Karlsruhe Institute of Technology

Humanoids and Intelligence Systems Laboratories

Prof. Dr.-Ing. Rüdiger Dillmann

Department of Computer Science

seminar
"Kognitive Automobile"
summer term 2010

Vehicle and Obstacle Detection

Sebastian Klaas

Mat.Nr.: 1343420

30. Juni 2010

I declare that I have developed and written the enclosed seminar thesis completely by myself, and have not used sources or means without declaration in the text.
Karlsruhe, 30. Juni 2010

Sebastian Klaas

Abstract: *This seminar thesis describes different approaches for on-road object detection. It mentions active as well as passive sensors and focuses on vision based vehicle and pedestrian detection using very different features like for example shape (symmetry, edges etc.) , appearance (intensity etc.) or motion (i.e. the human gait). The described approaches do all use a two or more step method containing a fast detection step and a more precise recognition step.*

Contents

1 Introduction

On-road object detection plays an important role in autonomous cars as well as in driver assistance systems. Although there has been a big deal of research and progress in recent years, real-time object detection is still a challenge we have to face.

There are many different approaches for on-road object detection, using many different sensors. Besides passive sensors (i.e. mono or stereo cameras and infrared cameras) many approaches use active sensors like radar or laser based sensors. Both kinds of sensors have their respective advantages and disadvantages, which have to be considered and are suited to different situations and goals.

Due to hard real-time constraints most current approaches to vehicle or pedestrian detection are based on two or more steps. Typically some kind of Hypothesis Generation (HG) and Hypothesis Verification (HV) (cf. [SBM04]).

The HG step consists of a fast detection phase, which generates regions of interest which are likely to contain a vehicle or a pedestrian, while the HV step verifies these regions, using some recognition techniques.

Both, pedestrian and vehicle detection, are highly affected by the variety of shapes and appearances of the objects to detect. Moreover, typical approaches to detect moving objects in video sequences like background subtraction are hard to adopt to a moving camera (which obviously results in a moving background). As we can see, the detection and recognition has to be very robust against the variety of shapes as well as changing background conditions and furthermore to lighting conditions such as shadows or low light, i.e. in tunnels or at night.

There is a variety of approaches to solve these problems. Many approaches are using vision-based sensors like color or grayscale cameras or far infrared images. Other papers propose the use of active sensors like laser scanners, radar sensors or lidar sensors. [GLT09] describes the use of near infrared images (NIR) for pedestrian detection. Some newer papers combine the advantages of different sensor technologies in order to construct a fast, robust and reliable obstacle detection system.

This paper concentrates on vision-based methods for on-road vehicle and pedestrian detection and provides an overview on different methods, which have been researched during the recent years. Section 2 discusses the pros and cons of different vision-based sensors compared to active sensors. Section 2 focuses on the common approaches for vehicle detection and classification and Section 4 discusses the different approaches for pedestrian detection and recognition in on-road systems. To conclude this paper, Section 5 summarizes what has been said before and gives an outlook on ongoing research.

2 Sensors for On-Road Object Detection: An Overview

There are different kinds of sensors which are used for obstacle detection systems in intelligent cars. Many approaches make use of active sensors like lidar and radar, while the most successful approaches are based on any kind of vision sensor.
Both kinds of sensors do in fact have their respective advantages and disadvantages, therefore they are perfectly suited to different situations and requirements.

A promising approach is to integrate different kinds of sensors in order to improve the detection and recognition of vehicles and pedestrians in different situations at night and day-time.

2.1 Active Sensors

Lidar (light detection and ranging) and radar (radio wave detection and ranging) sensors and other similar sensors are called active sensors because they emit a signal (e.g. laser or microwaves) which is reflected by the object to be detected and is recognized by the sensor again. With a kind of run-time measurement they are able to detect and range objects and are even specialized to that purpose, therefore they have a number of advantages compared to vision based sensor systems. First of all their output is 3D and needs less computational power to be processed. That would result in a lower amount of computational resources on board and save money.

The main disadvantages of lidar and radar sensors are their low spatial resolution resulting in a comparably low information density and the slow scanning speed. Due to the functionality of this kind of sensors another big disadvantage is the possible interference between the same kind of sensors on different vehicles or with other light or radar emitting sources. This last disadvantage isn't that much of a problem concerning prototypes of intelligent vehicles but it becomes more and more important in a transportation system containing many intelligent vehicles crossing and following each other.

2.2 Passive Sensors

Passive sensors provide more general sensor data to the ECUs (electronic control units), therefore those sensors need more complex algorithms to gather useful information, which results in a considerably higher need of computational power. On the other hand the sensors itself are quite cheap compared to active sensors. All kinds of optical sensors are superior to passive sensors when tracking cars in complex situations because they offer much more information on the objects than a lidar or radar would do.

Vision based sensors are not only useful in order to detect vehicles or pedestrians, but also in many other areas of DAS (driver assistance systems) containing lane detection or detection and recognition of traffic signs. They are not only able to detect and recognize objects, but also to identify and classify them.

As mentioned before object detection and recognition with vision based sensors is

a very challenging task to real-time algorithms as well as computational resources on board. They are often affected by bad or changing lighting conditions, cluttered backgrounds, poor visibility and the variability in human and vehicle shape and appearance.

There are different kinds of passive sensors, which play a role in the field of object detection. Cameras can be grayscale as well as color cameras and there are special low light cameras or infrared cameras (FIR (far infrared cameras)). A more complex approach using vision sensors is the stereo camera approach. Some papers do also propose camera system based on more than two cameras.

2.2.1 Monocular Camera

Monocular (typically grayscale) cameras are the most common sensors within vision based approaches, they are very cheap and need no calibration, therefore they are comparably easy to install and use. On the other hand they provide much information which has to be processed in any way in order to extract useful information. Today this work is still done on separate embedded systems with a huge amount of computational power, which makes those systems more expensive again.

As they are based on visible light, monocular cameras are heavily affected by poor visibility for example due to rain or fog. Poor lighting conditions or difficult backgrounds do also make the task of object detection a more difficult one. Some of those problems can be solved with specialized cameras such as low light cameras, others have to be solved using more robust and time consuming algorithms.

2.2.2 Infrared Camera

Far infrared cameras (FIR) show sources of heat, based on the infrared light emitted by human bodies or cars. Unfortunately human bodies as well as cars do not emit heat in the same intensity at every part of the body or vehicle. For example the human head has a much bigger intensity than his arms. Another big problem with infrared technology is the fact that humans and vehicles are not the only objects in the cars environment emitting heat. Houses, light bulbs and other heat emitting objects may be a problem for recognition tasks. Moreover hot environments like deserts may actually emit nearly as much heat as the human body, therefore the correct detection of the object in question becomes more difficult again.

There is another problem concerning FIR, since cheap infrared cameras are uncooled and may therefore produce much noise, which heavily affects the detection rate.

