2004 IEEE Intelligent Vehicles Symposium
University of Parma
Parma, Italy • June 14-17, 2004

High Accuracy Stereo Vision System for Far Distance Obstacle Detection

Prof. Sergiu Nedevschi,
Radu Danescu, Dan Frentiu, Tiberiu Marita,
Florin Oniga, Ciprian Pocol

Technical University of Cluj-Napoca
Department of Computer Science
sergiu.nedevschi@cs.utcluj.ro

Dr. Rolf Schmidt
Dr. Thorsten Graf

Volkswagen AG
Group Research, Electronics
rolf4.schmidt@volkswagen.de
thorsten.graf@volkswagen.de

Abstract

This paper presents a high accuracy stereo vision system for obstacle detection and vehicle environment perception in a large variety of traffic scenarios, from highway to urban. The system detects obstacles of all types, even at high distance, outputting them as a list of cuboids having a position in 3D coordinates, size and speed.

1. Motivation

Driven by the desire to control a continuously growing traffic density and a higher complexity in traffic control modern information society is in search of new solutions. European's mobility doubles in the past 30 years from 17 to 35 km per day, fleet increases even up to a factor of four [14]. Common goal remains the increase of traffic safety for all road users, even if the number of injuries decreases by 50% annually (Germany, 1980 until 2000) [15]. Passive safety measures played a main role in the past. In the future active systems so-called advanced driver assistance systems (ADAS) will become more and more important as a major part in electronic innovations for vehicles [16].

2. Advanced Driver Assistance Systems

ADAS will not only boost driving comfort and safety but also traffic flow. Today there are already active systems available for many cars, i.e. ABS (anti-lock brake system), ESP (electronic stability program) or BA (brake assistant). ACC (autonomous cruise control) increases driving comfort and will become available for high-volume cars in the near future.

Looking into major national [12] and international [13] research activities, points up current three-stage trends in R&D:

- Comfort functions to simplify driving tasks in monotonous situations, i.e. ACC
- Warning functions to warn the driver in critical situations, i.e. lane departure warning
- Safety function to reduce or avoid crashworthiness, i.e. emergency braking.

The development focuses more and more on the interaction between vehicles and their driving environment. First of all this includes a detection and interpretation of the driving environment by means of several sensor systems. The necessary abstraction layer of the environment is determined by the ADAS application itself: a simple longitudinal control task for ACC needs only distance and speed measurements of the target driving ahead, whereas warning and safety functions in much more complex driving situations have a need for dimensions of potentially dangerous obstacles. Comprehensive and reliable detection of the driving environment in complex situations like traffic jams on highways or inner city areas require much better sensor information like from a stereo vision sensor.

Obstacle detection through image processing has followed two main trends: single-camera based detection and two (or more) camera based detection (stereovision based detection). The monocular approach uses techniques such as object model fitting [1], color segmentation [2], or detection of specific characteristics such as texture [3] or symmetry axes [4,5,6]. The measurement of 3D characteristics is done after the detection stage, and it is usually performed through a combination of knowledge about the objects (such as size), assumptions about the characteristics of the road and knowledge about the camera parameters through calibration. The stereovision-based approaches have the advantage of directly estimating the 3D coordinates of an image feature, this feature being anything from a point to a complex structure. Stereovision involves finding correspondences from the left to the right image. The search for correspondences is a difficult, time demanding task, which is not free from the possibility of errors. Obstacle detection techniques involving stereovision use different approaches in order to make some simplifications of the classic problem and achieve real-time capabilities. For instance, [7] uses stereovision only to measure the distance of an object after it has been detected

0-7803-8310-9/04/$20.00 © 2004 IEEE

from monocular images, [8] detects the obstacle points from their stereo disparity compared to the expected disparity of a road point, [9] detects obstacle features by performing two correlation processes, one under the assumption that the feature is part of a vertical surface and another under the assumption that it is part of a horizontal surface, and comparing the quality of the matching in each of the cases. A stereovision system that uses no correspondence search at all, but warps images instead and then performs subtraction is presented in [10].

Our approach performs a full 3D reconstruction of the visible scene, the only limitation being that the reconstructed points must lie on vertical or oblique edges. The list of obtained 3D points is grouped into objects based solely on density and vicinity criteria. In this way, the system detects obstacles of all types, outputting them as a list of cuboids having 3D positions and sizes, without having to make any assumption about their type. Subsequent classification techniques can be employed for discrimination, if needed. The detected objects are then tracked using a multiple object tracking algorithm, which refines the grouping and positioning, and detects the speed.

3. Environment Model

All 3D entities (points, objects) are expressed in the world coordinate system, which is depicted in figure 1. This coordinate system, which is actually a car coordinate system, has its origin on the ground in the front of the car, and its Z axis points in our direction of travel. This last property is carefully ensured at camera calibration time.

Figure 2 shows the position of the left and the right cameras in the world coordinate system. This position is completely determined by the translation vectors $\mathbf{T}_L$ and $\mathbf{T}_R$, and the rotation matrices $\mathbf{R}_L$ and $\mathbf{R}_R$ between the cameras and the world coordinate systems. These parameters are essential for the stereo reconstruction process and for the epipolar line computation procedure.

