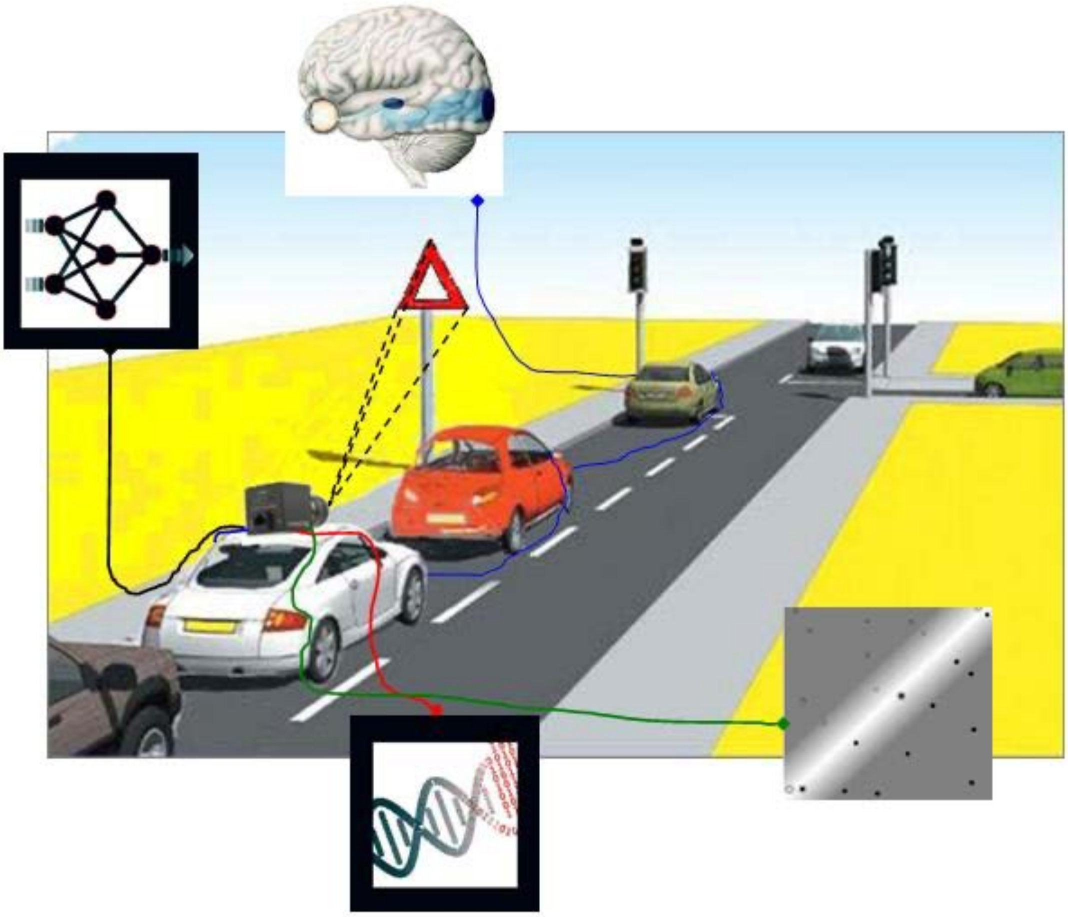
Computational Intelligence in Traffic Sign Recognition



Vincent van Elk

BMI Paper

Supervisor: Prof. Dr. A.E. Eiben

Vrije Universiteit
Faculty of Exact Sciences
Business Mathematics and Informatics
De Boelelaan 1081a
1081 HV Amsterdam

May 2009

Preface

This paper is the final compulsory deliverable of the master study Business Mathematics and Informatics at the Vrije Universiteit in Amsterdam. The main goal of this assignment is to investigate the available literature in reference to a topic that is practically oriented and covers the three aspects business management, mathematics, and computer science.

After seeing a documentary about traffic sign recognition systems utilized within cars my interest in this specific field has grown. Especially once I noticed that the major techniques used in this field belongs to Neural Networks, Support Vector Machines, and Evolutionary computing. These techniques, with a biological background, received my special attention during my study days, because they are generally used in easy to image practical applications. However, the mathematical background is often quite complicated. This also holds for traffic sign recognition systems within cars.

I enjoyed reading and writing about this subject and I would like to thank Gusz Eiben for supervising this research.

Vincent van Elk

Executive summary

Traffic sign detection and recognition is a field of applied computer vision research concerned with the automatic detection and classification or recognition of traffic signs in scene images acquired from a moving car. Driving is a task based fully on visual information processing. The traffic signs define a visual language interpreted by drivers. Traffic signs carry many information necessary for successful driving; they describe current traffic situation, define right-of-way, prohibit or permit certain directions, warn about risky factors etcetera. Traffic signs also help drivers with navigation, and besides that they occur in standardized positions in traffic scenes, their shapes, colours and pictograms are known (because of international standards). To see the problem in its whole complexness we must add additional features that influence the recognition system design and performance. Traffic signs are acquired from car moving on the (often uneven) road surface by considerable speed. The traffic scene images then often suffer from vibrations; colour information is affected by varying illumination. Traffic signs are frequently occluded partially by other vehicles. Many objects are present in traffic scenes which make the sign detection hard. Furthermore, the algorithms must be suitable for the real-time implementation. The hardware platform must be able to process huge amount of information in video data stream. From above problem definition follows, that, to design a successful traffic sign recognition system, one must execute all kind of image processing operations to finally detect, classify, or recognize the traffic signs.

The emphasized techniques in this paper (Support Vector Machines, Neural Networks, and Evolutionary Computing) may help the different image processing operations in classification, clustering, the estimation of statistical distributions, compression, filtering, and so on. Each technique has its advantages and disadvantages and their performance depends on the specific task and problem.

Support Vector Machines are a fairly new development and research showed that it has high classification accuracies and they also have the advantage that they are invariance of orientation, illumination, and scaling. Then again, the selection of the right kernel function is crucial for the overall performance. Neural Network models have received a lot of attention, but they require more attention in dimensionality reduction compared to the two other techniques. However, Neural Networks are very flexible, tolerant to imperfect data, and powerful. Evolutionary Computing can be used in every part of the image processing chain, but the novel algorithms are not fully integrated in the field of traffic sign detection and recognition. A hybrid model through integration of Evolutionary Computing and Support Vector Machines or Neural Networks may overcome the problems which they have to deal with normally. For instance, they can also help in shorten the time it takes to train a Neural Networks or Support Vector Machines. Then again they are not a solution to the limitations of Neural Networks and Support Vector Machines, so best would be to investigate what opportunities they can bring in combination with other methods.

Samenvatting

Verkeersborden detectie en herkenning behoort tot het onderzoek gebied van toegepaste computer vision, betreffende de automatische opsporing en de classificatie of herkenning van verkeersborden in beelden die van een rijdende auto worden verkregen. Het rijden is een taak die volledig is gebaseerd op de visuele informatieverwerking. De verkeersborden vormen een visuele taal die door bestuurders worden geïnterpreteerd. De verkeersborden leveren veel noodzakelijke informatie voor het succesvol rijden; zij beschrijven huidige verkeerssituaties, bepalen het recht van doorgang, belemmeren of het toelaten van bepaalde richtingen, waarschuwen voor gevaarlijke factoren enz. De verkeersborden helpen bestuurders met navigatie, en naast dat komen de verkeersborden in gestandaardiseerde posities in het verkeer voor, de vormen, kleuren en pictogrammen zijn algemeen bekend (wegens internationale normen). Om het probleem in zijn gehele complexiteit te zien moeten wij extra eigenschappen toevoegen die het ontwerp van het herkenningssysteem en de prestaties beïnvloeden. Verkeersborden worden vanuit de auto verkregen die vaak op een ongelijke weg rijdt. De beelden van het verkeer krijgen hierdoor vaak trillingen; de kleur informatie wordt beïnvloed door variërende verlichtingen. De verkeersborden worden vaak gedeeltelijk belemmerd door andere voertuigen. Veel objecten zijn aanwezig in het verkeer die de detectie belemmeren. Daarnaast moeten de algoritmen voor de implementatie in real time kunnen werken. Het hardwareplatform moet reusachtige hoeveelheid informatie van de video gegevensstroom kunnen verwerken. Van bovengenoemd probleem volgt de definitie, dat het ontwerpen van een succesvol herkenningssysteem van verkeersborden, men allerlei soorten handelingen moet verrichten om de verkeersborden te ontdekken, te classificeren of te herkennen.

De benadrukte technieken in dit document (Support Vector Machines, Neural Networks, en Evolutionary Computing) kunnen de verschillende handelingen van het beeldbewerkingproces helpen tijdens de classificatie, clustering, het bepalen van statistische distributies, compressie, filteren, enz. Elke techniek heeft zijn voordelen en nadelen en hun prestaties hangt van de specifieke taak en het probleem af.

Support Vector Machines is een vrij nieuwe ontwikkeling en onderzoek toont aan dat het hoge classificatie nauwkeurigheid heeft en het heeft ook het voordeel dat het niet afhankelijk is van de oriëntatie, licht en schaal. Maar de selectie van de juiste kernel functie is essentieel voor de algemene prestaties. Neural Networks modellen hebben heel wat aandacht gekregen, maar zij vereisen meer aandacht in dimensionaliteit vermindering in vergelijking tot de twee andere technieken. Desondanks, zijn Neural Networks zeer flexibel, tolerant aan onvolmaakte gegevens, en krachtig. Evolutionary Computing kan in elk deel van de keten van de beeldverwerking worden toegepast, maar de nieuwe algoritmen zijn niet volledig geïntegreerd op het gebied van de detectie en de herkenning van de verkeersborden. Een hybride model door integratie van de Evolutionary Computing en Support Vector Machines of Neural Networks kan de problemen overwinnen waar ze normaal mee geconfronteerd worden. Bijvoorbeeld kunnen zij ook helpen met het reduceren van de tijd die het nodig heeft om een Neural Networks of een Support Vector Machines te trainen. Maar ze zijn geen oplossing voor de standaard beperkingen van Neural Networks en Support Vector Machines, dus het beste zou zijn te onderzoeken welke mogelijkheden zij teweegbrengen als ze gecombineerd worden met elkaar.

Contents

1 Introduction

In the last three decades there was an increase of road traffic, although the number of people killed or seriously injured in road accidents has reduced. This indicates that even if our roads are now more overcrowded than ever before, they are safer due the main advances in vehicle design, such as improved crumple zones and side impact bars. This can also be assigned by passive technology, like seat belts, airbags, and antilock braking systems. According to the department for transport [18] the UK road traffic has increased by 70 percent since 1970 and the number of people killed or seriously injured in road accidents has reduced by 52 percent. We can also see in Figure 1 the same trend of traffic accidents in North South Wales in Australia. The fatality rate per 100000 population has declined dramatically over the last three decades. The most recent fatality rate is approximately the same as in 1908, however there are now approximately 27 times more motor vehicles as in 1908.

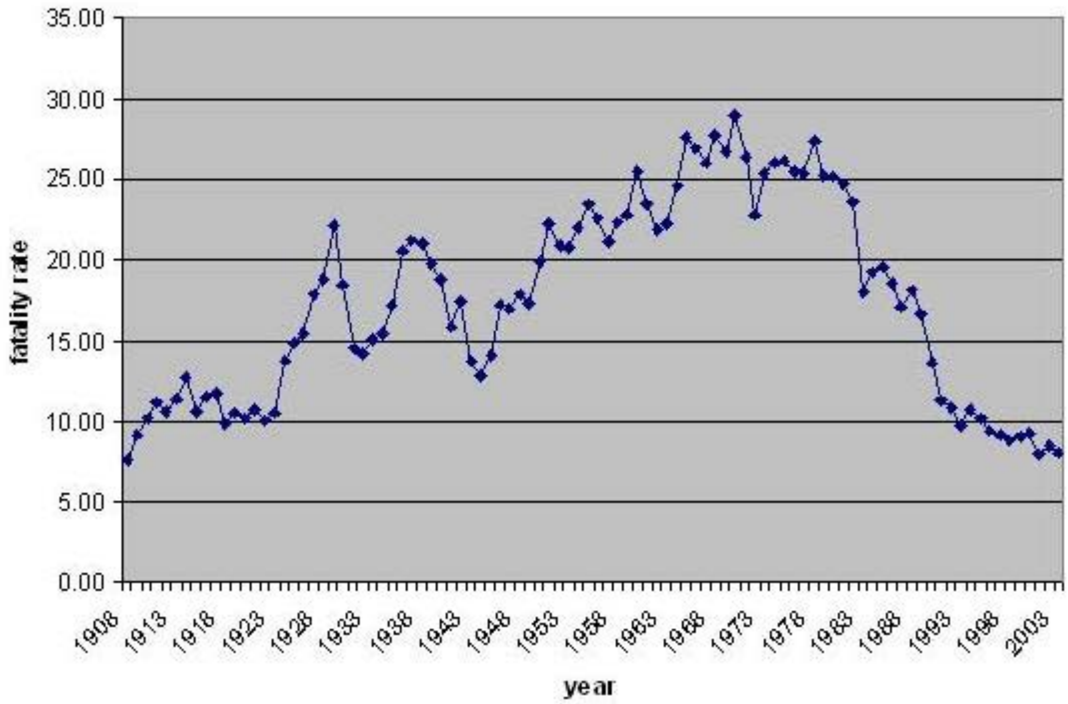


Figure 1 Fatality rate per 100000 population in New South Wales.

Even though there are still thousands of people killed or seriously injured in traffic accidents. Figure 2 presents some findings of a study [67] that compares the cause of accidents in the United States and Great Britain. This diagram shows that only 3 percent of accidents are caused solely by the roadway environment, 57 percent solely by drivers, 2 percent solely by vehicles, 27 percent to the interaction between road environment and drivers, and 3 percent to the interaction between the environment, drivers, and vehicles. In other words, the driver needs more help in his driving process, which should result in an increase of road safety. According to Kopf [47] is the fatality reducing potential of passive technology almost exhausted, and therefore is active technology, like advanced driver assistance system, one of the means in reducing the number of accidents.

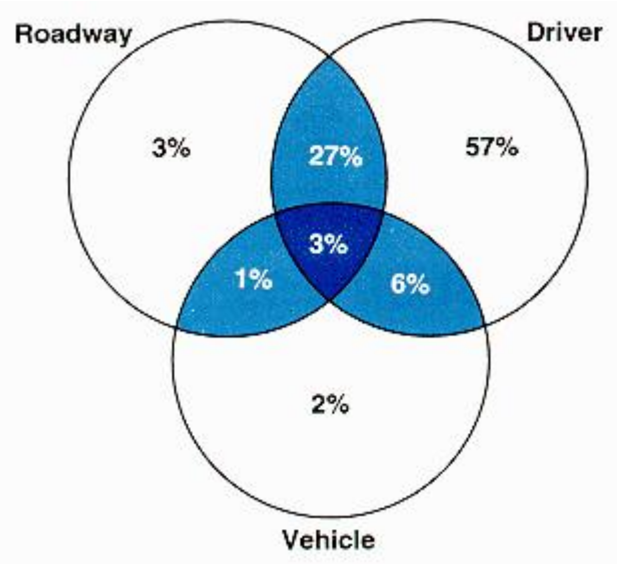


Figure 2 The causes of road accidents in the United States and Great Britain.

Advanced driver assistance system is one of the technologies of Intelligent Transportation Systems (ITS)[1]. ITS consist of a wide range of diverse technologies, which holds the answer to many transportation problems. ITS enables people and goods to move more safely and efficiently through a modern transportation system. One of the most important topics in the ITS field are:

- Advanced Driver Assistance System (ADAS) helps the driver in his driving process.
- Automated highway system is a technology designed to provide for driverless cars on specific rights-of-way.
- Brake assist is an automobile braking technology that increases braking pressure in an emergency situation.
- Dedicated short-range communications offers communication between the vehicle and roadside equipment.
- Floating car date is a technique to determine the traffic speed on the road.

ADAS consists of adaptive cruise control, collision warning system, night vision, adaptive light control, automatic parking, blind spot detection, and traffic sign detection and recognition. The remaining part of this paper focuses on the latter example of ADAS, Traffic Sign Detection and Recognition (TSDR).

1.1 Motivation

In 1968 the Europe countries signed an international treaty, called the *Vienna convention on road traffic*, for the basic traffic rules. More information about the treaty and traffic signs in The Netherlands can be found in Appendix 1. The aim of standardizing traffic regulations in participating countries in order to facilitate international road traffic and to increase road safety. A part of this treaty defined the traffic signs and signals, which results in well standardized traffic signs in Europe. Language differences can create difficulties in

[1] The intelligent transportation society of America was founded in 1991 to promote ITS systems that enhance safety and reduce congestion. More information can be found on the website: http://www.itsa.org.

understanding the traffic signs, therefore are symbols used, instead of words, during the development of the international traffic signs. It is expected that the introduction of the treaty results in traffic signs that can be easily recognized by human drivers.