Due to the characteristics of infrared technology, infrared cameras have special advantages concerning night vision and poor visibility in general. They are less affected by poor visibility than cameras operating with visible light, hence they are superior at night-time or in rainy or foggy environments.

2.2.3 Stereo Camera

Pairs of stereo cameras provide the possibility to measure distance information - a task that can not be accomplished with a monocular camera (respectively only with accurate

model knowledge). Therefore they enable us to use this information for example to extract foreground objects in a video sequence. However stereo camera systems have one big disadvantage: they require calibration, which is a challenge on its own, particularly in on road situations with a moving camera which is affected by vibrations and windy conditions, thus needing self-calibration in order to recover extrinsic camera parameters.

Moreover they have the same disadvantages monocular cameras have in terms of night vision and poor visibility. Additionally stereo camera systems do, because of more complex algorithms for information processing, typically need more computational power than monocular cameras or infrared systems.

3 Vehicle Detection

The main aim in vehicle detection is of course to build highly reliable collision avoidance systems. Therefore most systems concentrate on front view camera systems in order to detect possible rear-end-collisions. However, vehicle detection is theoretically able to detect all kinds of possible collisions and can therefore be used for different kinds of driver assistance systems. Rear vehicle detection is for example used for a lane change assist system in [LWD+07]. Other systems may try to detect overtaking cars to the left or right hand side of the vehicle.

As mentioned before most of the vehicle detection approaches in recent years are based on a two or more step scheme in order to improve real-time capabilities of the highly safety critical systems. One single recognition step on the whole recorded video picture is a time consuming task in particular with the comparably low computational power available on board. Thus the following review is divided in two main parts: Hypothesis Generation and Hypothesis Verification. The former is based on a fast and less precise algorithm in order to find possible objects, which results in a collection of candidates of which some will later on be validated as vehicles in the verification phase and others will prove to be false positives.

3.1 Hypothesis Generation

3.1.1 Motion-based Approaches

Motion may be one important cue for HG phase. Although they are obviously not able to detect static vehicles, which can nevertheless be a big threat, different approaches using optical flow have been developed. Problems with this kind of approaches are the difficult tasks of finding appropriate features and tracking those features in consecutive frames. It is of course impossible to compute a motion vector for each pixel under real-time constraints. A further challenge is the motion of the camera itself, which has to be considered in any kind of on-board vehicle detection system because it results in background motion.

Koller et al. propose a solution based on so-called blobs (connected regions) in [KHN91]. Their approach is a multistage approach, first of all extracting features in every single frame, followed by matching those features (i.e. the centroids of blobs, which represent local minima or maxima of the grayscale value) in consecutive frames to gather displacement vectors and last but not least they cluster nearby, parallel vectors of the same length in order to find possible objects. Unfortunately *Koller et al.* did not optimize their approach to real-time constraints, hence the estimation of the displacement vectors took about 45 sec. on a Sun workstation back in 1991. However an optimized version of their algorithm combined with state of the art technology may well be considerably faster.

Another approach on optical flow based vehicle detection from *Heisele et al.* uses color blobs. Their method, which is described in [HR95], is in contrast to [KHN91] developed for the aim of vehicle detection from a moving car. The first step of this

approach is a color segmentation step, clustering the frame pixels into color clusters based on 16 reference colors. The second step is a connectivity analysis step, computing connected areas of the same color along with their bounding box and centroid as well as a list of neighbors for each of them. Afterwards they match adjacent blobs in consecutive frames, based on color, blob area and the aspect ratio of the bounding box, this results in displacement vectors for every blob. The last step is to combine blobs with similar motion into motion segments, which indicate an object. This approach seems to produce relatively promising results and all steps are optimized to match real-time constraints.

A third approach using optical flow is described in [OTFO03] and is based on three horizontal line segments. *Okada et al.* compare the motion of those segments with the motion of the ground plane and the vehicle surface to detect vehicle candidates. The method is based on a motion constraint, which is computed for both the ground plane and the object surface. It is basically the cross ratio of four lines (the three line segments and the respective vanishing line of the ground plane ($y = 0$) or the object plane ($y = \infty$), which has to be constant in time if all of the four lines belong to the same plane. A group of lines is considered as an object if those lines satisfy the object motion constraint better than the ground plane motion constraint. According to *Okada et al.* their method is both fast and robust. Their experiments show promising results for vehicles which are not too far away from the camera. They achieved fast detection rates (20 - 100 fps).

3.1.2 Shape-based Approaches

This second category of HG approaches is probably the most important in on-road vehicle detection. Shape-based approaches (often referred to as knowledge-based approaches) play a major role in recent research projects and are typically based on shape information, such as edges, symmetry, color, corners or shadows. They make use of a-priori knowledge about the shape of vehicles in order to find typical patterns in a single image.

Color information for example could be used for HG in order to detect non-road areas, although this cue has not been used in a lare number of real projects. The main problem with color information is its dependence on too many factors like illumination, reflectance etc. (cf. [SBM04]).
Another cue which is difficult to exploit is symmetry, because background symmetry and occlusion of vehicles cause false detections (cf. [SBM04]).

Bertozzi et al. developed a corner-based approach. Their main thought is based on the rectangular shape of vehicles [BBC97]. They use templates in order to detect all corners in an image and try to match them to detect rectangular shapes, consisting of four different corners. According to their own tests, their not yet optimized implementation works quite well and relatively fast (about 4 fps).

More promising and more often used techniques are based on edges or shadows. *Srinivasa* describes an edge-based approach in [Sri02], which extracts horizontal and vertical edges in every single frame with the Sobel operator. Edges which exceed a certain size are combined to connected regions. A priori knowledge of vehicle edges is used to make a first decision whether edges belong to a vehicle or to the road. *Srinivasa* claims this

algorithm to be both robust and of low complexity, thus this algorithm is perfectly suited for on-road vehicle detection under real-time constraints.

Sun et al. use edges as a cue in their multiscale-driven approach described in [SBM06]. Due to the problems concerning the choice of a optimal parameter set, which may not be constant in every environment, they decided to subsample and smooth the video frames to 3 different resolutions. Beginning with the coarsest level of detail, they search for local maxima in vertical and horizontal edges and track those maxima down to the finest level. They generate hypotheses based on one horizontal maximum in combination with two vertical maxima. This multi-scale approach is said to be more robust concerning the choice of parameters and considerably faster than other approaches.

The use of shadows is widely spread around vehicle detection projects. The area underneath vehicles is typically darker than the surrounding environment. Unfortunately the intensity of those shadows depends on the illumination of the scene, therefore it may change due to weather conditions or at night-time and depends on the road color. That of course causes changing thresholds for shadow- based detection methods.

