to 40 ms. The analogue-to-digital converter yields 10-bit images of a dynamic range up to 100 dB. The filters applied to the sensor limit the bandwidth to the wavelength range of 450 nm to 1050 nm. In addition to the band-pass a RGB *Bayer-Pattern* filter was added to allow the implementation of additional applications not being connected with traffic sign recognition on the same hardware platform.

The lens used has a focal length of 7mm and a lens aperture of 2.0 while having very low radial distortions.

The data transition is realized via an LVDS to USB connector.

A.8 Composition of the Classifier Training Set

The sizes and composition of the first two sample sets are given in section 6.2. For the two following sets this information is placed in the appendix due to the size of the necessary explanation. The third set was collected and assembled for a diploma thesis written by Denis Koch [Koch, 2007], which was supervised by the author of this dissertation. The class labels are taken from the German [Bald and Giesa, 2002] and explained in section A.1 and shown in figure A.3.



Figure A.3: Sign types and names taken from the German advice book for the placement of traffic signalisation [Bald and Giesa, 2002]

A.9 Composition of the System Evaluation Set

The set labelled based on the driven distance consists of ten different sets, one for each of ten European countries. The length of the labelled sequences in kilometres, the size in gigabytes, the number of images in the set, the number of signs and supplementary signs is given in the table A.1. The difference in the size of the sets in different countries is owed to the accessibility of the given country when starting in southern Germany and to the size of the given country itself. The sum of kilometres driven in all ten countries is 109192, or 11511 gigabytes. In total 25.912.000 images are in the set and 124373 speed limit or *end of limit* signs with an additional 13711 supplementary sign objects were labelled in those images.

	Austria	Belgium	Switzerl.	Germany	Denmark	Total
Abbreviation	AT	BE	CH	DE	DK	
Kilometres	5917	3171	4582	55856	2236	
GBytes of Data	658	404	495	5418	188	
Images (x1000)	1469	879	1043	12184	443	
Sign No.	6999	2150	3338	76691	1198	
Suppl. Signs	1069	663	382	3997	21	
	Spain	France	Italy	Luxemb.	Netherl.	
Abbreviation	ES	FR	IT	LU	NL	
Kilometres	7752	10580	10397	994	7707	109192
GBytes of Data	874	1045	1258	158	1013	11511
Images (x1000)	1989	2300	2898	358	2349	25912
Sign No.	12267	7618	12434	711	4538	127944
Suppl. Signs	1682	3723	1988	144	42	13711

Table A.1: System evaluation set

A.9 Composition of the System Evaluation Set

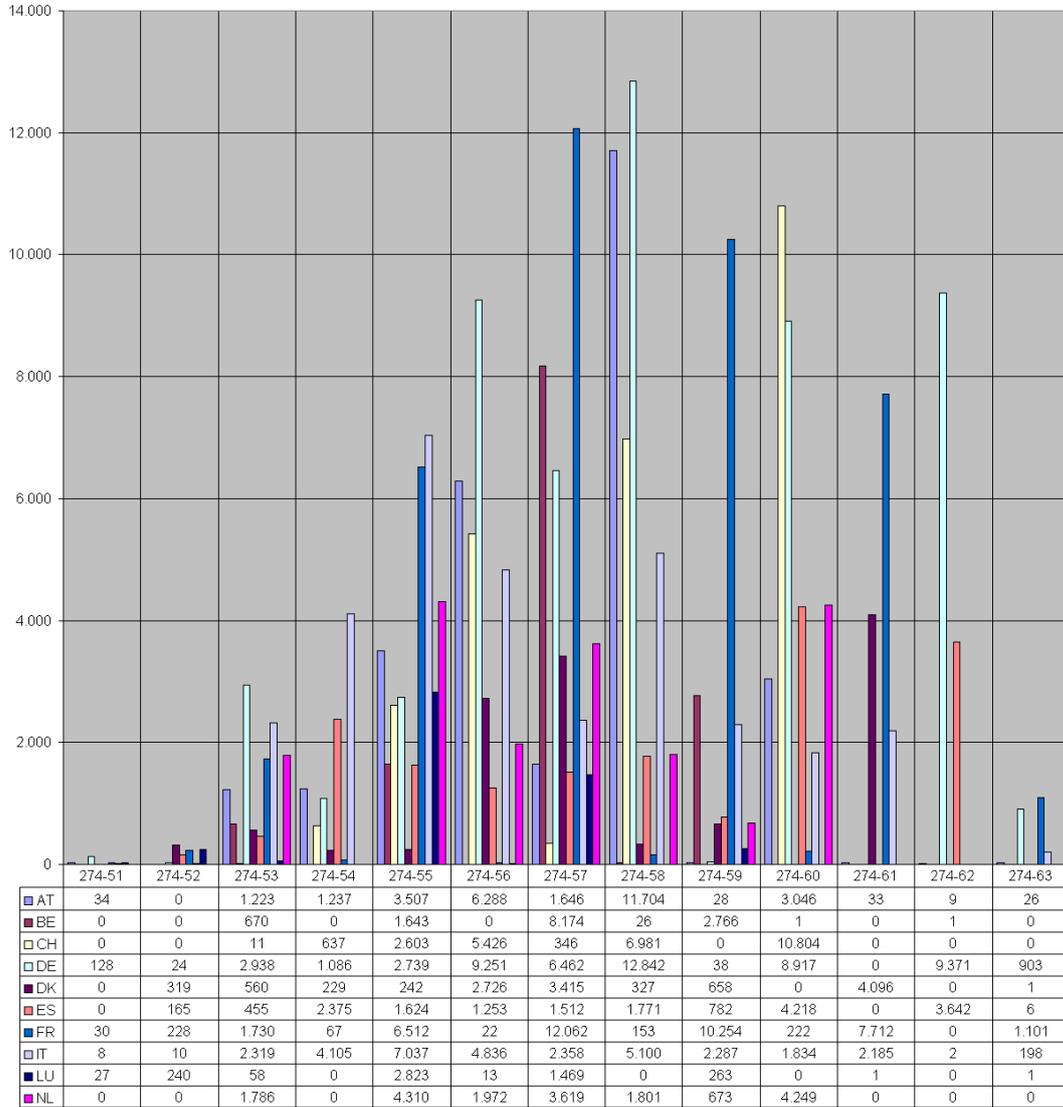


Figure A.4: Size and composition of sample set for inner circles of speed limits, see [Koch, 2007]. The ordinate axis shows the number of samples for each class, the abscissa is divided into the different classes and in the classes colour coded for country type.