However, according to a recent survey conducted by a motoring website[2], one in three motorists fail to recognize even the most basic traffic signs. Al-Madani & Al-Janahi [3] also concluded in their study that only 56 percent of the drivers recognized the traffic signs. In other words, the traffic signs are not that easily recognized by human drivers as we first thought[3]. To conclude, a TSDR system that assist the driver can significantly increase driving safety and comfort.

There are also other applications for a system that can detect and recognize traffic signs. For instance, a highway maintenance system that can verify the presence and conditions of traffic signs. Further more, it can be used in intelligent autonomous vehicles. They can function in far greater scope of locations and conditions than manned vehicles.

1.2 Difficulties in detecting and recognizing traffic signs

At first sight the objective of TSDR is well defined and seems to be quite simple. Lets consider a camera that is mounted into a car. This camera captures a stream of images and the system detects and recognizes the traffic signs in the retrieved images. For a graphical view see Figure 3. Unfortunately there are, besides the positive aspects, also some negative aspects.

The positive aspects of TSDR is the uniqueness of the design of traffic signs, colours contrast usually very well against the environment, the signs are strictly positioned relative to the environment and are often set up in a clear sight to the driver.

On the other hand, there are still a number of negative aspects of TSDR. We can distinguish the following aspects:

- Lightning conditions are changeable and not controllable. Lightning is different according to the time of the day and season, weather conditions and local light variations such as direction of light (Figure 4 and Figure 6).
- The presence of other objects like pedestrians, trees, other vehicles, billboards, and buildings. This can cause partial occlusion and shadows. The objects or surrounding could be similar to traffic signs by colour or shape (Figure 5 and Figure 8).
- The sign installation and surface material can physically change over time, influenced by accidents and weather, thus resulting in disoriented and damaged signs and degenerated colours (Figure 7).
- The retrieved images from the camera of a moving car often suffers from motion blur and car vibration.
- It is not possible to generate an offline model of all the possible appearances of the sign, because there are so many degrees of freedom. The object size depends on the

[2] Road sign failure for a third of motorists:
http://www.driver247.com/News/4378.asp
[3] You can check your own knowledge of traffic signs on the following website:
http://www.usa-traffic-signs.com/Test_s/50.htm

distance to the camera. Further more, the camera is not always perpendicular to the signs, which produces an aspect modification.

- The detection and recognition of traffic signs are caught up with the performance of a system in real-time. This requires a system with efficient algorithms and powerful hardware.
- Traffic signs exists in hundreds of variants often different from legally defined standards.

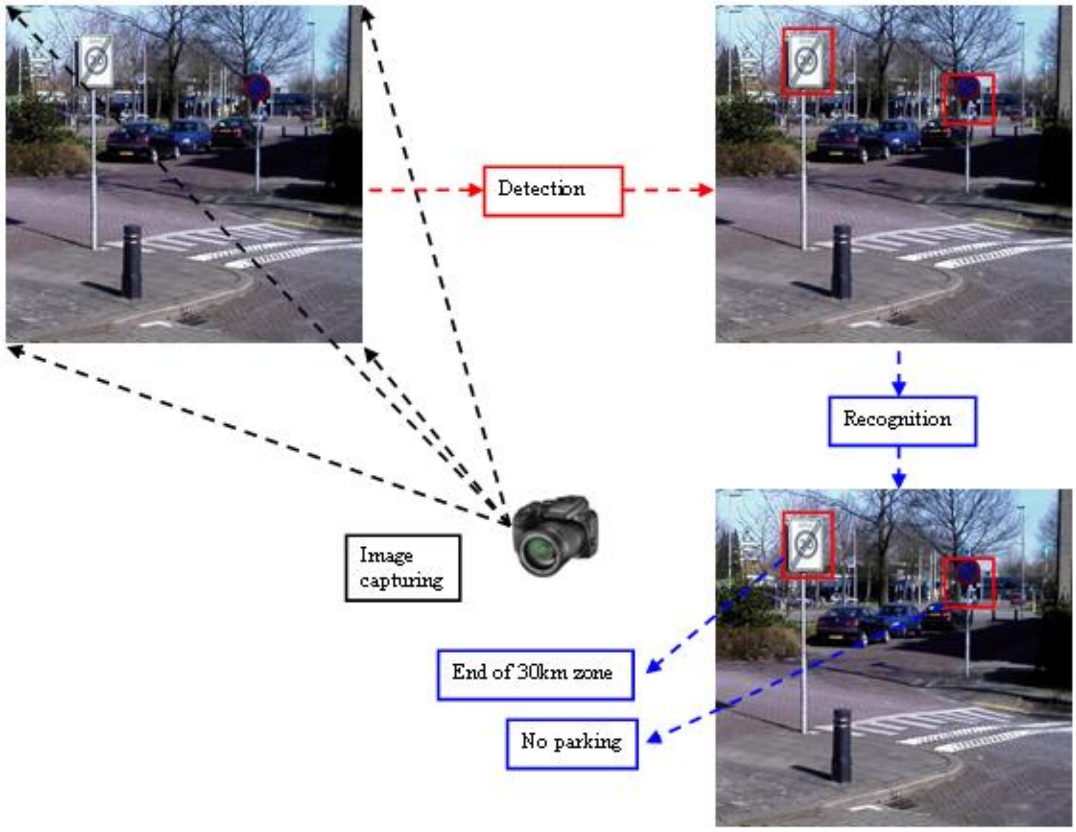


Figure 3 Simple overview of the traffic sign recognition system

Thus, to construct a successful TSDR system one must provide a large number of traffic sign examples to make the system respond correctly to real traffic images. This requires large databases what is expensive and a time consuming task.

Figure 4 Local lightning can make it difficult to recognize traffic signs.


Figure 5 Hard to recognize the blue traffic sign with the blue sky.


Figure 6 Bad weather conditions.

Figure 7 Damaged traffic signs.

Figure 8 Partial occlusion of traffic signs.

1.3 Previous work

The research of TSDR started in Japan in 1984. Since that time many different techniques have been used, and big improvements have been achieved during the last decade. Besides the commonly used techniques there also exist some uncommon techniques like optical multiple correlation. This technique is presented by, the well know trade-mark, P.S.A. Peugeot Citroen and the University of Cambridge.

One of the most important works in this field is described by Estable et al. [27] and Rehrmann et al. [63] research of Daimler-Benz[4] autonomous vehicle VITA-II. Daimler supports the traffic sign recognition research extensively. Its research group also reported papers concerning colour segmentation, parallel computation, and more. The traffic sign recognition system developed by Daimler is designed to use colour information for the sign detection. The recognition stage is covered by various neural networks or nearest neighbour classifiers [82]. The presence of colour is crucial in this system and is unable to operate with weak or missing colour information. Their biggest advantage is the library of 60000 traffic sign images used for system training and evaluation.

The research group at the Faculty of Transportation Sciences in Prague developed a traffic sign recognition system in 1995. They constructed a general framework for the traffic signs by testing various algorithms. The system uses local orientations of image edges to find geometrical shapes which could match with traffic signs. The recognition system has been further developed by Libal [50] into parallel environment of TMS320C80 DSP processors (Texas Instruments) and is capable of real-time operations. The detection phase does not require colour images and works even on badly illuminated scenes. The recognition algorithm is developed by Paclik [59] in the form of combination of statistical kernel classifiers.

[4] In 1926 merged Daimler with Benz and formed Daimler-Benz. Later on, in 1998 Daimler-Benz merged with Chrysler and formed DaimlerChrysler. In 2007, when the Chrysler group was sold off, the name of the parent company was changed to simply Daimler.

1.4 Objectives

The main objective of this paper is the explanation of several techniques, based on computational intelligence, utilized in TSDR systems. Besides that, we also describe the sequence of the executed parts to develop a successful TSDR systems. We can find all different kind of techniques proposed to TSDR, but we emphasize the use of Support Vector Machines (SVM), Neural Networks (NN), and Evolutionary Computing (EC). While the research continued it became clear that the chosen techniques were one of the most widely used in this specific field. Finally, we will give an overview of the researched papers in the field of TSDR.

1.5 Artificial Intelligence versus Computational Intelligence?

The title of this paper can be a little bit confusing, because there is no unifying opinion among researchers which specific methods belong to Artificial Intelligence (AI) and to Computational Intelligence (CI). It is also not clear if AI is a part of CI, or the opposite. Or maybe they are not even parts of each other. Subfields of AI are organized around particular problems, applications, and theoretical differences among researchers. Most researchers threat CI as an umbrella under which more and more methods are slowly added. For instance, Engelbrecht [22] used in his books the following five paradigms of CI: NN, EC, swarm intelligence, artificial immune systems, and fuzzy systems. In contrary, a few published books sponsored by the IEEE computational intelligence society tend to see computational intelligence as "*a consortium of data-driven methodologies which includes fuzzy logic, artificial neural networks, genetic algorithms, probabilistic belief networks and machine learning*" [13]. In general prevails that biological inspiration is a very important factor in CI, but the whole Bayesian foundation of learning, probabilistic and possibilistic reasoning, other alternative methods to handle uncertainty, kernel methods (SVM), information geometry and geometrical learning approaches, search algorithms and many other methods have little to no biological connections. Another problem is where to draw the line; some neural methods are more neural than others.

Duch & Mandziuk [19] analyzed all kind of journals and books about CI and concluded that there is no good definition of this field, because different people include or exclude different methods under the same CI heading. They also noticed that a good part of CI research is concerned with low level cognitive functions (in image processing we refer to low level computer vision[5]): perception, object recognition, signal analysis, discovery of structures in data, simple association, and control. Methods developed for this type of problems include supervised and unsupervised learning by adaptive systems, and they not only include neural, fuzzy, and evolutionary approaches, but also probabilistic and statistical approaches, such as kernel methods (SVM). In contrary, AI is involved with high level cognitive functions: systematic thinking, reasoning, complex representation of knowledge, planning, and understanding of symbolic knowledge. The overlap between these two is quite small. From this point of view AI is a part of CI focussing on problems that require higher cognition (concerned with acquisition of knowledge). All applications that require reasoning based on

[5] Low level computer vision reveals the content of an image. High level computer vision tries to imitate human cognition and the ability to make decisions according to the information contained in the image. The input of computer vision is an image and produces an interpretation as output.

perceptions, such as robotics, autonomous systems, automatic car driving, require methods for solving both low and high level cognitive problems and thus involves techniques from AI and CI. TSDR systems can comprise high level cognitive functions (high level computer vision) if, for instance, the system recognizes a speed limit sign and adjust the speed of the car according to this sign. For simplicity we assume in this paper TSDR systems without high level computer vision.

The intensive research by Duch & Mandziuk on this specific topic is quite recent, and it summarizes the different opinions of many researchers. Therefore we will go with their suggestion. A quotation on page nine of the book by Duch & Mandziuk: "*CI should not be treated as a bag of tricks without deeper foundations. Competition to solve CI problems using approaches developed in other fields should be invited* ". To conclude, according to the intensive research of Duch & Mandziuk we can treat the emphasized methods as CI, because these methods deals with low level computer vision problems in the TSDR system. Thus even SVM, which is rejected by many researchers, can be added to CI for this specific field.

2 Traffic sign detection and recognition system

The identification of traffic signs is usually accomplished in two main phases: detection and recognition. In the detection phase we can distinguish the following parts: pre-processing, feature extraction, and segmentation. As we can see a whole chain of image processing steps are required to finally identify the traffic signs. The first step in the detection phase is pre-processing, which may include several operations. These operations corrects an image which is influenced by noise, motion blur, out-of-focus blur, distortion caused by low resolution, etcetera. Secondly, feature images are extracted from the original image. These feature images containing relevant information of the original image, but in a reduced representation. Thereafter, the traffic signs has to be separated from the background. Meaning that regions of constant features and discontinuities must be identified by segmentation[6]. This can be done with simple segmentation techniques and with the more sophisticated segmentation techniques. After the segmentation phase follows another feature extraction part, but this time based on high level image analysis[7]. In the last part of the detection phase are the potential traffic signs detected from the segmented images, by using the extracted features of the previous part. The efficiency and speed of the detection phase are important factors in the whole process, because it reduces the search space and indicates only potential regions. After detection we can further analyze the image with several operations and modify it or extract further necessary information of it. Thereafter, in the recognition phase, the detected traffic signs can be classified into the necessary categories.

While studying TSDR papers it became clear that there is no general approach in the used chain of the different parts. Some studies leaves out the pre-processing, while others are using all the parts. The studied papers only used two different analyzing approaches for the detection and recognition: shape based analysis and colour based analysis (‘A’ and ‘B’ in Figure 9). These two detection and recognition approaches can be carried out alone or the results of each separate part can be joined together (‘C’ in Figure 9). Fleyeh & Dougherty [30] presented a good overview of different TSDR papers.

To describe each separate part we will use the image processing chain according to the image processing books [21, 37, 44], but we have to remember that the discussed TSDR papers may skip some parts:

- Pre-processing.
- Feature extraction.
- Segmentation.
- Detection.
- Classification and recognition.

[6] In image processing is segmentation the partitioning of a digital image into two or more regions.

[7] Low level image analysis is the same as image processing and high level image analysis is the same as low level computer vision. The input of image processing is an image and produces an image as output, which is the same as low level image analysis. The output of high level image analysis (low level computer vision) reveals the content of an image. Low level image analysis performs local analysis, based on colour distributions, statistics, and anything based on local filtering. High level image analysis performs global analysis, based on image segmentation, Fourrier transform, texture, and pattern.

The input of each part can be pixel based or feature based, therefore represents the arrows pixels or features. The input to the segmentation part is in the studied papers always feature based, therefore we added this part in the image processing chain .

A brief description of each part can be found below, for more details we refer to the Appendix.

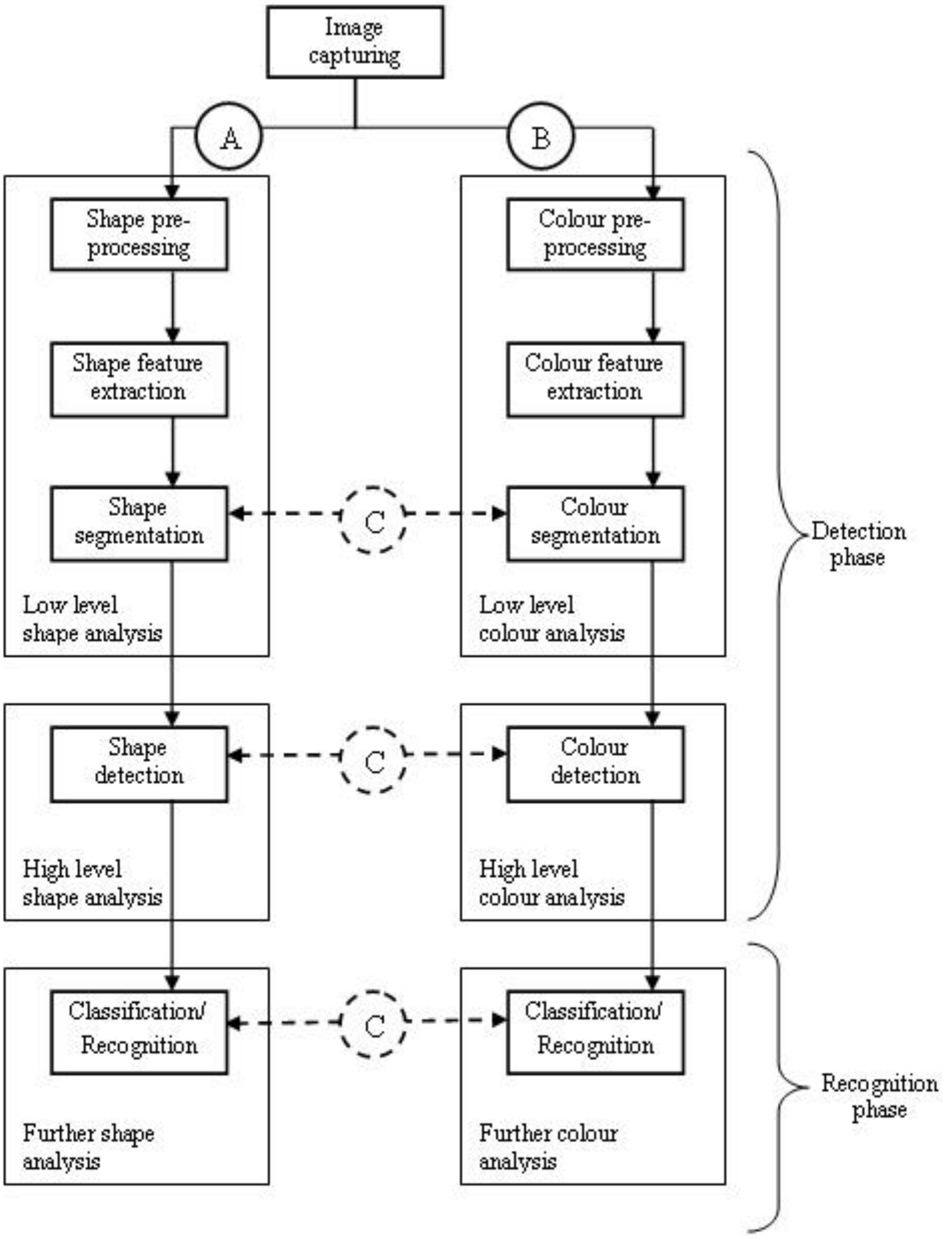


Figure 9 An overview of the traffic sign detection and recognition system. Some parts may be skipped in the discussed papers.