One shadow based approach is described in [Buc]. In order to deal with the difficult choice of thresholds, *Buchman* basically tries to estimate the threshold with an analysis of the road color, based on maximum likelihood estimation. Once the algorithm decides for a threshold, it clusters different shadow segments together, each cluster defining the base-line of a vehicle. In order to make his approach more robust to false shadows from bridges or buildings *Buchman* relies on a road detection system, which provides road borders for his algorithm. After applying further geometrical constraints and generating a bounding box for each vehicle (based on the shadow as a base-line and a priori aspect ratios), this box is proposed as a possible vehicle. *Buchman* tested this approach in an urban environment with many bridges and reached a degree of recognion rate of 80%.

[LWD+07] describes another shadow based approach, which does not rely on any road or lane information. To find shadows underneath a vehicle, *Liu et al.* propose a bottom up scan for a bright-to-dark transition in gray values. The transition lines are regarded as potential lower edges of vehicle regions. Those edges are matched with dark regions and filtered with perspective constraints. A region of interest is generated by the use of a bounding box, which can bee seen in Figure 3.1, based on aspect ratios as in *Buchman's* approach.

Figure 3.1: Gradient image of the sahdows underneath vehicles (a) and corresponding ROIs (b); from [LWD+07].

Since *Liu et al.* use gradients to find shadowed regions, this approach is able to find possible candidates under daylight as well as difficult conditions concerning lighting, tunnels etc. They reached a detection rate which is constantly higher than 95% and a false alarm rate of less than 4% with a performance of about 25 fps on a standard PC (Pentium IV).

3.1.3 Stereo vision-based Approaches

Stereo vision-based approaches use either disparity maps or IPM (inverse perspective mapping), in order to detect vehicles. Both kinds of approaches are very sensitive to extrinsic camera parameters and require calibration in order to make exact decisions. Unfortunately, vibrations as well as windy conditions may affect the camera parameters, hence robust methods need on line recalibration of the stereo camera system.

Disparity maps (which represent the horizontal displacement between the two images in every single corresponding pixel, which is inversely proportional to the distance between the camera and the object represented by the pixel) provide enough information to generate 3-D maps of the scene, therefore they are a very powerful tool. In [Han97] *Hancock* found a way to use disparity with comparably low computational complexity. This approach computes disparity by making the assumption that roads are (nearly) planar. With this assumption *Hancock* found a linear function of the image row, which represents disparity. By subtracting the warped right image from the original left one, he gathers a difference image in which pixels belonging to the background disappear and objects can be detected quite easily.

The other possible technique to exploit stereo vision for the aim of vehicle detection is inverse perspective mapping. IPM, meaning the geometrical transformation of one image from image coordinates into real world coordinates and the elimination of the perspective effect, can be used to find objects above the road plane. Under plane road assumption, the pixels which belong to the road plane are mapped to the same position in the remapped right and the remapped left image, while objects above the road plane result in non-zero pixels in the difference image of the two remapped images. *Zhao et al.* use IPM to transform the left image to world coordinates and furthermore to image coordinates of the right camera. The computed and the actual right image are compared in order to detect objects. As a additional cue they use the Sobel filter to detect edges in the original right image. Their approach is described in [ZY93] and they claim their results to be very promising in terms of usefulness for intelligent vehicles.

Bertozzi et al. describes a different approach using IPM in [BB98]. In contrast to *Zhao et al.*, they remap both pictures with the use of IPM to the world domain and compute the difference between the two remapped pictures. Non-zero pixels in this difference picture are clustered and do possibly show obstacles. The whole processing (together with lane detection) takes about 10 fps on the test system and shows good results for mid-range obstacles (5-45 m).

Since all of these stereo vision approaches reduce the problem of finding vehicles to the detection of free space instead of using any kinds of templates or a priori knowledge about vehicles, they can also be used to detect general obstacles, i.e. not only vehicles.

3.2 Hypothesis Verification

3.2.1 Template-based Approaches

Template-based hypothesis verification uses templates or patterns, which were computed previously, in order to verify the regions of interest, which were produced by the HG step. Those patterns are typically based on a priori knowledge in terms of shape features of vehicles like symmetry, rectangular form etc. Correlation between original image and template is used to decide whether an object matches the templates or not.

A template-based approach was used by *Betke et al.* to detect multiple cars in a gray-scale video. In [BHD96] they describe their approach, which after detecting a possible car estimates its size and correlates the image of the car with a template generated from a precomputed model deformed to the estimated size. The choice of the model to compare with, is based on the grayscale value in the ROI. If the recognition steps result is not clear enough, the object is tracked over a couple of frames in order to repeat the recognition and come to a definite conclusion.

In [FWSB95] another approach, based on deformable templates is presented. *Ferryman et al.* use PCA (principal component analysis) descriptions of their data, what enables them to differentiate between sub-classes of vehicles instead of differentiating only between vehicles and non-vehicles. *Ferryman et al.'s* model is based on 29 independent parameters and describes all kinds of vehicles. Using PCA, they are able to represent more than 97% of the variance in their training data by using only 6 eigenvectors, which can bee seen in Figure 3.2. Unfortunately this approach is by now not very precise, the vehicle classification is quite good on the training data, whereas test data did not show comparably good results. However, there is some space for improvements, since *Ferryman et al.* labeled the training data for the different sub-classes based on their own estimation without the use of expert knowledge. Another question which was not answered is, whether this approach is actually real-time capable or not.

3.2.2 Appearance-based Approaches

Appearance-based approaches use machine learning to distinguish between a vehicle-class and a non-vehicle-class. Classifiers like neural networks or SVMs are trained using images of vehicles and maybe examples of the non-vehicle-class as well. Another pattern classification method which is used in many approaches, is the modeling of the per class probability density functions for image features. The biggest problem for classification methods is the extraction of features, which are capable to describe both classes (vehicle and non-vehicle) in a sufficient way. The goal is of course to reduce the dimension of the feature space, which allows faster classification.

One possible method to achieve this, is the use of PCA for feature extraction, which projects the huge feature space into a low-dimensional space, spanned by the principal components of the training data. Another approach is the use of Wavelets (e.g. Haar-Wavelets) which provide a compact representation and can be computed using fast algorithms. Gabor features are a third kind of features for appearance-based approaches. Gabor features represent edge and line information, which makes them a strong cue for