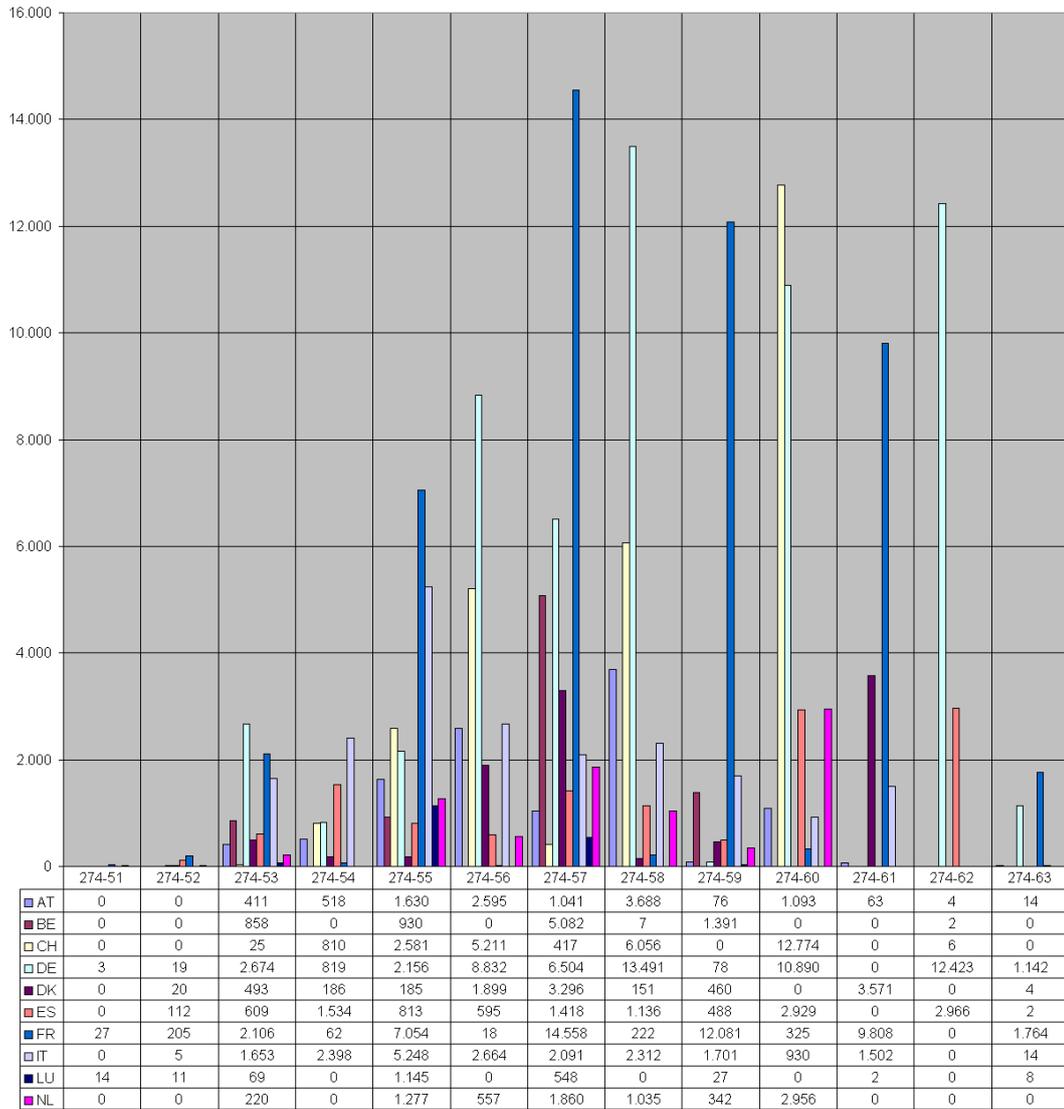


Figure A.5: Size and composition of sample set for outer circles of speed limits, see [Koch, 2007]. The ordinate axis shows the number of samples for each class, the abscissa is divided into the different classes and in the classes colour coded for country type.

A.9 Composition of the System Evaluation Set

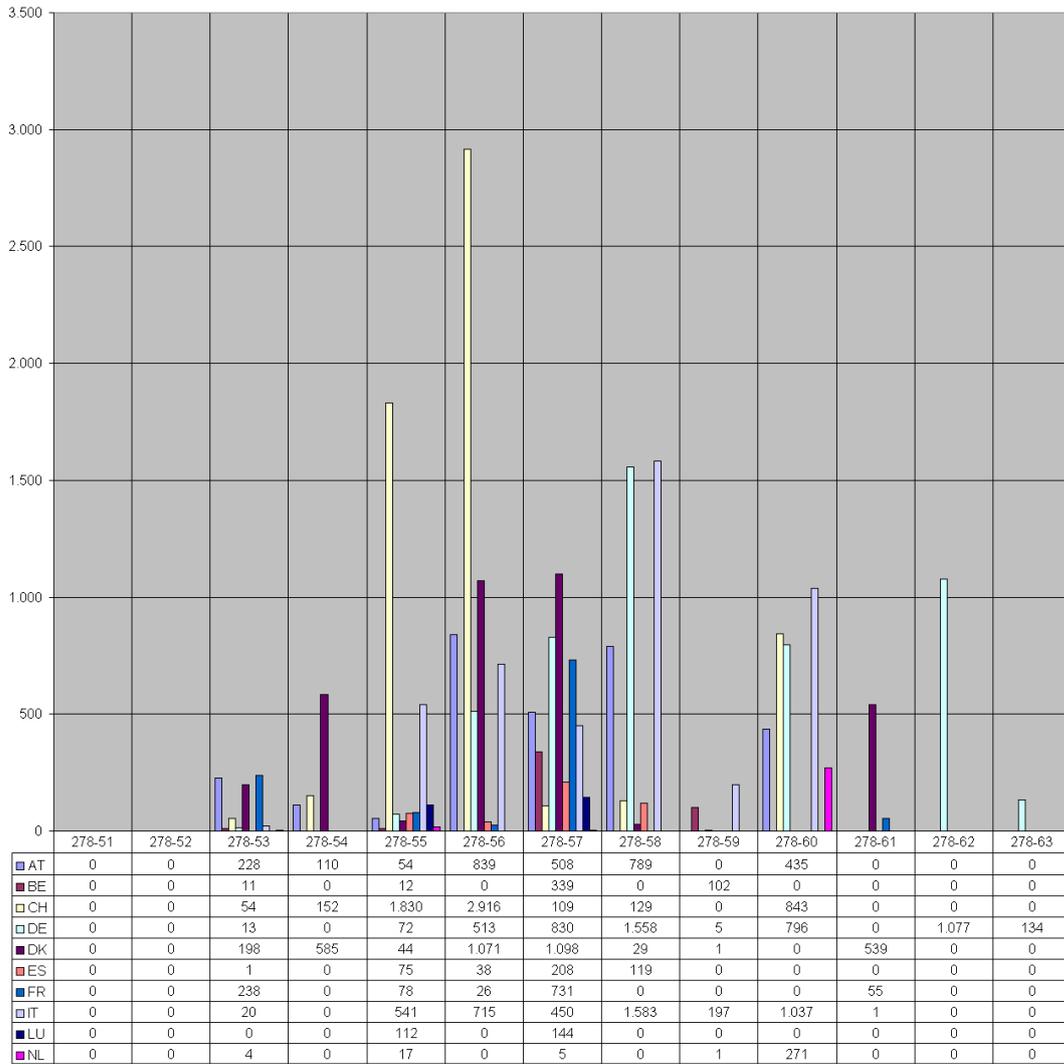


Figure A.6: Size and composition of sample set for circles of *end of speed limits*, see [Koch, 2007]. The ordinate axis shows the number of samples for each class, the abscissa is divided into the different classes and in the classes colour coded for country type.