2.1 Detection phase

2.1.1 Pre-processing

The goal of pre-processing is to adjust an image so that the resulting image is more suitable than the original. An image pre-processing method that works for one application may not work very well for another application. The input of the pre-processing part consist of the original (sensor) image and the output is a reconstructed, restored, and enhanced image. The input can be influenced by noise, motion blur, out-of-focus blur, distortion caused by low resolution, etcetera. We can split the image pre-processing methods in two different domains:

- Spatial domain operates directly on the pixels.
- Frequency domain operates on the Fourier transform of an image.

In Appendix 4 we can find an overview of the most common techniques of both domains. These techniques can be found in the reconstruction, restoration, and the enhancement of images. Image reconstruction problems are quite complex and each application needs its own unique approach. During restoration one requires to restore an image that is distorted by physical measurement system. The restoration can employ all information about the nature of the distortions introduced by the system. Unfortunately is the restoration problem ill-posed, because conflicting criteria need to be fulfilled: resolution versus smoothness. The goal of the image enhancement category is to increase specific (perceptual) features. We can find in literature enough papers using neural networks [64] in the pre-processing part of image processing applications. For instance, Adler et al. [2] uses an adaline NN for the reconstruction of images of the human body. Besides NN, we can also find EC [57] in the reconstruction of projections and SVM [51] in image restoration. Unfortunately, there were no papers found using NN, SVM, and EC in the pre-processing part of TSDR systems. The use of these three algorithms appears to be quite successful in a few applications, but the downside can be the performance. We have already explained, especially in the pre-processing part, that the performance is quite crucial to operate a system in real-time. Thus it is suspected that the pre-processing part of TSDR systems is better of with the traditional pre-processing techniques.

2.1.2 Feature extraction

If the input to an algorithm is too large to be processed or there is much data without much useful information, then the input will be transformed into a reduced representation set of features. This transformation is called feature extraction. Its objective is to select the right set of features which describes the data in a sufficient way without loss of accuracy. The set of all possible features represents a feature space. Feature extraction of an image can be classified into three types which are spectral features, geometric features, and textual features. For more information about these specific feature extraction approaches see Appendix 3. Since image data are by nature high dimensional, feature extraction if often a necessary step for segmentation or traffic sign recognition to be successful. Besides the lowered computational costs, it also helps in controlling the so called curse of dimensionality[8]. Some feature extraction approaches were designed to manage explicitly changes in orientation and scale of objects. One of the most generally used feature extraction approaches is principal component analysis. Addison et al. [1] compared the feature extraction capabilities of NN and EC to principal component analysis on different data sets. The results showed that NN and EC performed not as good as principal component analysis, especially NN performed poor. The same results holds if we compare SVM to principal component analysis [46] on different data sets. In contrary, according to Avola et al. [5] does the feature extraction capabilities of NN, SVM, and EC performs quite good in image feature extraction. This also indicates that the used approach depends on the specific application.

2.1.3 Segmentation

Segmentation refers to operations that partitions an image into regions that are consistent with respect to some conditions. The goal of segmentation is to simplify or change the representation of an image into something that is more meaningful or easier to analyze. The basic attribute for colour segmentation is image luminance amplitude for a monochrome image and colour components for a colour image. Image shape and texture are also useful attributes for segmentation. The pre-processing and feature extraction parts may help in reducing the difficulties of these image segmentation problems. Image segmentation approaches can be based directly on pixel data or on features, which one to prefer depends on the specific application and/or problem. The study of Ozyildiz et al. [58] shows that combining shape and colour segmentation has advantages over the use of each segmentation approach alone. An overview of the most widespread segmentation methods can be found in Appendix 5. Furthermore, segmentation does not involve classifying each segment. This part only subdivides an image; it does not attempt to recognize the individual segments or their relationships to each other. There is no general solution to the image segmentation problem, this is because there is not a single measure that clearly tells the segmentation quality. It is therefore hard to tell what the best used segmentation method is for a specific application

2.1.4 Detection

The segmentation part provide us with potential regions of traffic signs. The goal of the detection part is the identification of these potential regions with the use of rules that accept or reject a potential region as a traffic sign candidate. There also exist two different approaches in the traffic sign detection part: colour based and shape based. Based on the segmentation results, shape analysis is in general applied to these results in order to perform the detection of the traffic signs. Most authors share a common sequence of steps during the process. This sequence has a drawback; regions that have falsely been rejected by the colour segmentation, cannot be recovered in the further process. A joint modelling of colour and shape analysis can overcome this problem. However, many studies [30] showed that the detection can be achieved even if either of the colour or the shape is missing. For example, Figure 10 illustrates

[8] The curse of dimensionality is a property of classification and regression problem. The higher the dimension of the feature space leads to an increased number of parameters to be estimated.

how the traffic sign is detected with both approaches. We will take a closer look at both analyzing approaches below.

2.1.4.1 Colour based analysis

Colours can be an important source of information in TSDR systems. A camera mounted on a car produces an RGB image. This image is in most cases not suitable for detection, because any variation in ambient light intensity affects the RGB system by shifting the clusters of colours towards the white or the black corners. Therefore most colour based detection systems use colour space conversion. In other words, the RGB image is converted into another form that simplifies the detection process. There are many colour spaces available in the literature, among them are the HIS, HSB, L*a*b, YIQ and YUV colour systems. A few rely solely on grey scale data as it was thought that colour based analysis is absolutely unreliable. The majority of recently published sign detection approaches make use of colour information. Approximately 70 percent of colour analysis approaches used the hue as standard colour dimension[9], while the remaining 30 percent used other colour spaces. Colour analysis becomes easier if it is only applied on the hue value and not on the three RGB values. In comparison to RGB, is the hue value also insensitive to variations in ambient light. However, the hue is not appropriate for grey-scale analysis, because it has a constant level along the grey-scale axis. There are simple colour analyzing techniques, which are very fast and suitable for real-time applications. They are less accurate compared to complex techniques like fuzzy or NN based, but are computationally costly. This shows there is no standard procedure to analyse colours from the image under consideration.

2.1.4.2 Shape based analysis

Regardless of the broad use of colours in TSDR systems, it can also be done by using shapes. It is provided by many research groups that it is enough to use shapes of traffic signs to detect or recognize them. One of the points supporting the use of shape information for traffic sign detection and recognition is the lack to standard colours among the countries. Systems that rely on colours has to change their configuration by moving from one country to another. Another point is the fact that colours vary as daylight and reflectance properties changes. Hibi [39] showed that 93 percent of the signs could be successfully detected in bad light conditions, compared to 97 percent in good lightning conditions. Thus during sunset and night, shape detection will be a good alternative. Unfortunately also shape based detection and recognition has its own specific difficulties. Their may exist similar objects to the traffic sign in the scene like mail boxes, windows and cars. Traffic signs may appear damaged, occluded by other objects and disoriented. When the sign is very small, it will be unrecognizable. When the viewing angle is not head-on, the aspect ratio may also change. Working with shapes necessitates robust edge detection and matching algorithm. This is difficult when the traffic sign appears relatively small in the image.

[9] Hue is one of the dimensions of the HSV colour space. The two others are saturation and brightness.

Figure 10 Colour analysis is used for image segmentation, followed by shape analysis for detection.

2.2 Recognition phase

The output of the detection phase is a list of detected traffic sign. This list is forwarded to the recognition phase for further evaluation. To design a good recognizer, many features should be taken into account. Firstly, the recognizer should present a good discriminative power and low computational cost. Secondly, it should be robust to the geometrical status of the sign, such as the vertical or horizontal orientation, the size, and the position of the sign in the image. Thirdly, it should be robust to noise. Fourthly, the recognition should be carried out quickly if it is designed for real time applications. Furthermore, the recognizer must be able to learn a large number of classes and as much as possible a priori knowledge about traffic signs should be employed into the classifier design.

2.2.1 Further analysis

Several qualitative and quantitative techniques have been developed for characterizing the shape and colour of traffic signs within an image. These techniques are useful for classifying traffic signs in the TSDR system. In other words, the detected traffic signs are represented in another form such that the recognition of traffic signs becomes easier. There exist two different approaches in traffic sign analysis: colour based and shape based. Based on the segmentation and detection results, shape analysis is in general applied to these results in order to perform the recognition of the traffic signs. Most authors share a common sequence of steps during the process. This sequence has a drawback; regions that have falsely been rejected by the colour segmentation, cannot be recovered in the further process. A joint modelling of colour and shape analysis can overcome this problem. However, many studies showed that the detection and recognition can be achieved even if either of the colour or the shape is missing.

2.2.2 Classification and recognition

Classification and recognition are complementary tasks that lie at the end of the image processing chain. Classification is concerned with establishing criteria that can be used to identify or distinguish different populations of objects that appear in an image. Recognition is the process by which these tools are subsequently used to find a particular feature within an image. They include different processes, such as finding a traffic sign in an image, or matching that traffic sign to a specific traffic sign.

Once an image is detected and further analyzed, the next task is to classify or recognize the detected objects in the scene. Hence, the objective of pattern recognition is to classify or recognize objects in the scene from a set of measurements of the objects. A set of similar objects possessing more ore less identical features are said to belong to a certain pattern class. We can see in Appendix 3 that there are many types of features and each feature has a specific technique for measurement. The selection and extraction of appropriate features from patterns is the first major problem in pattern recognition. We can find in the literature of TSDR that in most systems the recognition is based on pixel data. The recognition based on features is less frequent used. Besides that, we also noted that there is a wide use of NN in the recognition of traffic signs. There is also enough literature available of SVM and EC in traffic sign recognition. In the remaining chapters we will discuss them in more detail.

3 Support vector machine

SVM are supervised learning algorithm, which demonstrates reliable performance in tasks like pattern recognition and regression. Supervised learning is a machine learning technique for learning a function from training data. The training data consist of pairs of input objects and desired outputs. The output of the function can be a continuous value (regression), or can predict a class label of the input objects (classification). The task of the supervised learner is to predict the value of the function for any valid input object after having seen a number of training examples. The most widely used supervised learning approaches are NN, SVM, k-Nearest Neighbours, Gaussian Mixture Model, Naïve Bayes, Decision Tree, and Radial Basis Function. A broadly used classifier in the field of TSDR is SVM.

3.1 SVM algorithm

SVM is based on two key elements: a general learning algorithm and a problem specific kernel that computes the inner product of input data points in a feature space. A SVM performs classification by constructing an *N*-dimensional hyper plane that optimally separates the data into two categories. SVM models are closely related to neural networks. In fact, a SVM model using a sigmoid kernel function is equivalent to a two-layer perceptron neural network. The input space is mapped by means of a non-linear transformation into a high dimensional feature space (Figure 11).

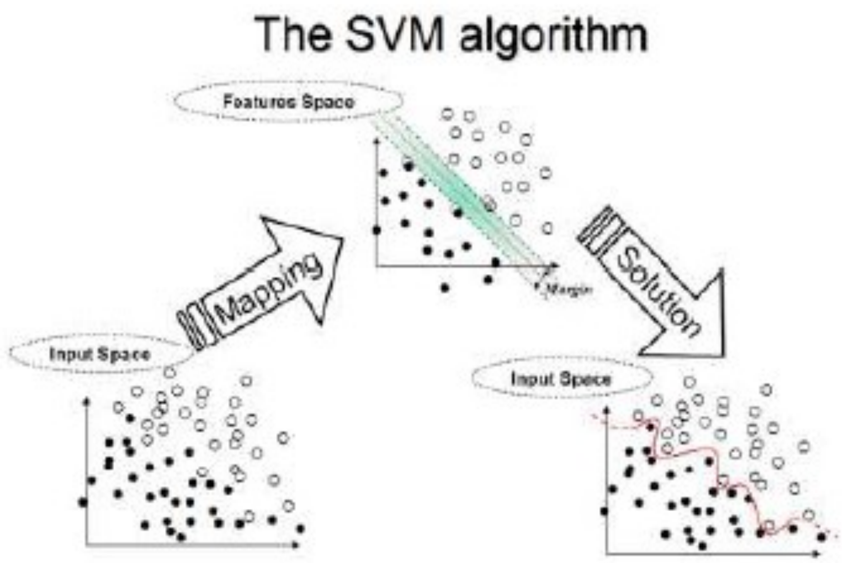


Figure 11 An overview of the support vector machine process.

The goal of SVM modelling is to find the optimal hyper plane that separates the data sets in such a way that the margin between the data sets is maximized. The vectors near the hyper plane are the support vectors (Figure 12). In other words the decision boundary should be as far away from the data of both categories as possible.

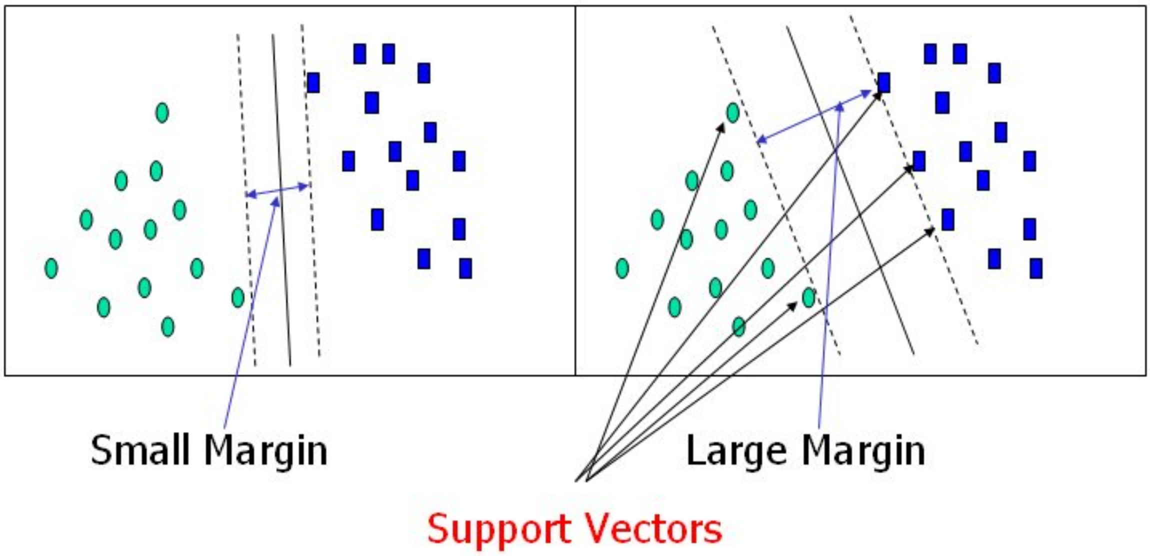


Figure 12 The left picture separates the two categories with a small margin. The right picture has a maximized margin between the two categories, which is the goal of SVM modelling.

The simplest way to divide two categories is with a straight line, flat plane or an N-dimensional hyper plane. This can unfortunately not been done with the two categories of Figure 13.