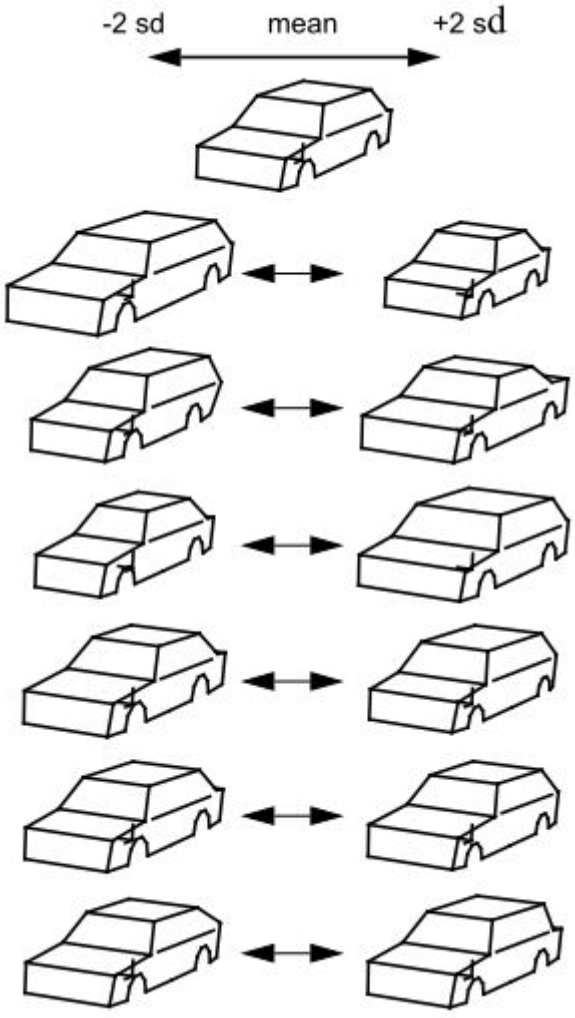

Figure 3.2: Deformable model based on the first 6 eigenvectors. The top vehicle represents the mean, each other illustrates vehicles which are displaced from the mean in the direction of one eigenvector; from [FWSB95].

vehicle detection.

[LWD+07] describes the use of SVM classification and Haar wavelets based on more than seventeen thousand training images (vehicles and non-vehicles) taken at different daytime and in different situations.

Wu et al. use a PCA classifier. In [WZ01] they describe their approach based on separate PCA analysis for every class.For the purpose of classification, new samples are projected on both PCA subspaces. To decide which class is the correct choice, they do also project the samples into the subspaces spanned by the eigenvectors which were not chosen as principal components. The norms of this projections are compared and the smaller one (i.e. the smaller error) is chosen as the correct class. On a car vs. non-car classification problem their PCA classifier performed better than a SVM as well as Fisher Linear Discriminant classification.

Gabor features are used in *Sun et al.'s* approach and combined with SVM. In [SBM02] they describe their approach, using local Gabor features, which are obtained by dividing the original image into subimages and running a Gabor filter on these subimages. A SVM classifier is trained by the collected features and used for classification. In [SBM06] they compare experimental results achieved with different features (as for example the Gabor features described above), using neural networks as well as SVMs. Throughout their experiments, SVMs performed far better than neural networks. According to these results, Gabor features and wavelets (local features) outperform PCAs, this may result from non-normalized image data. In fact, PCA seems to have a lack of robustness, if vehicles are changing their position and scale. Another result of their experiments was that combinations of different features perform better than single features. Unfortunately concatenation of different features exceeds real-time constraints.

Schneiderman et al. describe a statistic-based approach in [SK00]. They compute the conditional probabilities for the image, given each of the classes and the ratio between those two probabilities. This ratio is compared to the ratio of the a priori probabilities of both classes and used to classify the objects (Eq. (3.1)). The modeling of the likelihoods is based on different histograms, each describing one visual attribute, which are considered as independent. The likelihood of a image given a class is defined as the product of all likelihoods for the different attributes given the class (Eqs. (3.2), (3.3)). *Scheiderman et al.* achieved a detection rate of approximately 90% on their test data.

$$\frac{P(image|pedestrian)}{P(image|non-pedestrian)} > \frac{P(non-pedestrian)}{P(pedestrian)} \tag{3.1}$$

$$P(image|pedestrian) = \prod_k P(attribut_k|pedestrian) \tag{3.2}$$

$$P(image|non-pedestrian) = \prod_k P(attribut_k|non-pedestrian) \tag{3.3}$$

4 Pedestrian Detection

Pedestrian detection makes use of very much the same methods as vehicle detection does. The main difference between both techniques is probably the choice of features. Pedestrian detection as an additional method to pedestrian safety (in addition to recent improvements concerning passive safety) is maybe an even more safety critical task than vehicle detection, not only because humans are more vulnerable, but also because their visibility is worse than that of cars.

As with vehicle detection, feature selection is very important to build a highly robust and reliable pedestrian detection system. However, it is still difficult to decide which features to use in order to model the huge within class variability pedestrians have. Not only the size of people and the color of their clothing may differ, but in contrast to vehicles we do also have to consider different poses. Pedestrians may not only walk along a street but across the street and are therefore seen in a lateral view. Moreover one goal for pedestrian detection systems is to detect walking or standing humans as well as those who ride a bicycle.

Due to real-time constraints most pedestrian detection systems do also use two-step approaches using a fast hypothesis generation step to find ROIs (regions of interest) and a second hypothesis verification step to verify those ROIs. Different approaches for both steps are described in the following sections.

4.1 Hypothesis Generation

4.1.1 Motion-based Approaches

As in vehicle detection, motion can also be a strong cue for pedestrian detection. Unfortunately we do not only have the same advantages but also the same disadvantages as in vehicle detection: Motion-based approaches are not able to detect upright standing pedestrians. Most work, which is based on motion detection relies on the periodicity of the human gait, thus those approaches do often assume pedestrians walking laterally to the camera's viewing direction. Motion-based approaches can generally be divided into two different techniques. One that analyzes motion on pixel level and one that analyzes motion on a higher level considering areas or objects (i.e. clusters of pixels). The biggest advantage of the latter method is, that they are also able to detect pedestrians walking along the optical axis of the camera, while pixel-based methods require more pixel-variation which is typically not supplied in such scenes.

Cutler et al. use object-motion in their approach described in [CD99]. The first step of their method is motion segmentation. This step finds and segments motion regions by first stabilizing the scene (correcting affine transformation due to camera movement, cf. [HAD+94]) and then subtracting consecutive frames. Motion regions are then detected based on simple thresholding. The objects, which are found by this algorithm are afterwards tracked in order to detect and analyze periodicity, based on the similarity of an object at two different times t_1 and t_2. More precisely periodicity is indicated by the dark lines in the similarity plot (see Figure 4.1) which are parallel

to the diagonal of the plot. This approach has been implemented and tested on real-time capabilities, which were approved and it is moreover capable to detect not only pedestrians walking laterally to the cameras viewing direction, but as well those, walking along the optical axis or in other directions.