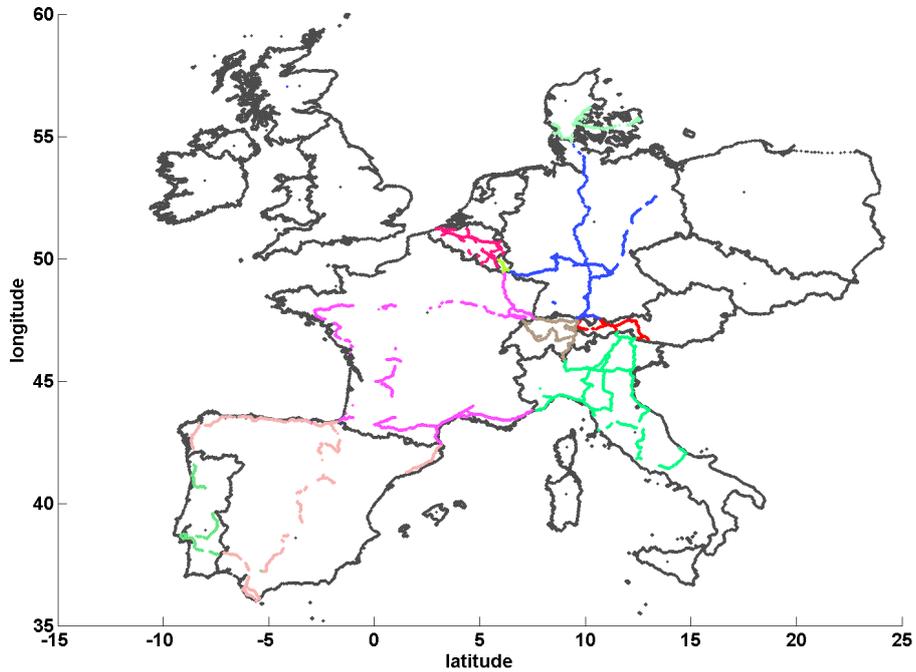


Figure A.7: Map showing the recorded streets travelled while recording the evaluation set. Some sections are missing due to drop-outs of the GPS recording.

A.10 Implementation on Mobile Devices

The system developed in this thesis has been ported to several mobile devices, one being a Basler Excite exA1390-19m camera having 1392 x 1040 Pixel resolution, 12 bit grey value depth and an internal 1 GHz Processor running the Linux operating system. The system performed even better than with the standard camera described in chapter 2.3 and reached a frame rate of 18 Hz, the maximum frame rate of the camera hardware. The time necessary to port the system from a desktop computer to the camera was less than 30 minutes due to the complete implementation of the software in ANSI C.

A second implementation on a portable device was implemented on the Apple iPhone 3G where the system was running at 12 Hz on the internal 400x304 Pixel preview image. The full capabilities of the imaging sensor or the mobile phone could not be used due to the restrictive software interface available to applications on this hardware. The results were promising even when the resolution of the images was low compared to the reference system and the capabilities of the sensor at night were not sufficient to allow for high vehicle speeds. Figure A.8

shows the mounting of the iPhone and the system at work.



Figure A.8: Prototypical interface of traffic sign recognition ported to iPhone 3G

A.11 Generalization of the Coarse Detector

The detector can be used for all convex forms allowing the use of the same algorithm as explained in 2.4.1 for detecting the triangular, rectangular and octagonal signs. The segmented line increments in the accumulator are replaced by the incrementation of an area in trapezoid shape as depicted in figure A.9. The edge detection, maximum extraction, and candidate creation stay the same. In the verification step the template searched for has to be replaced by the fitting shape. The detection algorithm has been tested on *Right of Way*, *Yield*, *Stop* and the various *Danger signs*. The additional computation time necessary for incrementing a trapezoid instead of just a line is balanced by the fact that only few edge directions necessitate the incrementing at all. This algorithm has been presented in the diploma thesis of Samuel Dirska [Dirska, 2005], which was tutored by the author of this thesis.

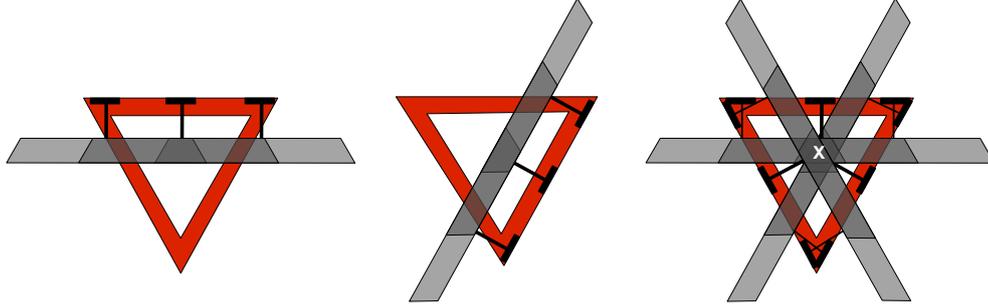


Figure A.9: Accumulator fill algorithm for a triangular traffic sign shape. Left three exemplary trapezoids for a horizontal edge, in the middle for a 60 edge and to the right for all edge directions.

A.12 Derivation of Equations in the Three-Dimensional Measurement

In the following the derivation of the equation 2.31 is explained.

$$\mathbf{S}_v = \mathbf{x}_{v0} + u \begin{pmatrix} \cos(\beta_0) \\ \sin(\beta_0) \end{pmatrix} \stackrel{!}{=} \mathbf{x}_{v1} + r_V \begin{pmatrix} \cos(\alpha + \beta_1) \\ \sin(\alpha + \beta_1) \end{pmatrix} \quad (\text{A.4})$$

Substituting to remove the auxiliary variable u leads to:

$$u = \frac{\sin(\alpha) + r \sin(\alpha + \beta_1)}{\sin(\beta_0)} \quad (\text{A.5})$$

$$\frac{\cos(\beta_0)}{\sin(\beta_0)} (d \sin(\alpha) + r \sin(\alpha + \beta_1)) = d \cos(\alpha) + r \cos(\alpha + \beta_1) \quad (\text{A.6})$$

Reorganizing the equation for r_V yields:

$$r_V \left(\cos(\alpha + \beta_1) - \frac{\cos(\beta_0)}{\sin(\beta_0)} \sin(\alpha + \beta_1) \right) = d \left(\frac{\cos(\beta_0)}{\sin(\beta_0)} \sin(\alpha) - \cos(\alpha) \right) \quad (\text{A.7})$$