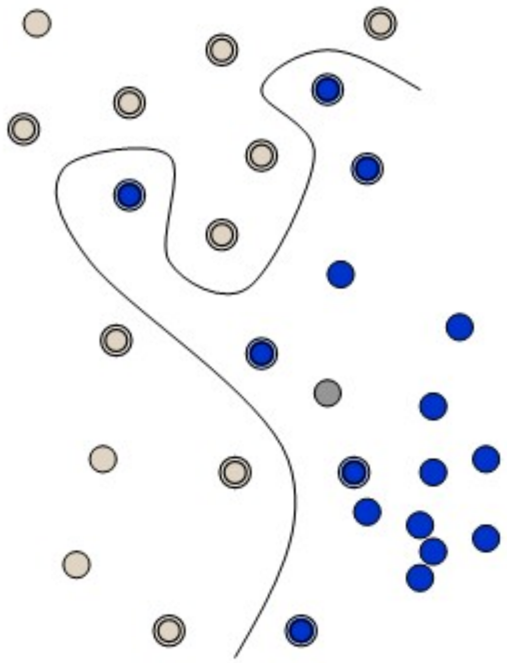


Figure 13 An example of non-linear separation.

To overcome this problem, the SVM uses a kernel function to map data into a different space where a hyper plane can be used to do the separation. The kernel function transforms the data into a higher dimension space to make it possible to perform the separation (Figure 14). There are a lot of different kernel function, used for a wide variety of applications.

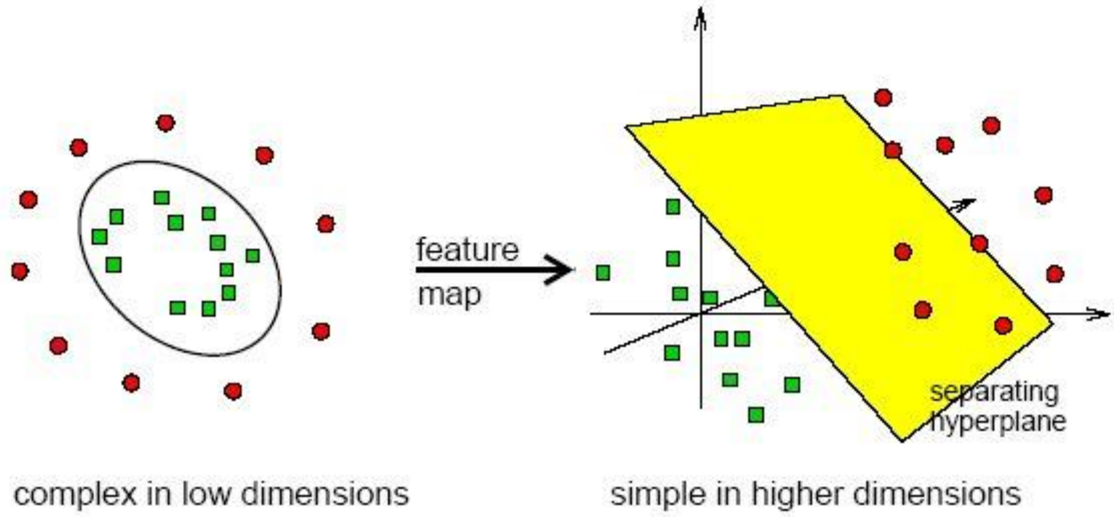


Figure 14 Separation in a higher dimension.

The SVM algorithm consists of two stages:

- Training stage: training samples containing labelled positive and negative input data to the SVM. This input data can consist of distance to border vectors, binary images, Zernike moments, and more. Each input data is represented by vector $\mathbf{x}_i$ with label $y_i = \pm 1$, $1 \le i \le l$, l is the number of samples. The decision boundary should classify all points correctly, thus $y_i\left(\mathbf{w}^{\mathbf{T}}\mathbf{x}_i + b\right) \ge 1,\ \forall i$. The decision boundary can be found by solving the following constrained optimization problem: $\mathit{Minimize}\ \frac{1}{2}\|\mathbf{w}\|^2 \quad \mathit{subjectto}\ y_i\left(\mathbf{w}^{\mathbf{T}}\mathbf{x}_i + b\right) \ge 1\ \forall i$. The Lagrangian of this optimization problem is: $L = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_i \alpha_i\left(y_i\left(\mathbf{w}^{\mathbf{T}}\mathbf{x}_i + b\right) - 1\right) \quad \alpha_i \ge 0\ \forall i$. The optimization problem can be rewritten in terms of α_i by setting the derivative of the Lagrangian to zero: $\mathit{Maximize}\ W(\alpha) = \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \quad \mathit{subjectto}\ \alpha_i \ge 0,\ \sum_i \alpha_i y_i = 0\ \forall i$

This quadratic programming problem is solver when: $\alpha_i\left(y_i\left(\mathbf{w}^{\mathbf{T}}\mathbf{x}_i + b\right) - 1\right) = 0 \quad \forall i \; \mathbf{x}_i$ with $\alpha_i > 0$ are support vectors. This is for a linear separable problem, for more details about the non-linear problem see Appendix .

- Testing stage: the resulting classifier is applied to unlabeled images to decide whether they belong to the positive or the negative category. The label of x is simply obtained by computing: $\mathbf{w} = \sum_{j=1}^{s} \alpha_{t_j} y_{t_j} \mathbf{x}_{t_j}$ with $t_j\left(j = 1, \ldots, s\right)$ the indices of the s support vectors. Classify $\mathbf{z}$ as category 1 if the sum is positive, and category 2 otherwise.

The question is, how to apply SVM into TSDR systems? SVM is most widely used in the classification and recognition part of TSDR systems. Lets consider the classification of the detected traffic signs. For example, classify traffic signs belonging to a circular or a rectangular shape. The key consists of finding the optimal decision frontier to divide these two categories. The optimal election will be the line that maximizes the distance from the frontier to the data. In multiple categories the frontier is a hyper plane. For instance, Lafuente-Arroyo et al. [48] used SVM, with bounding box as feature extraction approach, to classify the traffic signs by their shapes. The showed results were successful and it was invariance against rotations. The same can be done to recognize the specific traffic signs, but with other feature extraction approaches. Unfortunately, there were no TSDR papers found that used SVM in other parts besides detection, classification, and recognition. This is also not really surprising, because SVM is a supervised learning task which deals with regression and classification. However, it can also be used to retrieve the best set of image features. This part is normally called feature selection, but this is out of the scope of this paper

3.2 Advantaged and disadvantages of SVM

One of the advantages of SVM over other learning algorithms is that it can be analyzed theoretically using concepts from computational learning theory, and at the same time can achieve good results when applied to real problems. In the absence of a local optimal, training SVM is relatively easy compared to NN. SVM has been tested in lots of applications. Camps-Vals et al. [11] tested SVM against NN and showed that it was unfeasible to train a NN while working in high dimensional input space as compared to SVM which deals the problem in higher dimensional space. The tradeoffs between classifier complexity and error can be controlled clearly. However, the kernel function is one very important factor to the performance of the SVM. Selection of a suitable kernel function for a specific problem can improve the optimal performance of the SVM. In most cases this has to be done by hand, which is a time consuming job.

3.3 SVM papers

- Gil-Jimenez et al. [34] created a traffic sign image database test set that can be used to evaluate traffic sign detection and recognition algorithms. They developed two different methods for the detection and classification of traffic signs according to their shape. The first method is based on distance to borders measurement and linear SVM. The other is based on a technique called FFT [35]. In the segmentation part potential regions are extracted from the scene by thresholding using the hue and saturation dimensions of the HSV colour space. After the segmentation part the regions are classified into their shapes with the use of linear SVM. They used linear classification, because of its low computational cost. The input of the linear SVM consist of distance to border vectors, which has the advantage that it is robust to translations, rotations and scale. Table 1 shows the result for all categories. The first thing to notice is the successful classification of the traffic signs. On the other hand, there are also a high number of false alarms. This can be clarified by some extracted noisy regions, which are classified as potential regions by their shape. However, the loss probability is high especially in the categories different sizes and occlusion. This can be explained by the very high distance of the traffic signs from the camera and the rejection of traffic signs by a difficult segmentation mask. To conclude, the classification of the traffic signs works good, but there is need for other measures in extracting potential regions.

Table 1 Results for every category

Number Images	Category	Sub-category	Classification. Success	False Alarms	Loss Prob.
30	Dif. Shapes	Circular	41/41	43	22.23%
30	Dif. Shapes	Octagonal	33/34	49	11.2%
30	Dif. Shapes	Rectangle	33/35	78	8.11%
30	Dif. Shapes	Triangular	61/62	101	28.28%
40	Dif. Signs	-	53/54	91	17.25%
40	Dif. Positions	-	73/75	116	26.32%
30	Rotation	-	32/32	88	29.27%
37	Occlusion	-	45/46	116	47.62%
40	Dif. Sizes	-	37/38	74	50.95%
23	Deter. Signs	-	42/44	92	25%

- Simon et al. [72] have also build a traffic sign image database test set to evaluate the traffic sign detection and recognition algorithms. With the use of the SVM algorithm they created a classification function associated to the traffic sign of interest. The SVM algorithm uses the triangular kernel function. This way they can detect if a potential region belongs to the searched traffic sign. The model outperforms the earlier studied saliency model, but it needs a lot of manual configurations. For example, the choice of the right kernel function required a lot of experiments. Simon et al. [71] studied also the degree to which an object attracts attention compared to its scene background. They also made use of the SVM algorithm and came to the same conclusion as before; SVM performs better than the earlier studied models like the salience model.

- Shi et al. [69] presents an approach to recognize Swedish traffic signs by using SVM. The features binary image and Zernike moments are used for representing the input data of the SVM for training and testing. They also experimented with different features and kernel functions. They achieved a 100 percent accuracy in classifying shapes and a 99 percent accuracy in classifying speed limit signs.

- Gilani [33] presents in his paper an extension of the earlier work done by P.M. Doughtery at the ITS research platform of the Dalarna university. The focus of the paper is the extraction of invariant features. These invariant features are used as the input of a SVM, which performs the classification of the traffic signs. First the images are converted to the HSV colour space, thereafter they performed segmentation based on dynamic-threshold method, seeded region growing method, and the minimum-maximum method. The output of the segmentation phase is normalized to standardize the size of the potential regions, irrespective of the size in the original image. The methods for extraction of the invariant features are: Haar invariant features, effective invariant FT coefficients, geometric moments and orthogonal Fourier-Mellin moments. More details about these methods can be found in respective paper. The kernel function consist of a linear classifier. The results of the SVM with the different extraction methods are shown in Table 2. We can conclude that, besides the selected kernel function, also the extraction methods are quite important.

Table 2 Results of different feature extraction methods

feature extraction method	shape recognition accuracy	Speed-limit recognition accuracy
Haar features	97.77%	96.00%
Effective FT coefficient	99.04%	90.67%
Orthogonal Fourrier-Mellin	92.22%	50.67%
Geometric moments	92.22%	62.67%

- Shi [68] used the features binary representation and Zernike moments to achieve the pattern recognition that is irrespective of image size, position and orientation. The objective consists of the recognition of traffic sign shapes and speed limit signs. The results shown in Table 3 and Table 4 that the SVM recognition model with Zernike moments does not work as good as the SVM recognition model with binary representation. The linear kernel function also shows the highest correct classification rate. Just like in the previous works, we can conclude that the feature extraction method is just as important as the kernel function.

Table 3 Results of different kernel functions with binary representation

	Correct classification rate	
Kernel function	traffic sign shapes	speed limit signs
Linear	100%	98%
Polynomial	97.86%	96%
RBF	100%	97%
Sigmoid	99.29%	97%

Table 4 Results of different kernel functions with Zernike moments

	Correct classification rate

Kernel function	traffic sign shapes	speed limit signs
Linear	100%	82%
Polynomial	85.83%	56%
RBF	99.17%	72%
Sigmoid	99.17%	68%

- Maldonado-Bascon et al. [56] used the HIS colour space for chromatic signs and extracted the potential traffic signs by thresholding. A linear SVM is used to classify the potential traffic signs into a shape class and finally the recognition is done based on SVM with Gaussian kernels. Different SVMs are used for each colour and shape classification. The results can be found in Table 5 and shows that all signs have been correctly detected in each of the five sequences. The situation of confused recognition can be attributed to long distances from the traffic signs to the camera or to poor lightning. Moreover, the system is invariant to rotations, changes of scale, and different positions. In addition, the algorithm can also detect traffic signs that are partially occluded.

Table 5 Summary of results

Number of sequence	1	2	3	4	5
Number of images	749	1774	860	995	798
Number of traffic signs	21	21	20	25	17
Detections of traffic signs	218	237	227	285	127
Noisy potential traffic signs	601	985	728	622	434
False alarm	0	3	4	8	7
Confused recognition	4	4	4	2	7

- In another work of Gil-Jimenez et al. [36] we can find a new algorithm for the recognition of traffic signs. It is based on a shape detector that focuses on the content of the traffic sign to perform the recognition of traffic signs. The recognition is done by a SVM. The results illustrate that the success probability is not good enough for categories with a small number of samples, whereas for categories with enough number of samples are satisfactory, which makes the overall success probability acceptable. The study did not focus enough on the segmentation step, which is quite crucial for the correct operation of the whole system.

- Silapachote et al. [70] detected signs using local colour and texture features to classify image regions with a conditional entropy model. Detected sign regions are then recognized by matching them against a known database of traffic signs. A SVM algorithm uses colour to focus the search, and a match is found based on the correspondence of corners and their associated shape contexts. The SVM classifier has a 97.14 percent accuracy and 97.83 percent for the matcher.

- Zhu & Liu [83] applied a SVM network for colour standardization and traffic sign classification. The colour standardization technique maps the 24-bit bitmap into a single space of five elements, which significantly simplifies the complexity of the traffic signs' colour information and is more suitable for the traffic sign classification.

SVM is applied to the standardized colour traffic signs, which shows good results for the accuracy of the classification.

3.4 Overview

One of the first things we noticed in the researched papers is that the SVM algorithm is mainly used in the feature extraction, detection, classification, and recognition part. This is caused due the supervised learning task of the SVM algorithm. Maybe it is possible in the future to integrate the other parts into the image processing chain, but the explicit research area has not been evolved to this stage yet. Nevertheless, the performance of SVM is quite good, because they deal with the problem in a higher dimension

One of the disadvantages of SVM is to pick the right kernel function. This also fit in with the research of Shi [68]; there is a big difference in the correct classified traffic signs with different kernel functions. This is also confirmed by the research of Addison et al. [1]. Another disadvantage, which holds for all classification methods, is the extraction of useful features that functions as input to the classification and recognition system. This can be pixel based or feature based, but it needs the right information to classify the traffic signs in a correct way. Gilani [33] and Gil-Jimenez et al. [34] also shows in their studies that the selection of the right features is quite important. At last, the data set that functions as input to the SVM must not be to small. This is very apparent in the work of Gil-Jimenez et al. [36] and Maldonado-Bascon et al. [56].

To conclude, the SVM algorithm works quite good in the classification and recognition part, but the selection of the right kernel function and extracted features is crucial for the correct classification rate. Besides that, SVM is able to perform quite good in high dimensional input space, like images, compared to NN. Finally, one of the major advantages is the invariance of orientation, illumination, and scaling. Besides that it is also, according to Silapachote et al. [70] research, able to detect non standard text, which is an important factor in the recognition stage.

4 Neural network

Artificial neural network, usually called NN, applications has recently received considerable attention. The methodology of modelling, or estimation, is somewhat comparable to statistical modelling. NN should not be thought as a substitute for statistical modelling, but rather as an different approach to fitting non-linear data. NN is a computational model based on biological NN. It consists of an interconnected group of artificial neurons and processes information using a connectionists[10] approach to computation. In most cases a NN is an adaptive system that changes its structure based on internal and external information that flows through the network during the learning phase. In more practical terms NN are non linear statistical data modelling tools. They can be used to model complex relationships between inputs and outputs or to find patterns in data. The powerful side of NN is its ability to solve problems that are very hard to be solved by traditional computing methods. Usual computers apply algorithmic approaches, if the specific steps that the computer needs to follow are not known, the computer can not solve the problem. That means that traditional computing methods can only solve the problems that we have already understood and knew how to solve. However, NN are, in some way, much more powerful because they can solve problems that we do not exactly know how to solve. That is why the recent wide spread use of NN in areas like, virus detection, robot control, intrusion detection systems, pattern recognition (image, fingerprint, noise, etcetera.), and so on.

4.1 NN model

A NN is an information processing theory that is inspired by the way biological nervous systems, such as the brain, process information. The key element of this theory is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) working in harmony to solve specific problems. A NN is configured for a specific application, such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons. This also holds for NN[11]. See Figure 15 for a comparison between human and artificial neuron.