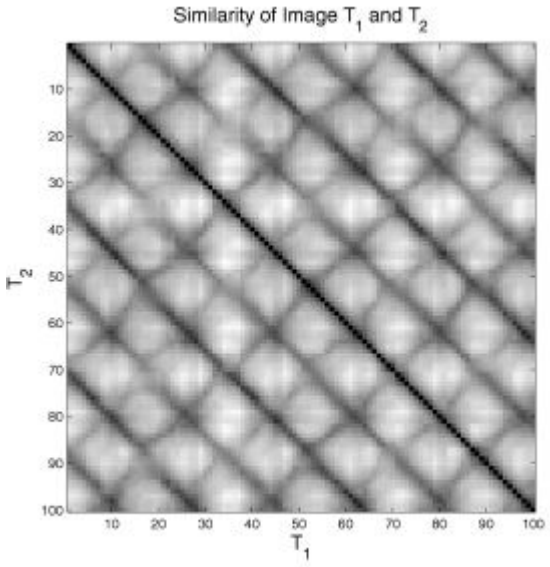

Figure 4.1: Similarity plot for a walking pedestrian; from [CD99]

[TSKK94] describes an approach which relies on motion parallel to the image plane. Basically they try to find cycles in the trajectories of single points on the human body or other moving objects. *Tsai et al.* consider those trajectories as spatio-temporal curves. They compute the auto-correlation of the curvature function in order to find self-similarities which may indicate a cycle. Afterwards a Fourier transform is performed on this autocorrelation function. A large impulse at the cycle frequency will occur in the Fourier transform, if there are any cyclic movements performed by the object of interest. In order to meet real-time constraints, the Fourier transform is computed using the Fast Fourier Transformation (FFT), with this optimization the algorithm needs less than 0.8 s for a video sequence consisting of 512 frames (640 fps).

4.1.2 Appearance and Shape-based Approaches

In contrast to motion-based approaches, this second line of methods does not rely on a sequence of images but on single images. Within those images pedestrian detection is based either on shape-information like for example edges and symmetry or on the whole appearance of pedestrians in terms of intensity in FIR images. Both are strong cues for pedestrian detection, using a priori knowledge in order to reduce computational complexity. Vertical symmetry for example is a typical shape feature of pedestrians in front or rear view, a pedestrian is an object containing mostly vertical edges (body, legs, etc.) and size of human bodies or aspect ratio can be an additional hint. Intensity

in the infrared image, meaning body heat, is of course very high in regions containing humans. One problem of this feature is that other objects like for example cars (i.e. engines, brakes) or light bulbs do as well emit heat. Other cues are needed to confirm the hypothesis. Moreover infrared-based approaches face problems in very hot environments., where it is a hard task to distinguish between hot foreground objects and the hot background.

In [ND02] *Nanda et al.* present an intensity-based real-time pedestrian detection algorithm. They use a pixel-based representation and detect objects which probably are human by simple intensity thresholding. The threshold is defined by the respective mean and standard derivation of the pedestrian and non-pedestrian class in a set of training images, using Bayes classification. The list of detected objects will of course not only contain pedestrians but as well other heat emitting objects, which have to be eliminated in the verification step.

Another approach using infrared images for pedestrian detection is described in [ZTH07]. *Zin et al.'s* approach uses shape-information in addition to intensity thresholding, in order to extract possible head regions of persons. They use a rough estimation, considering human heads as ellipse shaped regions in the threshold image.

Xu et al. do as well use thresholding to detect possible pedestrians in an infrared image in their approach, described in [XF02]. Their threshold is computed, based on the images mean intensity and the highest possible intensity in the system. The segmented regions are then constrained according to their size, aspect ratio and position in the picture, in order to eliminate a huge amount of false positives. Further elimination of false positives is done, by searching for head regions (i.e. disc shaped regions) in the segmented areas.

An approach based on visual sensors instead of infrared images, is described in [BBFS00]. *Broggi et al.* use shape information like vertical symmetry in their pedestrian detection and recognition system for the ARGO vehicle. Besides vertical symmetry they make use of vertical shapes, size, aspect-ratio and again the position of the pedestrian in the frame. First of all they do only consider possible pedestrians in the part of the picture which seems to be of interest (due to practical considerations (i.e. pedestrians are only interesting in a certain distance to the vehicle) and perspective). Afterwards vertical edges are extracted using a Sobel operator. In addition they make use of stereo vision in order to eliminate background objects before detecting symmetry with respect to the vertical axis. More false positives are detected by adding information from a horizontal edges map.

4.1.3 Stereo vision-based Approaches

Range or stereo vision-based pedestrian detection is a method which is less affected by the variability in human appearance or other difficulties like for example shadows or lighting conditions in general. Moreover it is perfectly applicable to foreground segmentation.

Unfortunately what has been said in the vehicle detection part is still applicable to pedestrian detection. Camera calibration and on line self-calibration of the cameras mounted on driving vehicles have to be considered in any stereo vision-based detection system. Unfortunately calibration increases the computational effort for stereo vision-

based approaches.

[WAPF98] uses stereo vision for the detection step. Based on the stereo image pair, a pair of feature images is generated. Each pixel is compared to the grayscale value of its 4 neighbours, this method results in 81 different classes and feature images describing edges and corners. Afterwards epipolar geometry is used to perform a correspondence analysis between those feature images, which results in a disparity map of the scene. The next step is the generation of a 2D depth map, based on the disparity information. This depth map shows the bird's eye view of the scene, which contains all objects above the ground plane, thus allows easy detection of pedestrians as well as other objects. Those detected objects are evaluated with regard to their position, size and motion and constitute the ROIs for the recognition step.

Zhao et al. describe their stereo vision-based approach in [ZT00], their system first computes area-based disparity maps before performing simple range thresholding in order to eliminate background objects. Connected regions are grouped together based on range information and constrained with regard to their size, in order to eliminate objects which are too small to be human. The result of this step can bee seen in Figure 4.2. Areas that are too big for a pedestrian, are searched separately in order to divide them into smaller areas containing pedestrians. The bounding boxes of the hypothesized pedestrians do then form the input data of the following recognition step.

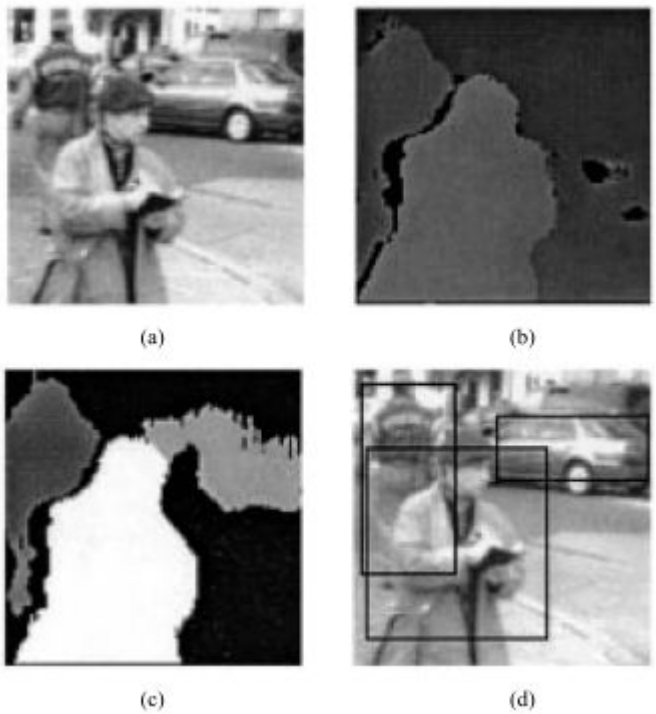

Figure 4.2: Different steps to pedestrian detection: input image (a) and computed disparity map (b); the segmented regions (c) and bounding boxes of detected objects (d); from [ZT00]

In [LF04] a stereo night vision system is presented. *Liu et al.'s* approach does first of all detect hot-spots in both the left and the right image using thresholding. Afterwards stereo information is used in order to compute camera motion (translation and rotation). The information, gained in the last step is then used to filter object movement from camera movement before detecting moving objects in the scene. This approach actually combines stereo vision and motion-based approaches in order to detect pedestrians. Using the stereo vision information does of course provide possibilities to improve the motion-based techniques and in contrast to some stereo vision-based approaches this combined approach may detect vehicles as well as pedestrians and classify both. On the other hand disadvantages of normal motion-based methods, like for example difficulties in detecting pedestrians walking along the optical axis, are still a problem.