Solving for r_V leads to the result shown in 2.31:

$$r_V = d \frac{\cos(\alpha) \sin(\beta_0) - \sin(\alpha) \cos(\beta_0)}{\sin(\beta_1 + \alpha) \cos(\beta_0) - \cos(\beta_1 + \alpha) \sin(\beta_0)} \quad (\text{A.8})$$

For the equation showing the result when observing the different positions of

turning centre and camera:

$$S_C = \begin{pmatrix} l_x \\ l_y \end{pmatrix} + u \begin{pmatrix} \cos(\beta_0) \\ \sin(\beta_0) \end{pmatrix} = d \begin{pmatrix} \cos(\alpha) \\ \sin(\alpha) \end{pmatrix} + l_x \begin{pmatrix} \cos(\alpha) \\ \sin(\alpha) \end{pmatrix} + l_y \begin{pmatrix} -\sin(\alpha) \\ \cos(\alpha) \end{pmatrix} + r_C \begin{pmatrix} \cos(\alpha + \beta_1) \\ \sin(\alpha + \beta_1) \end{pmatrix} \quad (\text{A.9})$$

Substituting to remove the auxiliary variable u leads to:

$$u = \frac{d \sin(\alpha) + l_x \sin(\alpha) + l_y \cos(\alpha) + r_C \sin(\alpha + \beta_1) - l_y}{\sin(\beta_0)} \quad (\text{A.10})$$

$$l_x + \frac{\cos(\beta_0)}{\sin(\beta_0)} (d \sin(\alpha) + l_x \sin(\alpha) + l_y \cos(\alpha) + r_C \sin(\alpha + \beta_1) - l_y) = d \cos(\alpha) + l_x \cos(\alpha) - l_y \sin(\alpha) + r_C \cos(\alpha + \beta_1) \quad (\text{A.11})$$

Reorganizing the equation for r_C yields:

$$r_C (\cos(\beta_0) \sin(\alpha + \beta_1) - \sin(\beta_0) \cos(\alpha + \beta_1)) = d (\sin(\beta_0) \cos(\alpha)) + l_x (\sin(\beta_0) \cos(\alpha) - \sin(\beta_0) + \cos(\beta_0) \sin(\alpha)) + l_y (-\sin(\beta_0) \sin(\alpha) + \cos(\beta_0) + \cos(\beta_0) \cos(\alpha)) \quad (\text{A.12})$$

Solving for r_C leads to the result shown in 2.31:

$$r_C = d \frac{\cos(\alpha) \sin(\beta_0) - \sin(\alpha) \cos(\beta_0)}{\sin(\beta_1 + \alpha) \cos(\beta_0) - \cos(\beta_1 + \alpha) \sin(\beta_0)} + l_x \frac{(\cos(\alpha) - 1) \sin(\beta_0) - l_x \sin(\alpha) \cos(\beta_0)}{\sin(\beta_1 + \alpha) \cos(\beta_0) - \cos(\beta_1 + \alpha) \sin(\beta_0)} + l_y \frac{-\sin(\alpha) \sin(\beta_0) - (\cos(\alpha) - 1) \cos(\beta_0)}{\sin(\beta_1 + \alpha) \cos(\beta_0) - \cos(\beta_1 + \alpha) \sin(\beta_0)} \quad (\text{A.13})$$

A.13 Types of Supplementary Signs

In figure A.10 the supplementary signs being most important for the speed limit recognition system in Europe are depicted.

A.14 Active Versus Blue Exit Signs Utilizing Colour Information

Should, as in the present setup, the sensor deliver colour information this is the main clue to separate the two types of sign. Since the colour information is far from perfect, especially when the size of the object is close to the lower limit of the

detector. The *Bayer-Pattern* delivers reliable colour information on areas when the area underneath the four colour filters have a near similar brightness and the white balance for the current lighting is known. For small *cutouts* this is not to be taken for granted. Because of this additional information can be gathered from the gain in the *cutout*, since the blue *exit* and *minimum speed* signs are passive the variance in relation to the mean value tends to be considerably lower than for the actively powered signs. This feature is available for grey value sensors as well and allows a high discriminatory power for those as well.

The features gathered from colour, mean brightness and brightness variance depend highly on the sensor and thus cannot be used for training a general classifier as it is possible for the normalized *cutouts* from the main traffic sign classifier. The four features given to the decision unit are:

- The mean brightness of the *cutout*
- The brightness variance in the *cutout*
- The mean of the red values compared to the green and blue values in the rim area of the sign
- The mean of the blue values compared to the red and green values in the centre area

The first two measurements are used as input features for a polynomial classifier and the threshold on the output of the classifier is adapted as explained in chapter 2.8.2. On some of the devices the system was tested on, see section A.10, colour information was available and the last two measurements were used in addition to the first two.

A.15 Screen shot of the Evaluation Tool

In figure A.11 the evaluation tool used for developing the scene evaluation and enhancing the over all performance of the traffic sign recognition. Explanation of the displayed information top to bottom:

- Blue: Yaw-Rate
- Green - flat: Ground truth allowed speed
- Red Line: Traffic sign recognition system output - behind the ground truth when correct, error when visible
- Green: travelled speed
- Red dots: Speed limit signs

- Bottom half: type of regulation or sign encountered
- Type of supplementary sign (weather, time, lorry, direction)
- Type of traffic sign (active sign, existing sign not valid for current lane or on a vehicle)
- The last four rows describe errors made by the system
- General: all errors generate a signal here
- false alarm
- not classified (classifier rejection or not detected)
- misclassified

The tool allows to create statistics on the error rate and types of errors encountered as well as direct access to the scenes in which the error occurred.