The particular learning tasks to which NN are applied, tend to fall within three major learning paradigms. These are supervised learning, unsupervised learning, and reinforcement learning. We have already explained supervised learning in chapter 3. Unsupervised learning is a class of problems in which one seeks to determine how the data are organized. It differs from the other two learning paradigms in that the learner is given only unlabeled examples. Tasks that fall within the paradigm of unsupervised learning are in general estimation problems, like clustering, the estimation of statistical distributions, compression, and filtering. Commonly used unsupervised learning algorithms among NN are Self Organizing Maps, and Adaptive

[10] Connectionism is a set of approaches in the fields of artificial intelligence, cognitive psychology, cognitive science, neuroscience and philosophy of mind, that models mental or behavioral phenomena as the emergent processes of interconnected networks of simple units. There are many forms of connectionism, but the most common forms use neural network models. More information about connectionism can be found on the following website: http://neuron-ai.tuke.sk/NCS/VOL1/P3_html/vol1_3.html

[11] A typical neural network may have a hundred neurons. In comparison, the human nervous system is believed to have about 3×10^{10} neurons. Thus, from this point of view, it is quite hard to compare these two.

Resonance Theory. Reinforcement learning is concerned with how an agent should take actions in an environment while maximizing some notion of long-term reward. It differs from the two other learning paradigms in that correct input and output pairs are never presented. Tasks that fall within this learning paradigm are control problems, games, telecommunications, and sequential decision making tasks. NN is frequently used in reinforcement learning as part of a overall algorithm. We can distinguish the following, commonly used, types of NN:

- Feed forward NN.
- Radial basis function network.
- Self organizing map.
- Recurrent network.
- Stochastic NN.
- Modular NN.
- Associative NN.

A NN with a supervised learning task aims at minimizing the error, thus the difference between the real output and the output generated by the network. For this it computes the output and compares this with the desired output. As long as the error found does not meet the demands (which can be pre-specified), the network will continue learning by updating its weights. This updating can be done in several ways, depending on (amongst other parameters) the learning algorithm and the network architecture. A supervised learning task, like pattern recognition, can be implemented by using a feed forward NN that has been trained accordingly. In a feed forward network information always moves one direction; it never goes backwards. During training, the network is trained to associate outputs with input patterns. When the network is used, it identifies the input pattern and tries to output the associated output pattern. The power of NN comes to life when a pattern that has no output associated with it, is given as an input. In this case, the network gives the output that corresponds to a taught input pattern that is least different from the given pattern A neuron can be described by:

- a set of links that describe the neuron inputs, with weights w_1 , w_2 , ...,w_m.
- a linear combiner for computing the weighted sum of the inputs:

  $$u = \sum_{j=1}^{m} w_j x_j$$
- and an activation function φ for limiting the amplitude of the neuron output $y = \varphi(u+b)$, where b is the bias

A neuron receives a number of inputs from the data and produces one output. Each input comes via a connection that has a strength (or weight); these weights correspond to synaptic efficiency in a biological neuron. Each neuron also has a single threshold value. The weighted sum of the inputs is formed, and the threshold subtracted, to compose the activation of the neuron[12]. The activation signal is passed through an activation function (also known as a transfer function) to produce the output of the neuron.

[12] Also called postsynaptic potential: http://en.wikipedia.org/wiki/Postsynaptic_potential

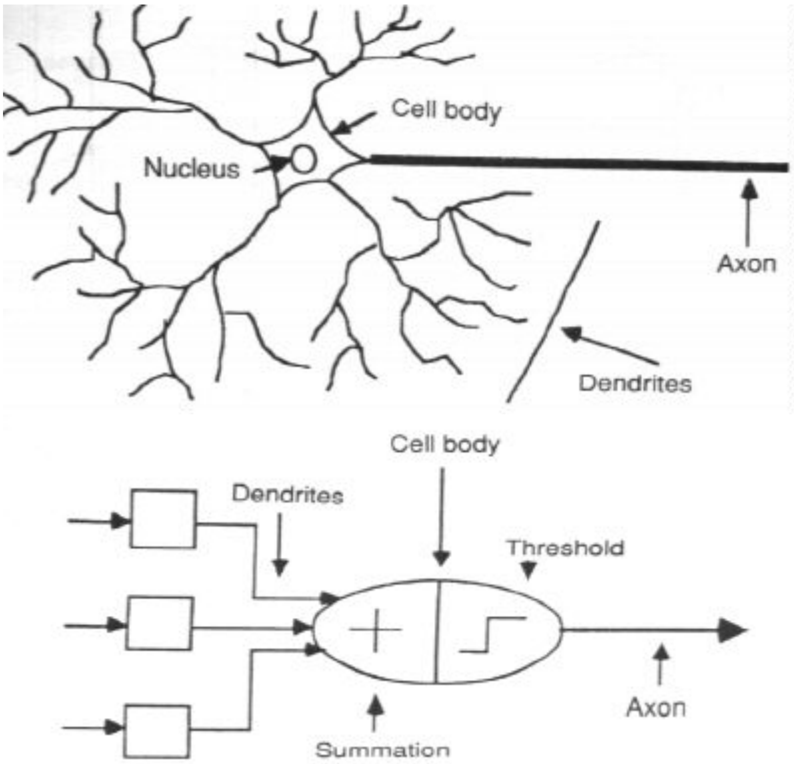


Figure 15 The upper picture illustrates a human neuron and the lower one an artificial neuron.

The simplest kind of a feed forward NN is the single layer perceptron network, which is just a linear classifier. The inputs feed directly the outputs via a series of weight. A multi layer perceptron has a feed-forward structure if the signal flow is forwarded from the input to the hidden units, and after that forwarded to the output units. The input layer consists of units which simply serve to introduce the values of the input variables. The hidden and output layer neurons are each connected to all of the units in the preceding layer. See, for example, Figure 16. When the network is executed (used), the input variable values are placed in the input units, and then the hidden and output layer units are progressively executed. Each of them calculates its activation value by taking the weighted sum of the outputs of the units in the preceding layer, and subtracting the threshold. The activation value is passed through the activation function to produce the output of the neuron. When the entire network has been executed, the outputs of the output layer act as the output of the entire network. The classical learning algorithm of a feed forward NN is based on the gradient descent method, and this method requires a function of the weights that is continuous and differentiable everywhere.

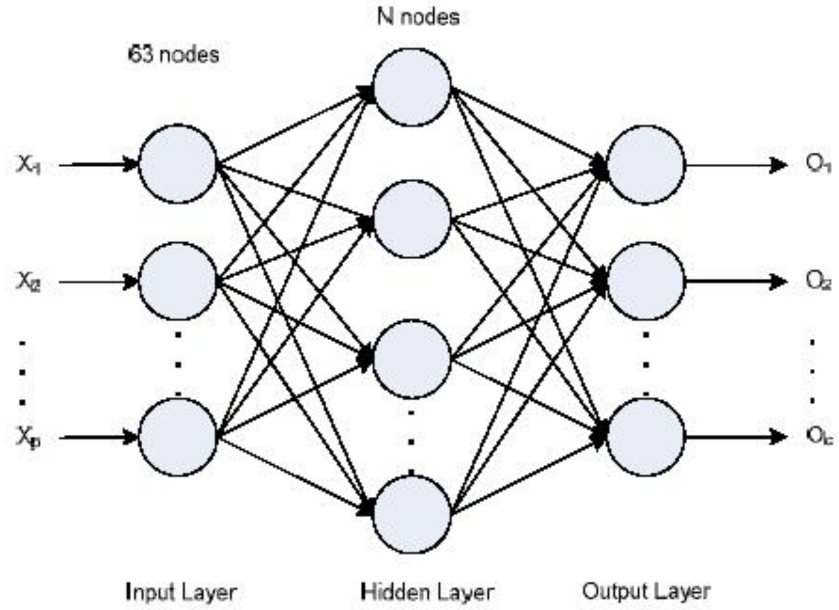


Figure 16 Multi layer perceptron structure with 63 input nodes.

Like we marked above, we can also use NN in the unsupervised and reinforcement learning tasks. A detailed description for the implementation of these two tasks can be found in the book of Freeman & Skapura [32]

4.2 Advantages and disadvantages of NN

Their ability to learn by example makes them very flexible, tolerant to imperfect data, and powerful. Furthermore there is no need to create an algorithm in order to perform a specific task; thus there is no need to understand the internal mechanisms of that task, which result in the applicability to a wide range of problems. They are also very well suited for real time systems because of their fast response and computational times, which are due to their parallel architecture. This is a major advantage in TSDR systems. Perhaps the most exciting aspect of NN is the possibility that some day conscious networks might be produced. The TSDR system can then be extended with extra functionality, like adjusting the speed of the car according to the speed limit sign. Integrating NN with EC and other CI methods will bring the best out of them.

One of the disadvantages of NN, just like SVM, is the large sample size to produce successful results. Minimizing overfitting[13] requires a great deal of computational effort and finding a local optimum. A specific image processing problem is how one should incorporate prior knowledge into pattern recognition techniques. At last, the individual relations between the

[13] Overfitting is fitting a model that has too many parameters. In both statistics and machine learning, in order to avoid overfitting, it is necessary to use additional techniques, that can indicate when further training is not resulting in better generalization.

input variables and the output variables are not developed by engineering judgment, so that the model tends to be a black box.

4.3 NN used in different image processing applications

Egmont-Petersen et al. [21] reviewed in his paper more than 200 applications of NN in image processing. Figure 17 shows the number of applications where NN accomplish a specific task. Just like SVM, does NN also plays a big role in the recognition part. Besides the recognition can NN also integrate very well in the other parts, even in image understanding, but that is beyond the scope of this paper. It is quite conspicuous that the different image processing parts is based on pixels, because NN has a hard time with high dimensional data. One explanation is the use of both supervised and unsupervised NN, supervised can directly measure, for instance, the information loss of feature extraction. Unsupervised NN does not have this ability and are maybe better of with pixel based input.

	Preprocessing	Compression/ feature extract	Segmentation	Recognition	Image understanding	Optimisation
Pixel	26	25	39	51	3	5
Feature	4	2	19	38	2	3
Structure			2	6		5
Object						1
Object set				2	2	
Scene						

Figure 17 Each cell contains the number of applications where NN accomplish a specific task in the image processing chain.

4.4 NN papers

- Ishak et al. [42] presents a real-time system to detect speed limit signs and remind drivers about the allowable speed limit on that specific road. The detection is based on colour segmentation and template matching is used to detect the circle shape of the signs. By calculating first the cross-correlation in the frequency domain improves the speed of the total detection process. Classification is performed on the potential regions by using multi-layer perceptron NN. The results in Table 6 proved the feasibility of this system. These results were also verified in another paper of Ishak et al. [41].

Table 6 Results of speed limit recognition

module	# of signs	# of identification	accuracy
detection	102	5	95%
recognition	102	8	92%

- Esclalera et al. [23] used colour thresholding and the corners of the shape of the signs to extract potential candidates from the image. For the classification, the detected sign was used as the input pattern for a NN. Several networks with different number of layers and nodes were trained and tested. All the algorithms can be achieved in real time and there were also some improvements of partial occlusion and the use of other examples of NN.

- The paper of Rahman et al. [61] describes a system that warns and navigates people through audio stream. It uses a multi-layer perceptron NN with a sigmoid transfer function to recognize the traffic signs. The input to the NN is pre-processed, which has the task of skewness correction, boundary deletion, and scaling. The obtained accuracy rate was calculated at 91.48 percent.

- The proposed recognition system of Fang et al. [29] is motivated by human recognition processing. The system consists of three components: sensory, perceptual, and conceptual analyzers. The sensory extract the potential regions from the retrieved image. The extracted regions serves as the input for a spatiotemporal attentional neural network. Potential features of traffic signs are extracted from the image areas corresponding to the focuses of attention. The extracted features are the input for the conceptual analyzer. The conceptual analyzer consists of two parts: a category part and an object part. The first one uses a configurable adaptive resonance theory neural network to determine the category of the input. The last one uses a configurable heteroassociative memory NN to recognize an object in the specific category. The results shows the feasibility of the computational model and the robustness of the developed detection system. The system classifies 99 percent correct and 85 percent of the extracted traffic signs can be recognized correctly.

- Bargeton et al. [7] presents an improved European speed-limit sign recognition system based on global number segmentation before digit segmentation and recognition. The gray-scale based system is insensitive to colour variability and quite robust to illumination variations, as shown by an on-road evaluation under bad weather conditions which yielded 84 percent good detection and recognition rate, and by night-time evaluation with a 75 percent correct detection rate. The multilayer perceptron NN is used for the pattern recognition. Due to recognition occurring at digit level, the system had the potential to be very easily extended to handle properly all variants of speed-limit signs from various European countries. Table 7 shows the results of the speed-limit sign recognition system.

Table 7 Global evaluation of European speed limit sign detection

sign recognition method	signs detected, recognized and validated with correct type	misclassified signs
Initial digit segmentation	85%	0.70%
New 'global number segmentation' before digit segmentation	94%	0.70%

- Fang et al. [28] describes a method for detecting and tracking traffic signs from a sequence of video images with messed up backgrounds and under various weather conditions. Two NNs were developed for processing features derived from a sequence of colour images, one for colour features and one for shape features. To extract traffic sign candidates, a fuzzy approach was introduced, which integrates the colour and shape features. The output of feature integration is used to detect the presence, sign, and location of traffic signs and candidates. The results showed that the system is accurate and robust. However, the large search space demands much time for detecting new traffic sign candidates. This can partially been solved by operate the NN in a parallel way, thus a second processor can reduce the search time of the feature extraction part.

- The recognition of sign patterns with the use of NN techniques is presented in a study of Lorsakul & Suthakorn [52]. Images are pre-processed with several image processing techniques, such as threshold techniques, Gaussian filter, Canny edge detection, contour, and fit ellipse. Then, a NNs is used to recognize the traffic sign patterns. The system is trained and validated to find the best network architecture. The results show highly accurate classifications of traffic sign patterns with complex background images as well as the results accomplish in reducing the computational cost of the proposed method.

- Hamdoun et al. [38] presents a prototype of the globally recognized end-of-speed-limit signs by a multilayer perceptron NN. The supplementary signs are detected by applying a rectangle detection in a region below recognized speed-limit signs, followed by a multilayer perceptron NN recognition. The performance of the detection and recognition of end-of-speed-limit signs is 82 percent and the supplementary signs have a 78 percent correct classification rate. The detection and recognition of supplementary signs can easily be extended to handle more kinds of supplementary signs.

- Zhang & Luo [80] and Zhang et al. [81] used a probabilistic NN for the recognition phase. Experimental results show a recognition rate of 98 percent. For the extraction of features they used central projection transformation, which results in global feature and invariant to object scales and variations. They also showed that the recognition rate is higher than that of other methods based on invariant methods and it has the real-time system abilities.

- Yok-Yen & Abbas [79] studied the existing traffic sign recognition. In this study, the issues associated with automatic traffic sign recognition are described, the existing methods developed to attempt the traffic sign recognition problem are reviewed, and a comparison of the features of these methods is given. The developed traffic sign recognition system is described, which consists of two modules: detection and classification. The detection module segments the input image in the hue saturation intensity colour space, and then detects traffic signs using a multi layer perceptron NN. The classification module determines the type of detected traffic signs using a series of one to one architectural multi layer perceptron NN. Two sets of classifiers are trained using the resillient backpropagation and scaled conjugate gradient algorithms. The two modules of the system are evaluated individually first. Then the system is tested as a whole. The experimental results demonstrate that the system is capable of achieving an average recognition hit rate of 95.96 percent using the scaled conjugate gradient trained classifiers. The same results were achieved in an earlier work of Yok-Yen & Abbas [78].

- Lu et al. [54] proposed an artificial neural network system for traffic sign recognition. The input image is first processed for extraction of colour and geometric information. A morphological filter is applied to increase the saliency by eliminating smaller objects. The coordinates of the resulting objects are determined, and the objects are isolated from the original image according to these coordinates. After this, the objects are normalized and sent to the NN which performs the recognition. The NN consists of classification sub-network, winner-takes-all sub-network (Hopfield network), and validation sub-network. By introducing the new concept of a validation sub-network, the network enhance the capability to correctly classify the different traffic signs and avoid misclassifying non-traffic signs into a traffic sign. The system is tested by simulation as a whole and in part on a large amount of data acquired by a video camera attached to a vehicle frame by frame. The performance is encouraging. It produced excellent results except for the images under very poor illumination such that the color threshold (pre-processing) fails to extract the color information.

4.4 Overview

We concluded in section 3.4 that SVM performs much better in high dimensional data compared to NN. So, it is quite clear that successful classification and recognition with NN needs to put more effort in the pre-processing and segmentation part. This reduces the dimension of the input data to the NN, which will enhance the performance significant. This is confirmed in the research of Bargeton et al. [7] and Fang et al. [28].

The examined TDSR papers only involved detection, classification, and recognition. We have already seen in the paper of Egmont-Petersen et al. [21] that the use of NN can be incorporated in each separate part of the image processing chain. There is thus room for further research in the other parts of the image processing chain.