As seen before stereo vision may as well be used as an additional hint in basically shape-based approaches. *Broggi et al.* use stereo vision in order to reduce the computational complexity of their approach ([BBFS00]) by eliminating background objects. To achieve this goal, they shift one image, using an offset computed with the camera system parameters. They assume, that object in far distances ("infinite") are then overlapped in both pictures and can be eliminated using subtraction.

4.2 Hypothesis Verification

4.2.1 Motion-based Approaches

While motion is not a common and maybe not a very good cue for hypothesis verification in vehicle detection, there are actually quite a few approaches for pedestrian verification using motion patterns. Those approaches use temporal information in order to eliminate false positives which made their way through the detection step. Very much like the detection approaches which use motion, the verification steps do typically rely on periodic motion like for example the periodicity of the human gait, in order to distinguish humans from other moving objects like for example vehicles. Basically the human gait is quite a good cue to reliably detect walking pedestrians. However there are different disadvantages, first of all it is impossible to detect stationary humans and difficult to detect humans performing unusual movement (i.e. cycling). Furthermore, which is a general problem of motion-based approaches, it is necessary to analyze a whole sequence of frames, which makes it harder to meet real-time constraints.

Woehler et al. use a time delay neural network in order to detect laterally walking pedestrians. Their approach, described in [WAPF98], does only consider the lower half of the ROI, where the legs are expected. To analyze one whole footstep, they consider a series of 8 images, which cover an interval of about 640 ms in length. The feed-forward time delay neural (see Figure 4.3) network's (TDNN) three dimensional input fields are fed with this image series. The second layer of the TDNN is working as a set of filters, each branch in the second layer consists of three-dimensional, spatio-temporal neurons, each of which filters its respective, so-called "receptive field", a small subset of the input layer. The filter coefficients are the weighting parameters of the neurons, which are learned from the training data. The output layer of the TDNN consists of

one row of neurons for each class to detect (i.e. there are 2 of them in case of pedestrian vs. non-pedestrian detection). The actual output neurons compute the sum of the single neurons in each row, which are connected to a series of neurons of the second layer, performing motion pattern detection on a short sequence. The performance of

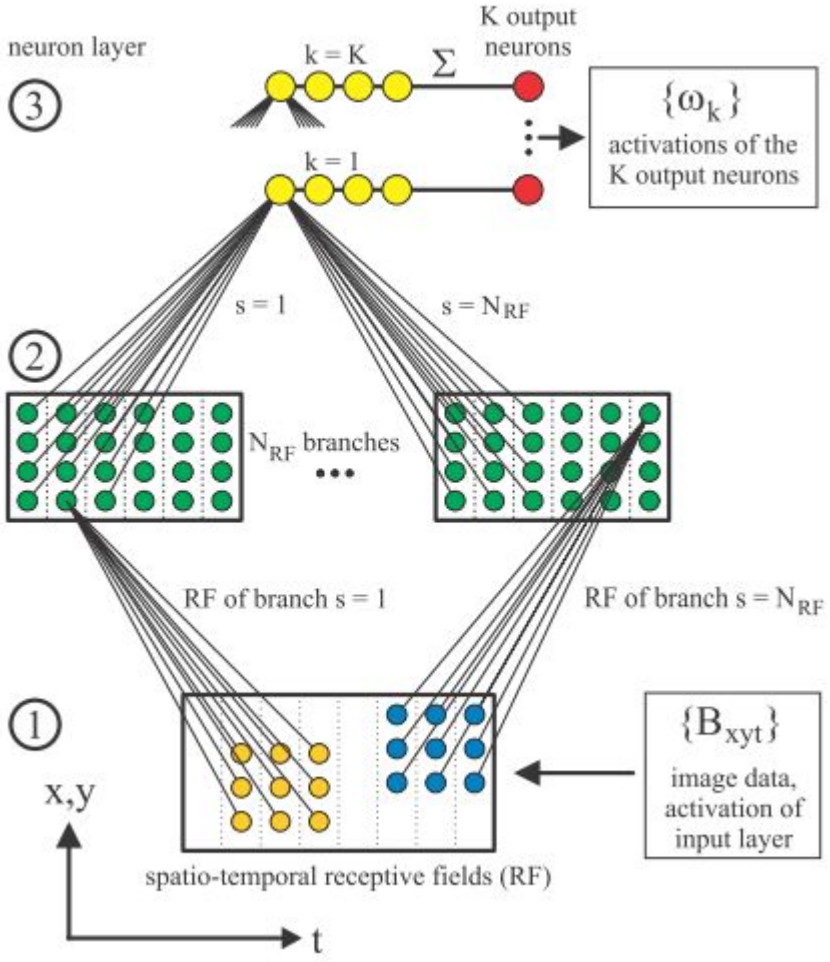

Figure 4.3: Architecture of an Time-Delay Neural Network for motion-based pedestrian detection; from [WAPF98]

this approach depends very much on the selection of its parameters (i.e. length of image series, number of branches on the second layer etc.). For their own test system *Wöhler et al.* achieved a recognition rate of 85.8% with a false positive rate of 1.6%. This proved to be sufficient, when considering a couple of seconds of video sequences. Moreover their system proved to be robust against difficult lighting and conditions and complex motion.

In [YM94] *Yasutomi et al.* perform pedestrian recognition by estimating temporal and spatial frequency of human footsteps from the change of the sum of intensities in a small window, expected to contain the feet of a person. Those frequencies are then compared

to a model and classified as pedestrian or non-pedestrian, according to whether they match the model or not. Experiments showed a detection rate of about 92% with a false positive rate of 0%, based on a test dataset containing pedestrians as well as cyclists and dogs as non-pedestrians.

4.2.2 Appearance and Shape-based Approaches

Shape and appearance-based verification approaches are maybe the most common line of techniques for the recognition of pedestrians in ROIs detected by a HG step. In contrast to motion-based approaches, they do not require temporal information and operate on single images, which provides two important advantages: The first one being the ability to detect stationary as well as moving pedestrians, the second one being the easier achievable goal of real-time capabilities. Disadvantages of shape- as well as appearance-based approaches are again the big effect of illumination, shadows and other influences on the detection rate.