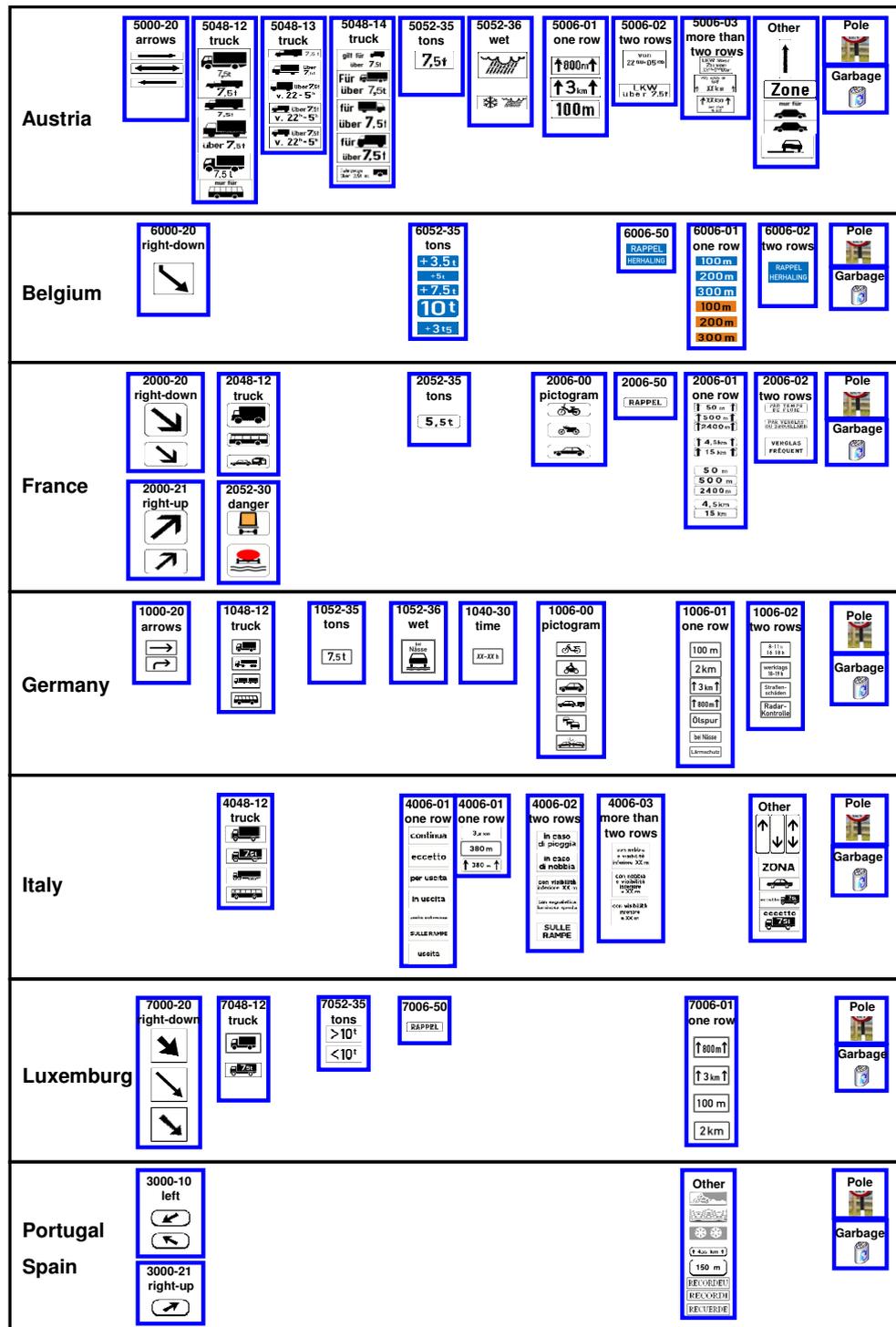


Figure A.10: European supplementary signs clustered for classification by meaning and appearance

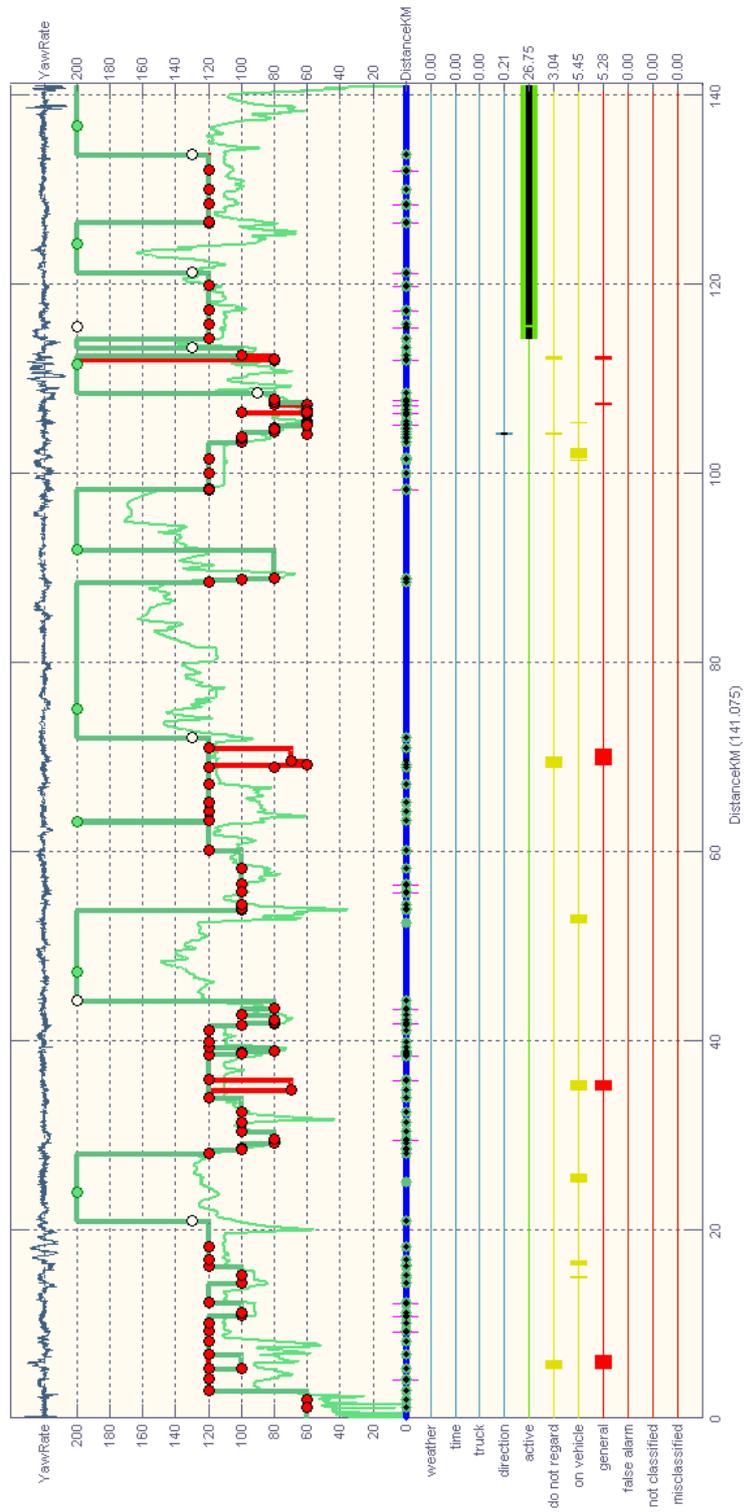


Figure A.11: Display of the evaluation tool for a 147 km long sequence. For Explanations see section A.15