Fang et al. [28] also showed that the joint analysis of shape and colour increases the accuracy, but the performance decreased significant. Therefore one can decide to put more effort in the pre-processing part or handle this task over to another algoritm.

The choice of the right NN architecture and the corresponding transfer function can also be a problem. Some NN configurations works good on a specific application or part in the image processing chain, but has a very low performance in other applications respectively parts in the image processing chain. We can see this back in the study of Lorsakul & Suthakorn [52] and Fang et al. [29].

To conclude, the research of NN in TSDR systems can easily be extended in several directions. The performance is in general quite good, but there has to be a balance between computational cost and dimensionality reduction.

5 Evolutionary computing

Over the last two decades, ideas taken from the theory of evolution in natural systems have inspired the development of a group of potent yet odd flexible optimization methods known collectively as evolutionary computation (EC). In computer science is EC a subfield of CI[14] that involves combinatorial optimization problems. The modern creation of EC derives from work performed in the 60s and 70s by researches such as Holland [40], Rechenberg [62], and Fogel et al. [31]. Holland introduced a method called genetic algorithm, while Fogel et al. called his framework genetic programming, and Rechenberg presented evolution strategies. Their stochastic search methods share the common themes of mimicking the metaphor of natural biological evolution. Many different problems from different domains have been successfully attempted the use of EC. We can think of optimization of dynamic routing in telecommunications networks (Kampstra [45]), designing finite-impulse-response digital filters, product design, routing problems, designing protein sequences with desired structures, and many others. More information about EC can be found in the book of Eiben & Smith [20].

5.1 Evolutionary Algorithms

Evolutionary techniques mostly involves meta-heuristic[15] optimization algorithms. The most popular techniques are evolutionary algorithms and swarm intelligence. The basic evolutionary algorithms (EA) encompasses genetic algorithm, genetic programming, and evolution strategies. EA share the common themes of optimization performed on a population of potential solutions applying techniques, inspired by biological evolution, to produce better and better approximations to a solution. Because of the biological inspiration, we talk about individuals that represent solutions or points of a search space, also called environment. On this environment, a maximum of a fitness (evaluation) function is then searched. Individuals (chromosomes) are usually represented as codes (genes). These codes can be real, binary, fixed or variable size, simple or complex. An EA evolves its population in a way that makes individuals more and more adapted to the environment. In other words, the fitness function is maximized. At each generation, a new set of approximations to a solution is created by the process of selecting individuals of this population according to their level of fitness in the problem domain and breeding them together using operators borrowed from natural genetics. EA model natural processes like selection, recombination or crossover[16], and mutation. The latter two are the most basic genetic operators used to maintain genetic diversity, which is crucial in the process of evolution. For a simple overview of the EA, see Figure 18. The EA work on populations of individuals instead of singe individuals, this way the search is performed in a parallel manner. Despite of the simplicity of an evolutionary process, building an efficient evolutionary algorithm is a difficult task, mostly because the process is sensitive to parameter and algorithm setting. The elaboration of an efficient evolutionary algorithm is

---

[14] Computational intelligence is a branch of artificial intelligence. It is an alternative to the 'good old-fashioned artificial intelligence', which relies on heuristic algorithms like fuzzy systems, neural networks, swarm intelligence, chaos theory, artificial immune systems, wavelets, and evolutionary computation. The 'good old-fashioned artificial intelligence' is an approach to achieving artificial intelligence.

[15] A meta-heuristic is a method for solving a very general class of computational problems, by combining user-given black-box procedures, in the hope of obtaining more efficient or more robust procedure.

[16] The notes recombination and crossover are equivalent in the area of evolutionary computing. Genetic algorithms mostly use the name crossover.

based on a good knowledge of the problem to be solved. A black box approach is definitely not recommend.

We now describe briefly the basic steps of an EA:

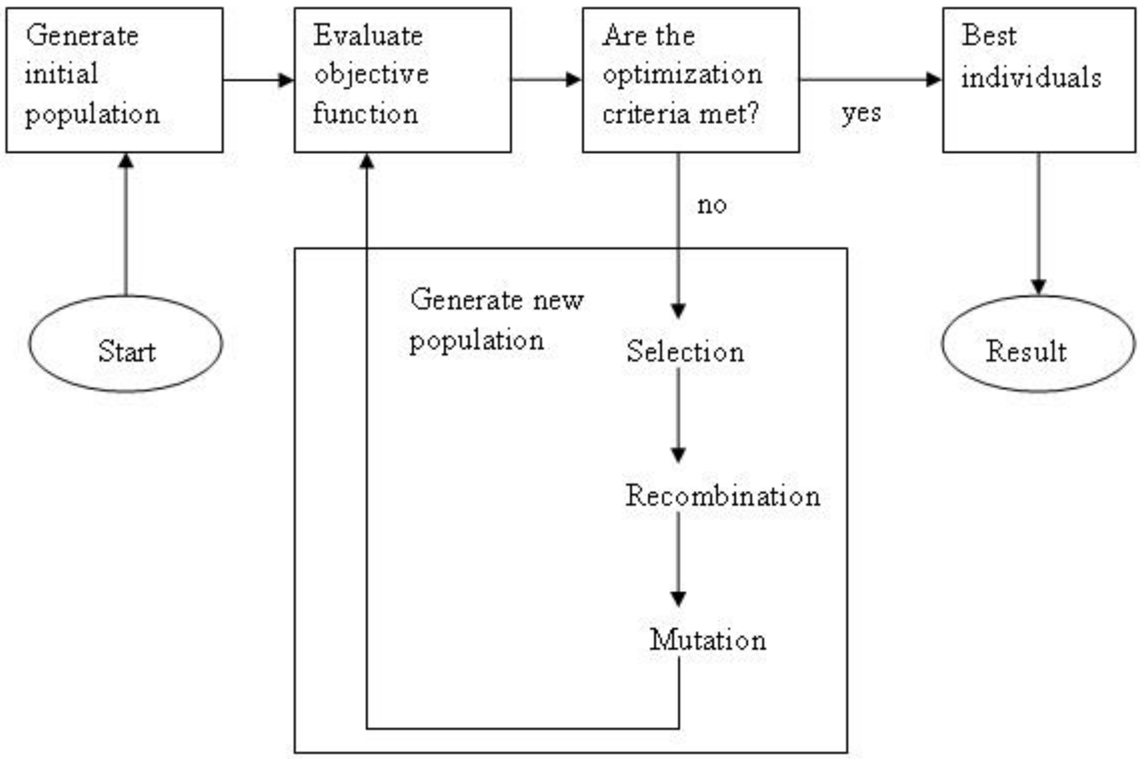


Figure 18 A simple overview of evolutionary algorithms.

First the assignment of fitness for each individual is performed, and thereafter the actual selection is done. We can distinguish the following general selection assignment schemes: proportional selection, rank based selection, and multi-objective ranking. The broadly used methods for the selection of the parents by means of their fitness are: roulette wheel selection, stochastic universal sampling, local selection, truncation selection, and tournament selection.
Parents are recombined to produce offspring in combining the information contained in the parents. All offspring will be mutated with a certain small probability. The fitness of the offspring is then computed. The offspring are inserted into the population replacing the parents, producing a new generation. This cycle is performed until the optimization criteria are reached.

Genetic operators directly depend on the choice of the representation, which, for instance, makes the difference between genetic algorithms, evolution strategies, and genetic programming.

Intuitively, selection and recombination tend to concentrate the population near good individuals (information exploitation). On the contrary, mutation limits the attraction of the best individuals in order to let the population explore other areas of the search space.

The following algorithms differ in the implementation and the nature of the particular applied problem.

5.1.1 Genetic Algorithm

Genetic algorithms Are the most popular type of EA. One seeks the solution of a problem in the form of strings of numbers, by applying genetic operators such as recombination and/or mutation. This type of EA is often used for optimization problems. are based on the use of binary representation of solutions, extended later to discrete representations.
Each individual of the population is represented by a fixed size string, with the characters (genes) being chosen from a finite alphabet. This representation is obviously suitable for discrete combinatorial problems. The most classical crossover operators used in optimization tasks can be seen in Figure 19.

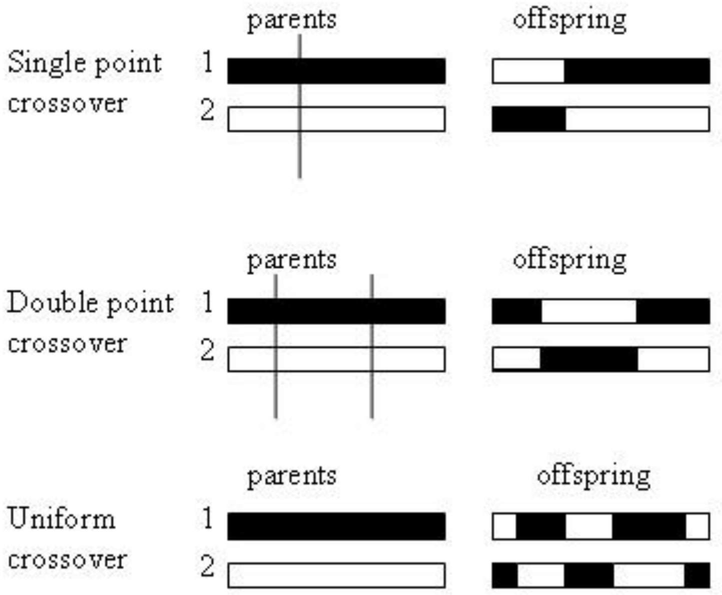


Figure 19 Binary crossover.

The single point crossover randomly chooses a position on the chromosome and then exchanges chain parts around this point. The double point crossover also exchanges portions of chromosomes, but selects two points for the exchange. Finally, the uniform crossover is a multipoint generalization of the previous one: each gene of an offspring is randomly chosen between the parents' genes at the same position. The classical binary mutation flips each bit of the chromosome with a specific probability. This specific probability is usual constant along the evolution and is very low, see Figure 20.

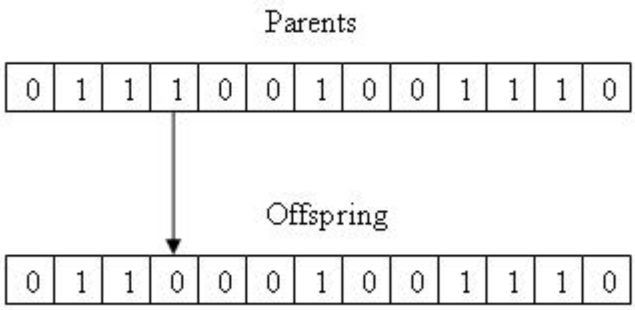


Figure 20 Binary mutation.

5.1.2. Evolution Strategies

The continuous representation, or real representation, is historically related to evolution strategies. This associated genetic operators are either extensions to continuous space of discrete operators, or directly continuous operators. The discrete crossover is a mixing of real genes of a chromosome, without change of their content. The previous binary crossover operators, can thus be adapted in a simple way. The benefit of continuous representation is surely better exploited with specialized operators, that is, continuous crossover that mixes more intimately the components of the parents to produce new offspring. The barycentric crossover, also called arithmetic, produces an offspring x' from a couple (x, y) thanks to a uniform random shot of a constant α in $[0,1]$ such that $\forall i \in 1,\ldots,n, \quad x_i = \alpha x'_i + (1-\alpha) y_i$. Many mutation operators have been proposed for the real representation. The most classical is the Gaussian mutation, that adds a Gaussian noise to the components of the individual.

5.1.3. Genetic Programming

Genetic programming corresponds to a representation of variable length structures as trees. The richness and versatility of the variable size tree representation are at the origin of the success of genetic programming. Recently [75] in the computer vision domain, genetic programming has been shown to achieve human competitive results. A genetic programming algorithm explores a search space of recursive programs made of elements of a function set, of a variable set, and of a terminal set. Individuals of the population are programs that, when executed, produce the solution to the problem at hand. Crossover are often subtree exchanges Mutations are more complex, and several mutations have to be used, producing different types of suppression on the chromosome structure.

5.2 Advantages and disadvantages of EC

It can be seen, from the above, that evolutionary algorithms differ substantially from traditional search and optimization methods. The most important differences are:

- The search is done in a parallel way.
- No derivative information or other secondary knowledge is required, only the objective function and the corresponding fitness levels manipulate the direction of search.
- Only probabilistic transition rules are used, no deterministic rules.
- More straightforward to apply, because no restrictions for the definitions of the objective function exists.
- Provide a number of potential solutions, so the choice is up to the user. This can be useful if a specific problem does not have one single solution.

There are several advantages of genetic algorithms over current methods for segmentation such as clustering. First, the genetic mechanism is independent of the prescribed evaluation function and can be tailored to support a variety of characterizations based on heuristics depending on genre, domain, user type, etc. Second, evolutionary algorithms are naturally suited for doing incremental segmentation, which may be applied to streaming media. Third, it can support dynamically updated segmentation that adapt to usage patterns, like adaptively increasing the likelihood that frequently accessed points will appear as segment boundaries.

5.3 EA in different image processing applications

We can find the discussed EA back in each separate part of the image processing chain. GA are the most frequently used in practice. Interest in the other EA types is growing, however, so that a rise in the number of their respective applications can be expected in the near future. ES already cover a range of management related applications. GP is a very recent technique that has attracted attention mainly from practitioners in the financial sector. Below we come across some examples of image processing applications, that utilizes the genetic algorithm, genetic programming, and evolutionary strategies in the different parts. By doing so, we demonstrate that EA can be useful in each separate part of the image processing chain. Unfortunately, the founded TDSR papers that uses EA were quite small and therefore it can be handy to show that there is room for extended research in this specific area.

- For instance, Chiu et al. [13] describes a genetic segmentation algorithm for image data streams and video that employs a segment fair crossover operation. This algorithm operates on segments of a string representation, which is similar to classical genetic algorithms that operates on bits of a string. One of the main advantages of genetic segmentation algorithms over standard algorithms is the easier adaptation of the fitness function and the incremental segmentation.

- Lutton & Vehel [55] uses find genetic algorithm in the pre-processing part of the image processing chain. They dealt with the denoising of complex signals in images, which were very difficult to handle with classical filtering techniques. The problem of denoising has been turned into an optimization problem: searching for a signal with a prescribed regularity that is as near as possible to the original noisy signal. The use of find GA have been found to be useful in this case, and yield better results than other algorithms.

- Cagnoni et al. [10] describes two tasks that have been designed to be possible parts of a license plate recognition system. The first task is designing automatically a set of binary classifiers for low resolution characters and the second task is the development of another image pre-processing procedure. The presented applications used GP to recognize the low resolution characters and developed an image pre-processing technique for license plate detection. The results shows that, even in a very simple configuration, the genetic programming outperforms NN and SVM and it is also 10 times faster.

- Ciesielski et al. [16] shows that genetic programming can be used for texture classification in three ways. The first is a classification technique for feature vectors generated by usual feature extraction algorithms. The second is a one step method that bypasses feature extraction and generates classifiers directly from image pixels. The last one is a method of generating new feature extraction programs. The results shows that the classifiers can be used for fast, accurate texture segmentation. They also showed that GP can overcome some of the traditional drawbacks of texture analysis techniques.

- Louchet [53] shows how evolution strategies can actually widen the scope of the basic feature extraction techniques. The author also illustrates how ES can be an important factor in image analysis, thanks to their ability to efficiently explore complex model parameter spaces. Further on, the author also shows that the algorithm is fast with interesting real-time and asynchronous properties. This could be an important property for the TSDR system.

5.4 EC Papers

- Aoyagi & Asakura [4] presents a GA for the traffic sign detection. They only use bright images because of the hue variations. After obtaining the laplacian of the original image, there is a thresholding. Those pixels that pass the threshold are analysed later. They do not take into account different scales for the horizontal and vertical axes, thus they do a matching with a circular pattern. They provided the gene information with the x position, the y position, and the radius. The population is formed by 32 individuals, the selection rate is 30 percent, 10 percent for the mutation rate, and there are 150 iterations. Finally there are multiple crosspoints.