In [ZTH07] an approach is described, which first of all estimates the persons leg and body regions and decides on the classification based on the respective histograms of both regions (in infrared images). If the local maxima of both histograms satisfy certain conditions (basically thresholds), the object is classified as pedestrian.

Broggi et al. present a system based on visible images instead of infrared images. Their approach, described in [BBFS00] uses a priori knowledge on shape information to verify ROIs generated in the detection phase. Those are checked against empirically determined width and height values of humans. Furthermore the aspect ratio of the bounding box is used to eliminate false positives. The last decision is based on the object's entropy and as well compared to previous frames in order to recognize pedestrians detected in previous frames, based on the position of the bounding box. *Broggi et al.* tested their system on different scenes containing pedestrians as well as other objects like cars, motorbikes and groups of overlapping pedestrians. The algorithm proved to be robust and makes a good distinction between pedestrians and other objects and last but not least the recognition worked for far as well as near pedestrians. The use of temporal information considering subsequent frames worked quite well to eliminate false positives not detected in the previous steps.

In [XF02] an SVM is used to perform pedestrian recognition in the HV step, using real videos but concentrating on the first 5 frames of each packet of 25 frames. *Xu et al.* tested their SVM on both grayscale and binary images. Their experiments showed good results using grayscale images, while binary images proved to be of no use for this aim, the detection rate was far too low, when using binary images. They made more comparisons using both whole body candidates and candidates consisting only of the upper half of the body. Both kinds of candidates showed nearly the same detection rates, though whole body candidates needed a higher number of support vectors, thus showing a lower efficiency. Moreover they compared two methods, the first one classifying all three kinds of pedestrians (pedestrians walking along-street, pedestrians walking across-street and cyclists) with one SVM or with different SVMs. The first approach has a huge number of support vectors, needs a longer training time and was slower than the second one.

To test their system, they used a set of 16 video scenes, 10 of them as training data and the rest as test data. The scenes contained both summer and winter scenes and pedestrians as well as cyclists. Both kinds of classifiers showed a huge detection rate but the single-classifier approach showed also a huge number of false positives. Nevertheless it proved to be the better approach due to speed considerations

4.2.3 Template-based Approaches

Precomputed templates or patterns, which describe the variability in human shapes, poses and appearances, are another tool for pedestrian recognition. In contrast to appearance-based approaches they do not that much rely on trained classifiers, although they do of course need some kind of training. Template-based approaches compare the templates with every subwindow of the ROIs in order to find matching patterns.

Nanda et al. present a probabilistic template for pedestrian detection in [ND02]. They developed this template from normalized threshold images of persons in various poses. The 128x48 pixels template itself consists of a probability $p(x, y)$ for each pixel, which represents the number of training images, in which this pixel belongs to the pedestrian. The test windows are then (after normalization and thresholding) classified with this template. For each pixel, $p(x, y)$ is the probability that it describes a pedestrian if its value $th(x, y)$ is 1 or $1 - p(x, y)$ if its value is 0. This method can be described with Eq. (4.1), computed for each pixel (i, j), describing the center of an 128x48 subwindow of the image.

$$prob(i,j) = \sum_{y=1}^{128} \sum_{x=1}^{48} (th(x,y) * p(x,y)) + (1 - th(x,y)) * (1 - p(x,y)) \tag{4.1}$$

Thresholds do then decide, which coordinates are considered as centroids of pedestrians. Experiments showed, that this method is relatively robust against occlusion and noise, detection rates between 75% and 90% are achieved.

5 Conclusion

To summarize what currently happens on the field of on-road object detection one would probably say that it is still a big challenge to develop a fast, real-time vehicle and pedestrian detection systems which is both robust and highly reliable. Future approaches will become more and more complex, relying on an increasing number of different sensors and sensor fusion is going to be more important. [WL07] describes an approach using laser scanners as well as color video cameras and FIR. Approaches like this one promise a wider range of information to be exploited but also a higher reliability due to redundancy. Laser based sensors are for example perfect to support visual sensors in times of bad visibility and FIR sensors are perfectly suited for night-time of foggy environments.

Another key factor in object detection is feature selection. A wide range of features is actually used for the aim of vehicle and pedestrian detection and all of those have their respective advantages and disadvantages. However we will probably not be able to find one feature suitable for all situations and therefore should exploit as many different features as possible in order to create robust and reliable techniques.

On the other hand the combination of different sensors and features does not support real-time capabilities of the developed systems, therefore much effort is needed to provide fast fusion and classification techniques on different abstraction layers.

Another topic for future research is the combination and integration of different driver assist systems. Nowadays lane detection, pedestrian detection, vehicle detection and other detection algorithms are developed separately. Although all of those systems seem to have different requirements, the approaches are not too different and a higher degree of connection (i.e. combined low-level feature extraction) may save computational time. Moreover similar tasks for different DASs could be dedicated to the same piece of hardware, which is optimized for this kind of work. Anyway a higher degree of optimized hardware implementation instead of software on standard hardware could be worth a try, in order to increase efficiency and achieve a higher degree of real-time capabilities.

Another possible improvement in terms of computational effort is the use of intelligent sensors, which do a big part of preprocessing on integrated micro-controllers or other specialized hardware, which is integrated in the sensor and optimized to fast preprocessing or feature extraction.

Altogether there are many possible improvements, which may be researched in future and which provide possibilities to increase detection speed as well as detection rate. But all improvements will be linked up with an increasing complexity and may require a higher computational power on board. Today there is no vehicle or pedestrian detection system which proved to be good enough for a completely autonomous vehicle interacting with normal cars as well as pedestrians in real urban environments and with normal driving speed.

List of Figures

References

[BB98] M. Bertozzi and A. Broggi. Gold: a parallel real-time stereo vision system for generic obstacle and lane detection. *Image Processing, IEEE Transactions on*, 7(1):62 –81, jan 1998.

[BBC97] Massimo Bertozzi, Alberto Broggi, and Stefano Castelluccio. A real-time oriented system for vehicle detection. *J. Syst. Archit.*, 43(1-5):317–325, 1997.

[BBF00] M. Bertozzi, A. Broggi, and A. Fascioli. Vision-based intelligent vehicles: State of the art and perspectives. *Robotics and Autonomous Systems*, 32(1):1–16, 2000.

[BBFS00] A. Broggi, M. Bertozzi, A. Fascioli, and M. Sechi. Shape-based pedestrian detection. In *Intelligent Vehicles Symposium, 2000. IV 2000. Proceedings of the IEEE*, pages 215 –220, 2000.

[BHD96] Margrit Betke, Esin Haritaoglu, and Larry S. Davis. Multiple vehicle detection and tracking in hard real-time. In *IEEE Intelligent Vehicles Symposium*, pages 351–356, 1996.