Bibliography

- [Akatsuka and Imai, 1987] Akatsuka, H. and Imai, S. (1987). Road signpost recognition system. In *SAE vehicle highway infrastructure safety compatibility*, pages 189–196.
- [Bahlmann et al., 2005] Bahlmann, C., Zhu, Y., Ramesh, V., Pellkofer, M., and Koehler, T. (2005). A system for traffic sign detection, tracking, and recognition using color, shape, and motion information. In *Intelligent Vehicles Symposium*, pages 255–260.
- [Bald and Giesa, 2002] Bald, S. and Giesa, S. (2002). *HAV. Hinweise fuer das Anbringen von Verkehrszeichen und Verkehrseinrichtungen*. Kirschbaum Verlag.
- [Barnes and Zelinsky, 2004] Barnes, N. and Zelinsky, A. (2004). Real-time radial symmetry for speed sign detection. In *IVS04*, pages 566–571.
- [Barrow et al., 1977] Barrow, H., Tenenbaum, J., Bolles, R., and Wolf, H. (1977). Parametric correspondence and chamfer matching, two new techniques for image matching. In *IJCAI77*, pages 659–663.
- [Bartneck and Ritter, 1992] Bartneck, N. and Ritter, W. (1992). Colour segmentation with polynomial classification. In *ICPR92*, pages II:635–638.
- [Bascon et al., 2007] Bascon, M., Arroyo, L., Jimenez, G., Moreno, G., and Ferreras, L. (2007). Road-sign detection and recognition based on support vector machines. *ITS*, 8(2):264–278.
- [Borgefors, 1988] Borgefors, G. (1988). Hierarchical chamfer matching: A parametric edge matching algorithm. *PAMI*, 10(6):849–865.
- [Bresenham, 1965] Bresenham, J. (1965). Algorithm for computer control of a digital plotter. *IBM Systems Journal*, 4(1):25–30.
- [Dalal and Triggs, 2005] Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages Vol. II, pp. 886–893.
- [de la Escalera et al., 2003] de la Escalera, A., Armingol, J. M., and Mata, M. (2003). Traffic sign recognition and analysis for intelligent vehicles. *IVC*, 21(3):247–258.

-
- [de la Escalera and Moreno, 1997] de la Escalera, A. and Moreno, L. (1997). Road traffic sign detection and classification. *IndEle*, 44:848–859.
- [Dirska, 2005] Dirska, S. (2005). Detektion und Klassifikation von dreieckigen Verkehrszeichen. Master’s thesis, Technische Universitaet Ilmenau.
- [Duda et al., 2000] Duda, R., Hart, P., and Stork, D. (2000). *Pattern Classification (2nd ed.)*. Wiley Interscience.
- [Duda and Hart, 1972] Duda, R. O. and Hart, P. E. (1972). Use of the Hough transformation to detect lines and curves in pictures. *Communications of the Association for Computing Machinery*, 15:11–15.
- [Eder, 1999] Eder, S. (1999). Entwurf und Implementierung eines Verfahrens zur Erkennung von Verkehrszeichen durch Auswertung von Bildfolgen. Master’s thesis, Universitaet Ulm.
- [Escalera and Radeva, 2004] Escalera, S. and Radeva, P. (2004). Fast greyscale road sign model matching and recognition. In *Recent Advances in Artificial Intelligence Research and Development*, pages 69–77.
- [Estable, 1996] Estable, S. (1996). *Reconnaissance d’Objets en Environnement Exterieur Dynamique - Application a la Reconnaissance de Panneaux Routiers*. PhD thesis, Universite Blaise Pascal - Clermont II.
- [Estable et al., 1994] Estable, S., Schick, J., Stein, F., Janssen, R., Ott, R., Ritter, W., and Zheng, J. (1994). A real-time traffic sign recognition system. In *Proceedings of the Intelligent Vehicles ’94 Symposium*, pages 213–218.
- [Fang et al., 2001] Fang, C. Y., Fuh, C. S., and Chen, S. W. (2001). Detection and tracking of road signs. *PRIA*, 11:304–308.
- [Fang et al., 2003] Fang, C. Y., Fuh, C. S., Chen, S. W., and Yen, P. S. (2003). A road sign recognition system based on dynamic visual model. In *CVPR03*, pages I: 750–755.
- [Fang et al., 2004] Fang, C. Y., Fuh, C. S., Yen, P. S., Cherng, S., and Chen, S. W. (2004). An automatic road sign recognition system based on a computational model of human recognition processing. *CVIU*, 96(2):237–268.
- [Fisher, 1936] Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. In *Annals of Eugenics*, 7, pages 179–188.
- [Freund and Schapire, 1996] Freund, Y. and Schapire, R. E. (1996). Experiments with a new boosting algorithm. In *Machine Learning: Proceedings of the Thirteenth International Conference*, pages 148–156.

- [Gao et al., 2008] Gao, X. H., Hong, K., Passmore, P., Podladchikova, L., and Shaposhnikov, D. (2008). Colour vision model-based approach for segmentation of traffic signs. *JIVP*, 2008.
- [Gao et al., 2006] Gao, X. W., Podladchikova, L., Shaposhnikov, D., Hong, K., and Shevtsova, N. (2006). Recognition of traffic signs based on their colour and shape features extracted using human vision models. *JVCIR*, 17(4):675–685.
- [Gavrila, 1998] Gavrila, D. M. (1998). Multi-feature hierarchical template matching using distance transforms. In *ICPR*, pages 439–444.
- [Gavrila, 1999] Gavrila, D. M. (1999). Traffic sign recognition revisited. In *21st DAGM Symposium fuer Mustererkennung*, pages 86–93. Springer Verlag, Bonn, Germany.
- [Gavrila and Philomin, 1998] Gavrila, D. M. and Philomin, V. (1998). Real-time object detection using distance transforms. In *IEEE Intelligent Vehicles Symposium*, pages 274–279.
- [Glennie and Lichti, 2010] Glennie, C. and Lichti, D. D. (2010). Static calibration and analysis of the velodyne hdl-64e s2 for high accuracy mobile scanning. *Remote Sensing*, 2(6):1610–1624.
- [Haar, 1910] Haar, A. (1910). Zur Theorie der orthogonalen Funktionensysteme. In *Mathematische Annalen Vol. 69*, pages 331–371.
- [Hoehmann and Kummert, 2010] Hoehmann, L. and Kummert, A. (2010). Car2x - communication for vision-based object detection. In *International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, pages 290 – 294.
- [Hoessler, 2007] Hoessler, H. (2007). Generation d’échantillons virtuels pour la classification selon des distribution reelles. Master’s thesis, Ecole Supérieure De Physique De Strassbourg.
- [Hoessler et al., 2007] Hoessler, H., Woehler, C., Lindner, F., and Kressel, U. (2007). Classifier training based on synthetically generated samples. The 5th International Conference on Computer Vision Systems, 2007.
- [Illingworth and Kittler, 1987] Illingworth, J. and Kittler, J. (1987). The adaptive hough transform. *IEEE Transactions PAMI-9 (5)*, pages 690–698.
- [Ishida et al., 2007] Ishida, H., Takahashi, T., Ide, I., Mekada, Y., and Murase, H. (2007). Generation of training data by degradation models for traffic sign symbol recognition. *IEICE*, E90-D(8):1134–1141.