- The paper of Escalera et al. [23, 24, 25, 26] used a genetic algorithm for the detection, allowing an invariance localisation to changes in position, scale, rotation, weather conditions, partial occlusion, and the presence of other objects of the same colour. They employed the HIS colour space for the colour classification since it gives different pieces of information in every component. Thereafter, thresholding is done, and the resulting potential traffic signs are located. Once the borders of the potential traffic signs are found, the algorithm has to detect traffic signs presented in the image. They used a GA for this search problem, and they used the same gene information as described in the paper of Aoyagi. The gene codification starts from a sign model representing a sign at a fix distance and perpendicular to the optical axes. The considered modifications are a change in the position and in the scales, due to the sign being farther or nearer than the model, or because the optical axis is not perpendicular to the sign producing a deformation in it, which is due to the magnification difference for every axis. All these factors can be expressed if there is an affine transformation between the ideal model without deformations and the model that is being looked for in the image[17]:

$$\begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = \begin{bmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_m \\ Y_m \\ 1 \end{bmatrix}$$

The transform coefficients are $a_{00} = E_x \cos\theta$, $a_{01} = E_x \sin\theta$, $a_{02} = T_x$, $a_{10} = -E_y \sin\theta$, $a_{11} = E_y \cos\theta$, and $a_{12} = T_y$. Where T_x is the horizontal displacement, T_y is the vertical displacement, E_x is the horizontal scale, E_y is the vertical scale, and θ is the horizontal rotation. See Figure 21 for a graphical example of the affine transformation of a deformed traffic sign to an ideal traffic sign. In the case of circular signs, there is no rotation and the transform coefficients are $a_{00} = E_x$, $a_{01} = 0$, $a_{02} = T_x$, $a_{10} = 0$, $a_{11} = E_y$, $a_{12} = T_y$.

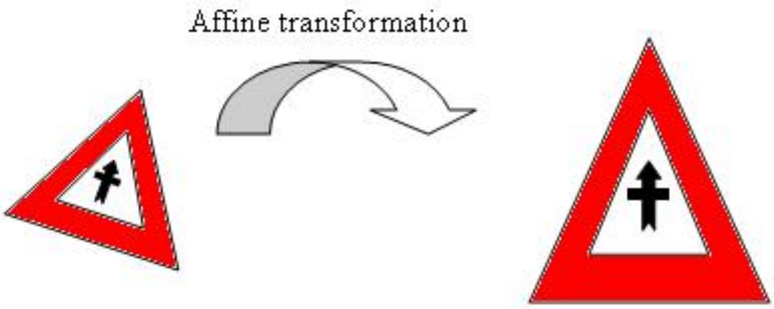


Figure 21 Affine transformation of the actual sign to the ideal sign without any deformations.

Initialisation: In a classical GA, the initial population is generated randomly, but, in this case, as some information is known from the colour analysis, some values can be obtained that will be nearer to the final one than a random start. To do this, a thresholding of the colour analysis image is performed and the number and position of the potential regions are obtained. A fixed number of individuals are assigned to every potential region. This way, the presence of enough individuals can be guaranteed despite the presence of bigger objects or occlusion.
Fitness evaluation: The fitness is based on the Hausdorff distance. The used fitness function can be immune to occlusion and noise and allows stopping if the percentage is high enough.
Selection: The process extends genes of good solutions through the population. This selection is done by using the ranking method. Following by a crossover and mutation step. Finally the best individual is kept. The classification is done by NN, because of their ability to generalise from training patterns and their invariance to occlusion.

[17] To refresh your memory about rotation, scaling, and translation; check the following website: http://www.senocular.com/flash/tutorials/transformmatrix/

- Soetedjo & Yamada [74] used geometric fragmentation to detect circular red traffic signs by finding the left and right fragments of elliptical objects to increase the accuracy of detection and handle occlusion. The search for fragments resembles a GA The objective function for evaluating individuals is devised to increase detection accuracy and reduce computation time. The results showed that GA compared to conventional template matching performed better in detection and execution time and does not require a large number of carefully prepared templates. The same results were achieved in an earlier study of Soetedjo & Yamada [73].

- Ishida et al. [43] present a novel training method for recognizing traffic sign symbols. The symbol images captured by a car mounted camera suffer from various forms of image degradation. To cope with degradations, similarly degraded images should be used as training data. The method artificially generates such training data from original templates of traffic sign symbols. Degradation models and a GA based algorithm that simulates actual captured images are established. The proposed method enables them to obtain training data of all categories without exhaustively collecting them. Experimental results show the effectiveness of the proposed method for traffic sign symbol recognition.

- Dang et al. [17] developed a radial basis function NN applications in the traffic sign recognition. Firstly traffic signs are detected by using their color and shape information. Then GA, which has a powerful global exploration capability, is applied to train RBFNN to obtain appropriate structures and parameters according to given objective functions. In order to improve recognition speed and accuracy, traffic signs are classified into three categories by special color and shape information. Three RBFNN are designed for the three categories. Before fed into networks, the sign images are transformed into binary images and their features are optimized by linear discriminate analysis. The training set imitating possible sign transformations in real road conditions, is created to train and test the nets. The experimental results show the feasibility and validity of the proposed algorithm.

5.5 overview

GA is the most used technique of EC, it is a fast and accurate algorithm which can outperform NN and SVM in some particular tasks. It is therefore very useful in TSDR systems. Besides GA, achieves GP and ES also excellent performance. This fits in the research of Soetedjo & Yamada [74, 73]

We can, just like NN, find EC in almost every part of the image processing chain. Unfortunately, is the use of EC not that widely spread in the field of TSDR. We can, once again, only find the use of EC in the detection, classification, and recognition part. To make it even worse, the retrieved TSDR papers only contained GA instead of all three EC techniques. Nevertheless, EC shows promising results in other image processing applications. Therefore we can assume that the use of EC is not really integrated in the field of TSDR. Besides that, the results were better than the traditional methods, which were invariant in rotation, occlusion, and scale.

We have already explained the advantages and disadvantages of EC in the image processing chain, but we like to add that the real potential of these techniques is unleashed when they are joined together.

6 Conclusion

This paper gives an overview of three, widely used, techniques on the topic of traffic sign detection and recognition. Statistical methods seem limited in this field and therefore much research has been done to find methods that are more accurate.

SVM are a fairly new development and research showed that it has high classification accuracies and besides that it is not too hard to explain them mathematically. They also have the advantage that they are invariance of orientation, illumination, and scaling. Then again, the selection of the right kernel function is crucial for the overall performance.

NN models have received a lot of attention, but these methods suffer from the disadvantage of a lack of explanation of their outcomes. Furthermore, they require more attention in dimensionality reduction compared to the two other techniques. However, NN are very flexible, tolerant to imperfect data, and powerful. In addition, there is no need to create an algorithm in order to perform a specific task; thus there is no need to understand the internal mechanisms of that task, which result in the applicability to a wide range of problems.

EC can be used in every part of the image processing chain, but the novel algorithms are not fully integrated in the field of traffic sign detection and recognition. The performance is, just like the other two techniques, quite good, and the difference between the performance of the techniques depends on the problem specific task. They also have the advantage that they are invariance of orientation, illumination, and scaling.

A hybrid model through integration of EC and SVM or NN may overcome the problems which they have to deal with normally. For instance, they can also help in shorten the time it takes to train a NN or SVM. Then again they are not a solution to the limitations of NN and SVM, so best would be to investigate what opportunities they can bring in combination with other methods.

As a final word, the choice of a method and the use of a technique depends on the complexity of the problem specific task. It can be a time consuming job to find the right settings of the different techniques, but with the use of EC we can speed things up.

The research in the field of traffic sign detection and recognition is limited, but NN is mostly used in this specific field, also in the general computer vision. Observing the good results, but poorly available research, of each emphasized technique, follows by the conclusion that there is room for a lot more promising research.

7 Further research

The study of the three emphasized methods in traffic sign detection and recognition can be easily extended with more research. The results are already very good, but the integration of these techniques together should unleash there full power.

Some hybrid systems integrating EA with NN, fuzzy sets, and rule based systems are documented in the field of computer vision. Since they are expensive to develop and may yield considerable strategic advantage over competitors, it can be assumed that much work in hybrid systems. Cho [15] presented GA method of combining NN for producing an improves performance on real-world recognition problems. The experimental results for classifying a large set of handwritten digits show that it improves the generalisation capability significantly. Thus there is much potential in pattern recognition problems for hybrid systems. Especially for TSDR systems, because they are capable to perform in real-time.

References

1. Addison J.F.D., Wermter S., Arevian G.Z. *A comparison of feature extraction and selection techniques.* Proceedings of international conference on artificial neural networks, pp. 259-262, 2003.
2. Adler A., Guardo R. *A neural network image reconstruction technique for electrical impedance tomography.* IEEE Trans. Medical Imaging, vol. 13, pp. 594-600, 1994.
3. Al-Madani H., Al-Janahi A. *Assessment of drivers' comprehension of traffic signs based on their traffic, personal and social characteristics.* Transportation research part F: Traffic psychology and behaviour, vol. 5, issue 1, pp. 63-76, 2002.
4. Aoyagi Y., Asakura T. *A study on traffic sign recognition in scene image using genetic algorithms and neural networks.* Industrial electronics, control, and instrumentation, vol. 3, pp. 1838-1843, 1996.
5. Avola D., Ferri F., Grifoni P. *Genetic algorithms and other approaches in image feature extraction and representation.* Artificial intelligence for maximizing content based image retrieval, pp. 19, dea Group Publ, 2008.
6. Bahlmann C., Zhu Y., Ramesh V., Pellkofer M., Koehler T. *A system for traffic sign detection, tracking, and recognition using color, shape, and motion information.* Intelligent vehicles symposium, pp. 255-260, 2005.
7. Bargeton A., Moutarde F., Nashashibi F., Bradai B. *Improving pan-European speed-limit signs recognition with a new 'global number segmentation' before digit recognition.* IEEE Intelligent vehicles symposium, pp. 1022-1027, 2008.
8. Betke M., Makris N.C. *Fast object recognition in noisy images using simulated annealing.* Computer vision, pp. 523-530, 1995.
9. Burges C.J.C. *A tutorial on support vector machines for pattern recognition.* Data mining and knowledge discovery, pp. 121-167, 1998.
10. Cagnoni S., Mordonini M., Adorni G. *Sub-machine-code genetic programming for binary image analysis.* Genetic and evolutionary computation for image processing and analysis, Hindawi publishing corporation, vol. 8, pp. 47-64, 2007.
11. Camps-Vals L.G., Calpe-Maravilla J., Soria-Olivas E., Martin-Guerrero J.D., Moreno J. *Support vector machines for crop classification using hyperspectral data.* Pattern recognition and image analysis, vol. 2652, pp. 134-141, 2003.
12. Ceylan H., Bell M.G.H. *Traffic signal timing optimisation based on genetic algorithm approach, including drivers' routing.* Elsevier, transportation research part b, vol. 38, pp. 329-342, 2004.
13. Chen S.H., Wang P. *Computational intelligence in economics and finance.* Advanced information processing series, Springer, pp. 480, 2003.
14. Chiu P., Girgensohn A., Polak W., Rieffel E., Wilcox L., Bennet III F.H. *A genetic segmentation algorithm for image data streams and video.* Genetic and evolutionary computation conference, pp. 666-673, 2000.
15. Cho S-B. *Pattern recognition with neural networks combined by genetic algorithm.* Elsevier, fuzzy sets and systems, vol. 103, pp. 339-347, 1999.
16. Ciesielski V., Song A., Lam B. *Visual texture classification and segmentation by genetic programming.* Genetic and evolutionary computation for image processing and analysis, Hindawi publishing corporation, vol. 8, pp. 195-214, 2007.
17. Dang J., Wang Y., Zhu Z. *Traffic sign detection and recognition by improved RBFNN.* Intelligent systems and control, pp. 433-437, 2007
18. Department for transport. *Transport statistics, transport trends.* London, HMSO, 2003a. http://www.dft.gov.uk/
19. Duch W., Mandziuk J. *Challenges for computational intelligence.* Springer, pp. 499, 2007.

20. Eiben A.J., Smith J.E. *Introduction to evolutionary computing.* Springer, pp. 299, 2003.
21. Egmont-Petersen M., Ridder D. de, Handels H. *Image processing with neural networks – a review.* Pattern recognition, vol. 35, pp. 2279-2301, 2002.
22. Engelbrecht A.P. *Computational intelligence: An introduction.* Wiley, pp. 310, 2003.
23. Escalera A. de la, Moreno L.E., Salichs M.A., Armingol J.M. *Road Traffic sign detection and classification,* IEEE transactions on industrial electronics, vol. 44, pp. 848-859, 1997.
24. Escalera A., Armingol J.M., Salichs M.A. *Traffic sign detection for driver support systems.* International conference on field and service robotics, pp. 7, 2001.
25. Escalera A. de la, Armingol J.M., Mata M. *Traffic sign recognition and analysis for intelligent vehicles.* Image and vision computing, vol. 21, issue 3, pp. 247-258, 2003.
26. Escalera A. de la, Armingol J.M., Pastor J.M., Rodriguez F.J. *Visual sign information extraction and identification by deformable models for intelligent vehicles.* IEEE transactions on intelligent transportation systems, vol. 5, no. 2, pp. 57-68, 2004.
27. Estable S., Schick J., Stein F., Janssen R., Ott R., Ritter W., Zheng Y.-J. *A real-time traffic sign recognition system.* Intelligent vehicles symposium, pp. 213-218, 1994.
28. Fang C.Y., Chen S.W., Fuh C.S. *Road-sign detection and tracking.* IEEE transactions on vehicular technology, vol. 52, pp. 1329-1341, 2003.
29. Fang C.Y., Fuh C.S., Yen P.S., Cherng S., Chen S.W. *An automatic road sign recognition system based on a computational model of human recognition processing.* Computer vision and image understanding, pp. 237-268, 2004.
30. Fleyeh H., Dougherty M. *Road and traffic sign detection and recognition.* Advanced OR and AI methods in transportation, pp. 747-753, 2005.
31. Fogel L., Owens A., Walsh M. *Artificial intelligence through simulated evolution.* Wiley, pp. 162, 1999.
32. Freeman J.A., Skapura D.M. *Neural Networks. Algorithms, Applications, and Programming Techniques.* Computation and neural system series, C. Koch, Addison-Wesley, pp. 414, 1991.
33. Gilani S.H. *Road sign recognition based on invariant features using support vector machine.* Master thesis, pp. 87, 2007.
34. Gil-Jimenez P., Lafuente-Arroyo S., Maldonado-Bascon R., Lopez-Ferreras F., Maldonado-Bascon S. *Traffic sign shape classification evaluation I: SVM using distance to borders.* Intelligent vehicles symposium, pp. 557-562, 2005.
35. Gil-Jimenez P., Lafuente-Arroyo S., Maldonado-Bascon R., Lopez-Ferreras F., Maldonado-Bascon S. *Traffic sign shape classification evaluation II: FFT applied to the signature of blobs.* Intelligent vehicles symposium, pp. 607-612, 2005.
36. Gil-Jimenez P., Lafuente-Arroyo S., Maldonado-Bascon R., Gomez-Moreno H. *Shape classification algorithm using support vector machines for traffic sign recognition.* Proceeding of IWANN, pp. 873-880, 2005.
37. Gonzalez R.C., Woods R.E. *Digital image processing, 2nd edition.* John Wiley & Sons, pp. 723, 2001.
38. Hamdoun O., Bargeton A., Moutarde F. *Detection and recognition of end-of-speed-limit and supplementary signs for improved European speed limit support.* World congress on intelligent transport systems, vol. 15, pp. 289-303, 2008.
39. Hibi T. *Vision based extraction and recognition of road sign region from natural colour image, by using HSL and coordinates transformation.* Automotive technology and automation symposium, pp 201-206, 1996.
40. Holland J.H. *Adaptation in natural and artificial systems.* Ann arbor: The university of Michigan press, pp. 228, 1975.