[Buc] Gilad Buchman. On Road Vehicle Detection using Shadows.

[CD99] Ross Cutler and Larry Davis. Robust real-time periodic motion detection, analysis, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:781–796, 1999.

[FWSB95] J M Ferryman, A D Worrall, G D Sullivan, and K D Baker. A generic deformable model for vehicle recognition, 1995.

[Gav01] Dariu M. Gavrila. Sensor-based pedestrian protection. *IEEE Intelligent Systems*, 16(6):77–81, 2001.

[GLT09] Junfeng Ge, Yupin Luo, and Gyomei Tei. Real-time pedestrian detection and tracking at nighttime for driver-assistance systems. *Trans. Intell. Transport. Sys.*, 10(2):283–298, 2009.

[GM07] D. M. Gavrila and S. Munder. Multi-cue pedestrian detection and tracking from a moving vehicle. *Int. J. Comput. Vision*, 73(1):41–59, 2007.

[GNW96] Christian Goerick, Detlev Noll, and Martin Werner. Artificial neural networks in real-time car detection and tracking applications. *Pattern Recognition Letters*, 17(4):335 – 343, 1996. Neural Networks for Computer Vision Applications.

[HAD+94] M. Hansen, P. Anandan, K. Dana, G. van der Wal, and P. Burt. Real-time scene stabilization and mosaic construction. In *Applications of Computer Vision, 1994., Proceedings of the Second IEEE Workshop on*, pages 54 –62, 5-7 1994.

[Han97] John A. Hancock. High-speed obstacle detection for automated highway applications. Technical report, 1997.

[HR95] B. Heisele and W. Ritter. Obstacle detection based on color blob flow. In *Intelligent Vehicles '95 Symposium., Proceedings of the*, pages 282 –286, 25-26 1995.

[KHN91] D. Koller, N. Heinze, and H.H. Nagel. Algorithmic characterization of vehicle trajectories from image sequences by motion verbs. In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR '91., IEEE Computer Society Conference on*, pages 90 –95, 3-6 1991.

[LF04] Xia Liu and K. Fujimura. Pedestrian detection using stereo night vision. *Vehicular Technology, IEEE Transactions on*, 53(6):1657 – 1665, nov. 2004.

[LWD+07] Wei Liu, XueZhi Wen, Bobo Duan, Huai Yuan, and Nan Wang. Rear vehicle detection and tracking for lane change assist. In *Intelligent Vehicles Symposium, 2007 IEEE*, pages 252 –257, 13-15 2007.

[LWLW06] Zhenjiang Li, Kunfeng Wang, Li Li, and Fei-Yue Wang. A review on vision-based pedestrian detection for intelligent vehicles. In *Vehicular Electronics and Safety, 2006. ICVES 2006. IEEE International Conference on*, pages 57 –62, 13-15 2006.

[ND02] H. Nanda and L. Davis. Probabilistic template based pedestrian detection in infrared videos. In *Intelligent Vehicle Symposium, 2002. IEEE*, volume 1, pages 15 – 20 vol.1, 17-21 2002.

[OTFO03] R. Okada, Y. Taniguchi, K. Furukawa, and K. Onoguchi. Obstacle detection using projective invariant and vanishing lines. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 330 –337 vol.1, 13-16 2003.

[Per05] Marco Perez. Vision-Based Pedestrian Detection for Driving Assistance. 2005.

[SBM02] Zehang Sun, G. Bebis, and R. Miller. On-road vehicle detection using gabor filters and support vector machines. In *Digital Signal Processing, 2002. DSP 2002. 2002 14th International Conference on*, volume 2, pages 1019 – 1022 vol.2, 2002.

[SBM04] Zehang Sun, George Bebis, and Ronald Miller. On-road vehicle detection using optical sensors: A review. In *In IEEE International Conference on Intelligent Transportation Systems*, pages 585–590, 2004.

[SBM06] Zehang Sun, G. Bebis, and R. Miller. Monocular precrash vehicle detection: features and classifiers. *Image Processing, IEEE Transactions on*, 15(7):2019 –2034, july 2006.

[SK00] H. Schneiderman and T. Kanade. A statistical method for 3d object detection applied to faces and cars. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, volume 1, pages 746 –751 vol.1, 2000.

[Sri02] N. Srinivasa. Vision-based vehicle detection and tracking method for forward collision warning in automobiles. In *Intelligent Vehicle Symposium, 2002. IEEE*, volume 2, pages 626 – 631 vol.2, 17-21 2002.

[TSKK94] Ping-Sing Tsai, Mubarak Shah, Katharine Keiter, and Takis Kasparis. Cyclic motion detection for motion based recognition. *Pattern Recognition*, 27(12):1591 – 1603, 1994.

[WAPF98] Christian Wöhler, Joachim K. Anlauf, Till Pörtner, and Uwe Franke. A time delay neural network algorithm for real-time pedestrian recognition. In *INTERNATIONAL CONFERENCE ON INTELLIGENT VEHICLE*, pages 247–251, 1998.

[WL07] L. Walchshausl and R. Lindl. Multi-sensor classification using a boosted cascade detector. In *Intelligent Vehicles Symposium, 2007 IEEE*, pages 1045 –1049, 13-15 2007.

[WZ01] Junwen Wu and Xuegong Zhang. A pca classifier and its application in vehicle detection. In *Neural Networks, 2001. Proceedings. IJCNN '01. International Joint Conference on*, volume 1, pages 600 –604 vol.1, 2001.

[XF02] Fengliang Xu and Kikuo Fujimura. Pedestrian detection and tracking with night vision. In *Intelligent Vehicle Symposium, 2002. IEEE*, volume 1, pages 21 – 30 vol.1, 17-21 2002.

[YM94] S. Yasutomi and H. Mori. A method for discriminating of pedestrian based on rhythm. In *Intelligent Robots and Systems '94. 'Advanced Robotic Systems and the Real World', IROS '94. Proceedings of the IEEE/RSJ/GI International Conference on*, volume 2, pages 988 –995 vol.2, 12-16 1994.

[YYWZ] Ming Yang, Qian Yu, Hong Wang, and Bo Zhang. Vision-based Real-time Obstacles Detection and Tracking for Autonomous Vehicle Guidance.

[ZT00] L. Zhao and C.E. Thorpe. Stereo- and neural network-based pedestrian detection. *Intelligent Transportation Systems, IEEE Transactions on*, 01(3):148 –154, sep 2000.

[ZTH07] Thi Thi Zin, H. Takahashi, and H. Hama. Robust person detection using far infrared camera for image fusion. In *Innovative Computing, Information and Control, 2007. ICICIC '07. Second International Conference on*, pages 310 –310, 5-7 2007.

[ZY93] Guo-Wei Zhao and S. Yuta. Obstacle detection by vision system for an autonomous vehicle. In *Intelligent Vehicles '93 Symposium*, pages 31 –36, 14-16 1993.