-
- [Kalata, 1984] Kalata, P. (1984). The tracking index: A generalized parameter for a $\alpha\beta$ and $\alpha\beta\gamma$ target trackers. *IEEE Transactions on Aerospace and Electronic Systems*, pages 174–182.
- [Kalman, 1960] Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME—Journal of Basic Engineering*, 82(Series D):35–45.
- [Kehtarnavaz and Ahmad, 1995] Kehtarnavaz, N. and Ahmad, A. (1995). Traffic sign recognition in noisy outdoor scenes. In *Proceedings of the Intelligent Vehicles 95 Symposium*, pages 460 – 465.
- [Kehtarnavaz et al., 1993] Kehtarnavaz, N., Griswold, N. C., and Kang, D. S. (1993). Stop-sign recognition based on color-shape processing. *MVA*, 6:206–208.
- [Keller et al., 2008] Keller, C. G., Sprunk, C., Bahlmann, C., Giebel, J., and Barattoff, G. (2008). Real-time recognition of u.s. speed signs. In *Proceedings of IEEE Intelligent Vehicles Symposium*.
- [Koch, 2007] Koch, D. (2007). Klassifikation von Geschwindigkeitsbegrenzungen fuer Westeuropa. Master’s thesis, Hochschule Aalen fuer Technik und Wirtschaft.
- [Kressel et al., 1999] Kressel, U., Lindner, F., Woehler, C., and Linz, A. (1999). Hypothesis verification based on classification at unequal error rates. In *Ninth International Conference on Artificial Neural Networks*, pages 874 – 879 vol.2.
- [Krueger et al., 2011] Krueger, J., Emmert, V., Feldmann, A., and Lindner, F. (2011). Evaluating the accuracy of calibration for driver assistance systems. In *Photogrammetrie Laser Scanning Optische 3D Messtechnik Oldenburger 3D Tage 2011*, pages 329–337.
- [Kuhn, 1955] Kuhn, H. (1955). The hungarian method for the assignment problem. In *Naval Research Logistic Quarterly*, 2, pages 83–97.
- [Lauziere et al., 2001] Lauziere, Y. B., Gingras, D., and Ferrie, F. P. (2001). A model-based road sign identification system. In *CVPR01*, pages I:1163–1170.
- [Li et al., 1986] Li, H., Lavin, M. A., and Master, R. J. L. (1986). Fast hough transform: A hierarchical approach. *Computer Vision Graph. Image Processing*, 36(2-3):139–161.
- [Lindner, 2010] Lindner, F. (2010). Ground Truthing von monokularen Entfernungsschaetzungen mittels eines Mehrzeilen-Laserscanners. In *Oldenburger 3D Tage 2010*, pages 210–220.

- [Lindner et al., 2004] Lindner, F., Kressel, U., and Kaelberer, S. (2004). Robust recognition of traffic signals. In *IVS04*, pages 49–53.
- [Liu et al., 2007] Liu, Y. S., Duh, D. J., Chen, S. Y., Liu, R. S., and Hsieh, J. W. (2007). Scale and skew-invariant road sign recognition. *IJIST*, 17(1):28–39.
- [Lombardi et al., 2005] Lombardi, L., Marmo, R., and Toccalini, A. (2005). Automatic recognition of road sign passo-carrabile. In *CIAP05*, pages 1059–1067.
- [Lopez and Fuentes, 2007] Lopez, L. D. and Fuentes, O. (2007). Color-based road sign detection and tracking. In *ICIAR07*, pages 1138–1147.
- [Loy and Barnes, 2004] Loy, G. and Barnes, N. (2004). Fast shape-based road sign detection for a driver assistance system. In *Intelligent Robots and Systems*, pages 70–75 vol.1.
- [Loy and Zelinsky, 2002] Loy, G. and Zelinsky, A. (2002). A fast radial symmetry transform for detecting points of interest. In *ECCV (1)*, pages 358–368.
- [Loy and Zelinsky, 2003] Loy, G. and Zelinsky, A. (2003). Fast radial symmetry for detecting points of interest. *PAMI*, pages 959–973.
- [Lukin and Kubasov, 2004] Lukin, A. and Kubasov, D. (2004). High-quality algorithm for Bayer pattern interpolation. *Program. Comput. Softw.*, 30(6):347–358.
- [MacQueen, 1967] MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings Fifth Berkeley Symposium on Mathematical Statistics and Probability, Vol. 1*, pages 281–297. University of California Press.
- [Mandler and Oberlaender, 1990] Mandler, E. and Oberlaender, M. (1990). One-pass encoding of connected components in multivalued images. In *10th ICPR, Atlantic City, New Jersey*, pages 66–69, Vol 2.
- [Miura et al., 2002] Miura, J., Kanda, T., Nakatani, S., and Shirai, Y. (2002). An active vision system for on-line traffic sign recognition. *IEICE*, E85-D(11):1784–1792.
- [Miura et al., 2000] Miura, J., Kanda, T., and Shirai, Y. (2000). An active vision system for real-time traffic sign recognition. In *ITS00*, pages 52–57.
- [Moutarde et al., 2007] Moutarde, F., Bargeton, A., Herbin, A., and Chanussot, L. (2007). Robust on-vehicle real-time visual detection of american and european speed limit signs, with a modular traffic signs recognition system. In *Intelligent Vehicles Symposium, 2007 IEEE*, pages 1122 – 1126.

-
- [Murino et al., 1995] Murino, V., Regazzoni, C. S., Foresti, G. L., and Vernazza, G. (1995). A multilevel fusion approach to object identification in outdoor road scenes. *PRAI*, 9:23–65.
- [Ott, 1977] Ott, R. (1977). *Ueber zweistufige quadratische Klassifikatoren*. PhD thesis, Erlangen-Nuernberg, University.
- [Paclik et al., 2006a] Paclik, P., Novovicova, J., and Duin, R. P. W. (2006a). Building road-sign classifiers using a trainable similarity measure. *ITS*, 7(3):309–321.
- [Paclik et al., 2006b] Paclik, P., Novovicova, J., and Duin, R. P. W. (2006b). A trainable similarity measure for image classification. In *ICPR06*, pages III: 391–394.
- [Paclik et al., 2004] Paclik, P., Verzakov, S., and Duin, R. P. W. (2004). Multi-class extensions of the gldb feature extraction algorithm for spectral data. In *ICPR04*, pages IV: 629–632.
- [Pearson, 1901] Pearson, K. (1901). On lines and planes of closest fit to a system of points in space. In *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science. Series 6, 2*, page 559–572.
- [Piccioli et al., 1994] Piccioli, G., de Micheli, E., and Campani, M. (1994). A robust method for road sign detection and recognition. In *ECCV94*, pages A:493–500.
- [Powell, 1992] Powell, M. J. D. (1992). The theory of radial basis function approximation in 1990. *Advances in Numerical Analysis II: Wavelets, Subdivision, Algorithms, and Radial Basis Functions*, pages 105–210.
- [Priese et al., 1994] Priese, L., Klieber, J., Lakmann, R., Rehrmann, V., and Schian, R. (1994). New results on traffic sign recognition. In *Proceedings "Intelligent Vehicles Symposium '94", Paris, 14-16th October 1994*, pages 249–254.
- [Priese et al., 1993] Priese, L., Rehrmann, V., Schian, R., and Lakmann, R. (1993). Traffic sign recognition based on color image evaluation. In *Proceedings "Intelligent Vehicles Symposium '93", Tokyo, 14-16th July, 1993*, pages 95–100.
- [Radon, 1917] Radon, J. (1917). Ueber die Bestimmung von Funktionen durch ihre Integralwerte laengs gewisser Mannigfaltigkeiten. In *Berichte Saechische Akadademie der Wissenschaften, Mathematisch Physikalisches Institut, Vol. 69*, pages 262–277.