41. Ishak K.A., Sani M.M., Tahir N.M. *A speed limit sign recognition system using artificial neural networks.* Student conference on research and development, pp. 1-8, 2006.
42. Ishak K.A., Abdul S., Hussain A. *Development of a real-time speed limit sign recognition system.* Proceedings of the international conference on electrical engineering and informatics, pp. 731-734, 2007.
43. Ishida H., Takahashi T., Ide I., Mekada Y., Murase H. *Generation of training data by degradation models for traffic sign symbol recognition*. IEICE Transactions on Information and Systems, pp. 1134-1141, 2007.
44. Jähne B. *Digital image processing, 5th revised and extended edition.* Springer, pp. 575, 2002.
45. Kampstra P. *Evolutionary computing in telecommunications. A likely EC success story.* Master thesis, pp. 66, 2005.
46. Kotaro S., Masaru T., Atsuyuki S. *Feature extraction based on support vector machine decision boundaries.* IEIC technical report, vol. 102, no. 318, pp. 49-53, 2002.
47. Kopf M. *Advanced driver assistance systems research within INVENT.* Global automotive manufacturing & technology, pp. 111-112, 2002.
48. Lafuente-Arroyo S., Garcia-Diaz P., Acevedo-Rodriguez F.J., Gil-Jimenez P., Maldonado-Bascon S. *Traffic sign classification invariant to rotations using support vector machines.* Proceedings of advanced concepts for intelligent vision systems, pp. 256-262, 2004.
49. Li Y., Hu H. *Image compression using support vector machine.* Advanced intelligent computing theories and applications, with aspect of theoretical and methodological issues, vol. 4681, pp. 922-929, 2007.
50. Libal V., Paclik P., Kovar B., Mosna P., Vleek M., Zahrdadnik P. *Road sign recognition system using TMS320C80.* European DSP educational and research conference, vol. 2, pp. 41-48, 1998.
51. Liu L., Cao Z. *Omnidirection image restoration using a support vector machine.* International conference on information and automation, pp. 606-611, 2008
52. Lorsakul A., Suthakorn J. *Traffic sign recognition using neural network on opencv: Toward intelligent vehicle/driver assistance system.* Journal of intelligent service robotics, Springer, pp. 335, 2007.
53. Louchet J. *Model based image analysis using evolutionary computation*. Genetic and evolutionary computation for image processing and analysis, Hindawi publishing corporation, vol. 8, pp. 283-308, 2007.
54. Lu S.W., Ghica D., Yuan X.. *Recognition of traffic signs by artificial neural networks.* IEEE international conference on neural networks, vol. 3, pp. 1444-1449, 1995.
55. Lutton E., Vehel J.L. *Evolutionary multifractal signal/image denoising.* Genetic and evolutionary computation for image processing and analysis, Hindawi publishing corporation, vol. 8, pp. 23-45, 2007.
56. Maldonado-Bascon S., Lafuente-Arroyo S., Gil-Jimenez P., Gomez-Moreno H., Lopez-Ferreras F. *Road-sign detection and recognition based on support vector machines.* IEEE transactions on intelligent transportation systems, vol. 8, no. 2, pp. 264-279, 2007.
57. Nakata J., Ogawa K. *Image reconstruction from projections using genetic algorithm.* Nuclear science symposium and medical imaging conference, vol. 4, pp. 1706-1709, 1994.
58. Ozyildiz E., Krahnstover N., Sharma R. *Adaptive texture and color segmentation for tracking moving objects.* Pattern recognition, vol. 35, pp. 2013-2029, 2002.

59. Paclik P., Novovicova J., Pudil P., Somol P. *Road sign classification using Laplace kernel classifier.* 11th Scandinavian conference on image analysis, vol. 21, pp. 1165-1173, 2000.
60. Priese L., Rehermann V., Schian R., Lakmann R.. *Traffic sign recognition bases on color image evaluation.* Intelligent vehicles symposium, pp. 95-100, 1993.
61. Rahman M.O., Mousumi F.A., Scavino E., Hussain A., Basri H. *Real time road sign recognition system using artificial neural networks for Bengali textual information box.* European journal of scientific research, vol. 25, pp. 478-487, 2009.
62. Rechenberg I. *Evolutionsstrategie: Optimierung technischer systeme nach princiepien der biologischen evolution.* Fromman-Holzboog, pp. 434, 1993.
63. Rehrmann V., Lakmann R., Priese L. *A parallel system for real-time traffic sign recognition.* pp. 5, 2008.
64. Ridder D. de, Duin R.P.W., Verbeek P.W. *The applicability of neural networks to non-linear image processing.* Pattern analysis and applications, vol. 2, pp. 111-128, 1999.
65. Rosenberger C., Chehdi K. *Genetic fusion: Application to multi components image segmentation.* Proceedings of the IEEE international conference on acoustics speech and signal processing, vol. 4, pp. 2219-2222, 2000.
66. Rosenberger C., Chehdi K. *Supervised genetic image segmentation.* Proceedings of the IEEE international conference on systems, man and cybernetics, vol. 5, pp. 969-980, 2002.
67. Rumar K. *The role of perceptual and cognitive filters in observed behaviour*. L. Evans & R. C. Schwing, human behaviour and traffic safety, Plenum Press, pp.151-165, 1985.
68. Shi M. *Road and traffic signs recognition using support vector machines.* Master thesis, pp. 100, 2006.
69. Shi M., Wu H., Hasan F. *A robust model for traffic signs recognition based on support vector machines.* Image and signal processing, vol.4, pp. 516-524, 2008.
70. Silapachote P., Weinman J., Hanson A., Weiss R., Mattar M.A. *Automatic sign detection and recognition in natural scenes.* IEEE conference on computer vision and pattern recognition, vol. 3, pp. 1-27, 2005.
71. Simon L., Tarel J-P., Bremond R. *A new paradigm for the computation of conspicuity of traffic signs in road images.* Proceedings of 26th session of commision internationale de L'Eclairage, vol. 2, pp. 161-164, 2007.
72. Simon L., Tarel J-P., Bremond R. *Towards the estimation of conspicuity with visual priors.* Proceedings of 26th session of commision internationale de L'Eclairage, vol.2, pp. 38-41, 2007.
73. Soetedjo A., Yamada K. *Fast and robust traffic sign detection.* Systems, man and cybernetics, vol. 2, pp. 1341-1346, 2005.
74. Soetedjo A., Yamada K. *An efficient algorithm for traffic sign detection.* Journal of advanced computational intelligence and intelligent informatics, vol. 10, nr. 3, pp. 409-418, 2005.
75. Trujillo L., Olague G. *Synthesis of interest point detectors through genetic programming.* In proceedings of the 8th annual conference on genetic and evolutionary computation, pp. 887-894, 2006.
76. Vlacic L., Parent M., Harashima F. *Intelligent vehicle Technologies: Theory and applications, pp. 498,* Butterworth-Heinemann, 2001.
77. Yang M., Kpalma K., Ronsin J. *A survey of shape feature extraction techniques.* Pattern recognition techniques, technology, and applications, I-Tech, Vienna, pp. 626, 2008.

78. Yok-Yen N., Abbas K. *Automatic road sign recognition using neural networks.* International joint conference of neural networks, pp. 3955-3962, 2006.
79. Yok-Yen N., Abbas K. *Detection and classification of road signs in natural envirnments.* Springer, Neural computing and Applications, vol. 17, nr. 3, pp. 265-289, 2008.
80. Zhang H., Luo D. *A new method for traffic signs classification using probabilistic neural networks.* Springer, Advances in neural networks, pp. 33-39, 2006.
81. Zhang K., Sheng Y., Wang P., Luo L, Ye C., Gong Z. *Automatic recognition of traffic signs in natural scene image based on central projection transformation.* The international archives of the photogrammetry, remote sensing and spatial information sciences, vol. 37, pp. 627-633, 2008.
82. Zheng Y., Ritter W., Janssen R. *An adaptive system for traffic sign recognition.* Intelligent vehicles symposium, pp. 165-170, 1994.
83. Zhu S., Liu L. *Traffic sign recognition based on color standardization.* IEEE International conference on information acquisition, pp. 951-955, 2006.
84. Lemmens K. *Alles over verkeersborden.* 2008. http://proto.thinkquest.nl/~klb019/
85. *Vienna convention on road traffic*. 2008. http://en.wikipedia.org/wiki/Vienna_Convention_on_Road_Traffic

Appendix 1

Traffic on road may consists of pedestrians, cyclists, motor–cycles, ridden or herded animals and vehicles. The rules on the road are both the traffic laws and the informal rules that may be developed over time to facilitate the orderly and timely flow of traffic. Rules on the road are the basic practices and procedures that road users follow, they manage interactions with other vehicles and pedestrians. In 1968 the Europe countries signed an international treaty, called the *Vienna convention on road traffic*, for the basic traffic rules. The aim of standardizing traffic regulations in participating countries in order to facilitate international road traffic and to increase road safety. A part of this treaty defined the traffic signs and signals. As a result, in Europe the traffic signs are well standardized, although not all countries are participants of these rules and local variations in practice may be found (see Figure 22). Since language differences can create difficulties to understanding, international signs using symbols instead of words have been developed in Europe and in most countries of the world. Annexe 1 of the Vienna convention on road traffic distinguishes eight different signs [85]:

1. Danger warning signs.
2. Priority signs.
3. Prohibition signs.
4. Mandatory signs.
5. Special regulation signs.
6. Information, facilities, or service signs.
7. Direction, position, or indication signs.
8. Additional panels.

These eight signs differ in their shapes and colours. Triangular shapes are used in warning signs. Prohibition signs are round with a red border. Informative and various other signs are of rectangular shape.

Figure 22 Different stop signs in Europe. From left to right: Spain, Italy, France, Germany, United Kingdom and The Netherlands.

We follow in the Netherlands also the Vienna principle. The directional signs, which has not been coordinated under the principle, has always a blue background colour. The destinations on the signs are white. If the destination is not a town, then the destination is black on a separate white background. All the different signs used in The Netherlands can be found on the following website 84.

Appendix 2

The optimization of a non-linear separable problem is given below. First we allow error ξ_i in the classification. By minimizing $\sum_i \xi_i$, ξ_i can be obtained by:

$$\mathbf{w}^{\mathbf{T}}\mathbf{x}_i + b \geq 1 - \xi_i \qquad y_i = 1$$
$$\mathbf{w}^{\mathbf{T}}\mathbf{x}_i + b \leq 1 - \xi_i \qquad y_i = -1$$
$$\xi_i \geq 0 \qquad \forall i$$

$\xi_i = 0$ if there is no error of $\mathbf{x}_i$ and ξ_i is an upper bound of the number of errors.

We like to minimize: $\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_i \xi_i$. C is the trade-off parameter between error and margin. The optimization becomes:

Minimize $\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_i \xi_i$

Subject to $y_i\left(\mathbf{w}^{\mathbf{T}}\mathbf{x}_i + b\right) \geq 1 - \xi_i \qquad \xi_i \geq 0$

The dual problem:

$$\mathit{Maximize}\ W(\alpha) = \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \qquad \mathit{subjectto}\ C \geq \alpha_i \geq 0, \sum_i \alpha_i y_i = 0\ \forall i$$

This is very similar to the optimization problem in the linear separable problem, except there is an upper bound C on α_i. To find α_i we can use the quadratic problem solver again. The key idea to generalize linear decision boundary to become a non-linear decision boundary is: transform $\mathbf{x}_i$ to a higher dimension space to make things easier. Input space is the space where $\mathbf{x}_i$ is located. The feature space is the space of $\phi\left(\mathbf{x}_i\right)$ after transformation. Linear operations in the feature space is equivalent to non-linear operations in the input space. Hereby, classification can become easier with a proper transformation. Unfortunately, computations can be very costly in the feature space due to the higher dimension. The solution is the kernel trick. In the dual problem the data points appear as an inner product. As long as we can calculate the inner product in the feature space, we do not need the mapping explicitly. Many common geometric operations can be expressed by inner products. Define the kernel function K by $K\left(\mathbf{x}_i, \mathbf{x}_j\right) = \phi\left(\mathbf{x}_i\right)^{\mathbf{T}} \phi\left(\mathbf{x}_j\right)$. Kernel functions can be considered as a similarity measure between the input objects. Examples of kernel functions:

Polynomial kernel: $K(\mathbf{x}, \mathbf{y}) = \left(\mathbf{x}^{\mathbf{T}}\mathbf{y} + 1\right)^d$

Radial basis function: $K(\mathbf{x},\mathbf{y}) = \exp\left(-\|\mathbf{x}-\mathbf{y}\|^2 / \left(2\sigma^2\right)\right)$

Sigmoid with parameters θ and κ: $K(\mathbf{x},\mathbf{y}) = \tanh\left(\kappa\mathbf{x}^{\mathbf{T}}\mathbf{y} + \theta\right)$

Appendix 3

Feature extraction is a special form of dimensionality reduction in image processing and in pattern recognition. When the input data to an algorithm is too large to be processed and does not contain much important information then the input data will be transformed into a reduced representation set of features. Transforming the input data into the set of features is called features extraction. If the features extracted are carefully chosen it is expected that the features set will extract the relevant information from the input data in order to perform the desired task using this reduced representation instead of the full size input. It can be used in the area of image processing (segmentation), which involves using algorithms to detect and isolate various potential features of a video stream or digitized image. Besides the lowered computational costs, it also helps in controlling the so called curse of dimensionality[18]. Some feature extraction approaches were designed to manage explicitly changes in orientation and scale of objects.

One of the most used feature extraction techniques is shape based. According to Yang et al. [77] must shape based feature extraction contain the following properties to be efficient:

- Identifiability
- Translation, rotation, and scale invariance
- Affine invariance
- Noise resistance
- Occultation invariance
- Statistically independent
- Reliability

We can distinguish the following most common detection and extraction techniques in image processing:
Shape based:

a) Thresholding is the simplest method of image extraction.. From a grey-scale image, thresholding can be used to create binary images. Individual pixels in an image are marked if their value is greater than some threshold value. There also consist local or dynamic thresholding, then there exists different thresholding values for different regions in the image.
b) Blob extraction is generally used on the resulting binary image from a thresholding step. It categorizes the pixels in an image as belonging to one of many discrete regions. Blobs may be counted, filtered, and tracked.

[18] The curse of dimensionality is a property of classification and regression problem. The higher the dimension of the feature space leads to an increased number of parameters to be estimated.

c) Template matching is a technique for finding small parts of an image which match a template image. It can be used to detect edges in an image. It can be easily used in gray-scale images or edge images.
d) Hough transform has its purpose in finding imperfect instances of objects within a certain class of shapes by a voting procedure. It is most commonly used for the detection of regular curves such as lines, circles, ellipses, etcetera.

Low (pixel) level:

e) Edge detection detects sharp changes in image brightness, and therefore captures it important events and changes in objects of the scene. It filters information out that may be regarded as not relevant, while preserving the important structural properties of an image. The downside is the edge extraction from non-trivial images which are often troubled by fragmentation, meaning that the edge curves are not connected.
f) Corner detection extracts certain kinds of features and gather the contents of an image.
g) Blob detection are aimed at detecting points and/or regions in the image that are either brighter or darker than the surrounding.
h) Scale-invariant feature transform are invariant to image scale and rotation. They are also robust to changes in illumination, noise, and minor changes in viewpoint. Object description by a set of these features are also partially invariant for occlusion. Three of these features of an object are enough to compute its location and position. Recognition can be done close to real-time, assuming that the database is not too large and an up to date computer system.

If no export knowledge is available, then the following general dimensionality reduction techniques may help:

1) Principal component analysis
2) Semi-definite embedding
3) Multifactor dimensionality reduction
4) Nonlinear dimensionality reduction
5) Isomap
6) Kernel principal component analysis
7) Latent semantic analysis
8) Partial least squares
9) Independent component analysis

Appendix 4

We can split the pre-processing techniques in two domains: spatial domain and frequency domain. The spatial domain is the normal image space, in which a change in position in this image directly projects to a change in position in the projected scene. The frequency domain is a space in which each image value at image position F represents the amount that the intensity value in this image vary over a specific distance related to F. In the spatial domain we can distinguish the following most common techniques:

- Histogram equalisation enhances contrast in images by uniformly stretching the histogram.
- Histogram matching equals the intensity distribution in an image to a reference.
- Local enhancement applies histogram equalisation and histogram matching locally.
- Gray-scale morphology are operations by which each pixel in the image gets replaces by some function of its neighbouring pixels. Neighbouring pixels is defined by a structuring element, such as a 3x3 window.

In the frequency domain we can distinguish the following techniques:

- Deblurring removes focus and motion blur.
- Frequency filtering removes noise and repetitive patterns.
- Homomorphic filtering removes multiplicative components and separates illumination and reflection.

Thus pre-processing techniques are used to alter an image to improve performance of image processing tasks. The choice of the right technique is determined by the specific application.

Appendix 5

Segmentation refers to the process of partitioning a digital image into multiple segments. The goal is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. We can distinguish the following segmentation methods:

- Clustering methods are approaches that partition an image into K clusters.
- Histogram-based methods computes a histogram of all the pixels in the image, and the peaks and valleys in the histogram are used to locate the clusters in the image. Colour or intensity can be used as the measure.
- Edge detection methods is a well developed technique within image processing and is often combined with other segmentation techniques.
- Region growing methods iteratively marks neighbouring pixels by using the intensity as measure of similarity.
- Level set methods can be used to efficiently address the problem of curve, surface, etcetera spread in an implicit approach.
- Graph partitioning methods uses pixels or group of pixels and compare their similarity to neighbouring pixels.
- Watershed transformation are using gradient magnitude intensities which represent the region boundaries.
- Model based segmentation assumes that objects of interest have a repetitive form of geometry.