- [Rehrmann et al., 1995] Rehrmann, V., Lakmann, R., and Priebe, L. (1995). A parallel system for realtime traffic sign recognition. In *Proceedings "International Workshop on Advanced Parallel Processing Technologies '95 (APPT)", Peking, 26-27th September, 1995.*, pages 72–78. Publishing House of Electronics Industry.
- [Reina et al., 2006] Reina, A. V., Sastre, R. J. L., Siegmann, P., Arroyo, S. L., and Moreno, H. G. (2006). An approach to the recognition of informational traffic signs based on 2-d homography and svms. In *ACIVS06*, pages 1163–1173.
- [Ritter, 1992] Ritter, W. (1992). Traffic sign recognition in color image sequences. In *Proceedings of the Intelligent Vehicles Symposium 1992;92 Symposium*, pages 12–17.
- [Ritter, 1996] Ritter, W. (1996). *Automatische Verkehrszeichenerkennung*. PhD thesis, Universitaet Koblenz Landau.
- [Ritter et al., 1995a] Ritter, W., Stein, F., and Janssen, R. (1995a). Traffic sign recognition using colour information. *MathMod*, 22:149–161.
- [Ritter et al., 1995b] Ritter, W., Stein, F., and Janssen, R. (1995b). Traffic sign recognition using colour information. In *Mathematical and Computer Modelling, Volume 22*, pages 149–161.
- [Ruta et al., 2007] Ruta, A., Li, Y., and Liu, X. (2007). Towards real-time traffic sign recognition by class-specific discriminative features. In *BMVC07*.
- [Schuermann, 1977] Schuermann, J. (1977). *Polynomklassifikatoren fuer die Zeichenerkennung*. Oldenbourg Verlag.
- [Schuermann, 1996a] Schuermann, J. (1996a). *Pattern Classification*, pages 244–248. John Wiley & Sons.
- [Schuermann, 1996b] Schuermann, J. (1996b). *Pattern Classification*, pages 102–186. John Wiley & Sons.
- [Sekanina and Torresen, 2002] Sekanina, L. and Torresen, J. (2002). Detection of norwegian speed limit signs. In *Proceedings of 16th European Simulation Multiconference*, pages 337–340.
- [Shaposhnikov et al., 2002] Shaposhnikov, G. D., Lubov, N., Golovan, E. V., and Shevtsova, A. (2002). Road sign recognition by single positioning of space-variant sensor window. In *Proceedings 15th International Conference on Vision Interface*.

-
- [Shneier, 2005] Shneier, M. (2005). Road sign detection and recognition. In *Unmanned Systems Technology VIII. Proceedings of the SPIE*, pages 6230, pp.
- [Silapachote et al., 2005] Silapachote, P., Hanson, A. R., and Weiss, R. (2005). A hierarchical approach to sign recognition. In *WACV05*, pages I: 22–28.
- [Somol et al., 1999] Somol, P., Pudil, P., Novoviova, J., and Paclik, P. (1999). Road sign classification using laplace kernel classifier. In *SCIA99*, pages 275–282.
- [Tarajan, 2004] Tarajan, E. (2004). Ortsbasierte Automatisierte Erfassung von Verkehrszeichen. Master’s thesis, Technische Universitaet Ilmenau.
- [Torresen et al., 2004] Torresen, J., Bakke, J. W., and Sekanina, L. (2004). Efficient recognition of speed limit signs. In *Intelligent Transportation Systems*, pages 652–656.
- [Torresen et al., 2006] Torresen, J., Bakke, J. W., and Yang, Y. (2006). A camera based speed limit sign recognition system. In *Proceedings of 13th ITS World Congress and Exhibition*.
- [Tsai et al., 2007] Tsai, L. W., Tseng, Y. J., Hsieh, J. W., Fan, K. C., and Li, J. J. (2007). Road sign detection using eigen color. In *ACCV07*, pages 169–179.
- [Tsai, 1986] Tsai, R. (1986). An efficient and accurate camera calibration technique for 3d machine vision. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 364–374.
- [United Nations Economic and Social Council, 1968] United Nations Economic and Social Council (1968). *Vienna Convention on Road Signs and Signals*. UNESCO.
- [Viola and Jones, 2001] Viola, P. and Jones, M. J. (2001). Robust real-time object detection. In *Technical Report CRL 2001/01*. Cambridge Research Laboratory.
- [Vitabile et al., 2001] Vitabile, S., Pollaccia, G., Pilato, G., and Sorbello, E. (2001). Road signs recognition using a dynamic pixel aggregation technique in the hsv color space. In *CIAP01*, pages 572–577.
- [Wei, 1994] Wei, L. S. (1994). Recognition of traffic signs using a multilayer neural network. In *Canadian Conference on Electrical and Computer Engineering*.
- [Woehler and Anlauf, 1999] Woehler, C. and Anlauf, J. K. (1999). A time delay neural network algorithm for estimating image-pattern shape and motion. In *IVC*, pages 281–294.

- [Wu and Tsai, 2005] Wu, J. P. and Tsai, Y. C. (2005). Real-time speed limit sign recognition based on locally adaptive thresholding and depth-first-search. *PhEngRS*, 71(4):405–414.
- [Zadeh et al., 1997] Zadeh, M., Kasvand, T., and Suen, C. Y. (1997). Localization and recognition of traffic signs for automated vehicle control systems. In *Proceedings SPIE Vol. 3207 Intelligent Transportation Systems*, pages 272–282.
- [Zhang et al., 2008] Zhang, K., Sheng, Y., Wang, P., Luo, L., Ye, C., and Gong, Z. (2008). Automatic recognition of traffic signs in natural scene image based on central projection transformation. In *ISPRS08*, pages 627–.
- [Zheng et al., 1994] Zheng, Y.-J., Ritter, W., and Janssen, R. (1994). An adaptive system for traffic sign recognition. In *Intelligent Vehicles '94 Symposium*, pages 165–170.
- [Zhu and Liu, 2006] Zhu, S. and Liu, L. (2006). Traffic sign recognition based on color standardization. In *International Conference on Information Acquisition*, pages 951 – 955.
- [Zuniga et al., 1982] Zuniga, O., Shapiro, L., and Lumia, R. (1982). A new connected components algorithm for virtual memory computers. In *PRIP82*, pages 560–565.