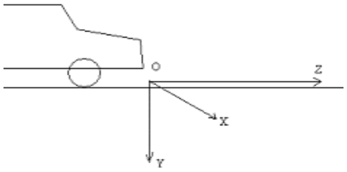


Figure 1: The world coordinate system

In order to estimate the translation vectors and the rotation matrices, camera calibration is performed after the cameras are mounted and fixed on the car. A general-purpose calibration technique is used.

The objects are represented as cuboids, having position, size and velocity. The position (X, Y, Z) and velocity ($\mathbf{v}_X$ and $\mathbf{v}_Z$) are expressed for the central lower point C of the object.

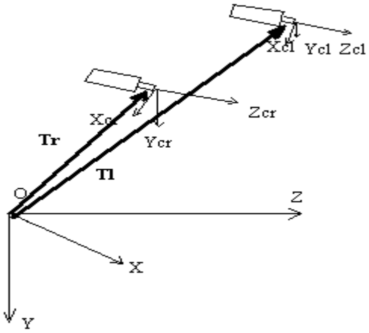


Figure 2: The cameras coordinate systems

4. Extracting 3D Information by Stereovision

The stereo reconstruction algorithm used is mainly based on the classical stereovision principles available in the existing literature [11]: find pairs of left-right correspondent points and map them into the 3D world using the stereo system geometry determined by calibration.

Constraints, concerning real-time response of the system and high confidence of the reconstructed points, must be used. In order to reduce the search space and to emphasize the structure of the objects, only edge points of the left image are correlated to the right image points. Due to the cameras horizontal disparity, a gradient-based vertical edge detector was implemented. Non-maxima suppression and hysteresis edge linking are being used. By focusing to the image edges, not only the response time is improved, but also the correlation task is easier, since these points are placed in non-uniform image areas.

Area based correlation is used. For each left edge point, the right image correspondent is searched. The sum of absolute differences (SAD) function [9] is used as a measure of similarity, applied on a local neighborhood (5x5 or 7x7 pixels). Parallel processing features of the processor are used to implement this function. The search is performed along the epipolar line computed from the stereo geometry. Two modes are used: image rectification, search along the horizontal line or without rectification and the search is performed along the epipolar line determined by the system geometry.

To have a low rate of false pairs, only strong responses of the correlation function are considered as correspondents. If the global minimum of the function is not strong enough relative to other local minimums than the current left image point is not correlated. In figure 3 a successful correlation is shown along the first column, while the last two columns show ambiguous similarity functions with rejected correspondents. Repetitive patterns are rejected and only robust pairs are reconstructed.

To achieve a better 3D depth resolution, the sub-pixel right correspondent is computed by fitting a parabola to the correlation function [9].

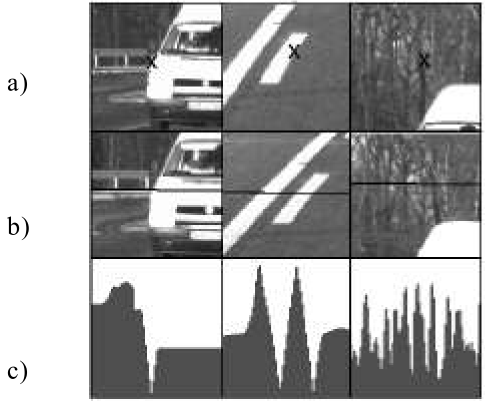


Figure 3: Three correlation scenarios are shown on each column. Left image point marked by 'x' on row a), right image search area and the epipolar line on row b) and the correlation function (lower means better match) on row c)

The parabola is fitted to a local neighborhood (3 or 5 points) of the global minimum. The obtained accuracy is about 1/4 to 1/6 pixels. This accuracy is dependent of the image quality (especially noise level and contrast). Our tests proved that the 3-neighbors parabola works better than the other one.

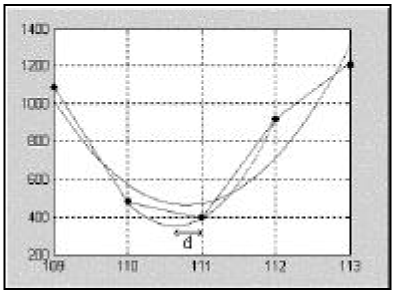


Figure 4: Linear piecewise approximation of the correlation function for 5 points. Two parabolas fitted to 3 and 5 neighbors are presented. The sub-pixel displacement '*d'* for the 3-neighbors parabola is shown.

After this step of finding correspondences, each left-right pair of points is mapped into a unique 3D point [11]. Two 3D projection rays are traced, using the camera geometry, one for each point of the pair. By computing the intersection of the two projection rays, the coordinates of the 3D point are determined. The reconstruction formulas are simple, when image rectification is used, or complex, if the original images are used for correlation.

While image rectification provides a simple search area for correspondents and straightforward 3D reconstruction, the general geometry mode, without rectification, provides a better resolution since no image re-sampling is done.

5. Grouping 3D Points into Objects

We use only 3D points situated at the level of traffic objects. Objects implied in traffic are just above the road. Points at the road level and too high points are rejected. Also points that are too lateral or too far are rejected. The remaining points belong to the so-called Space of Interest (SOI). Our SOI is a parallelepiped, which is parallel with the road and just above it. The road is assumed to be planar.

The extrinsic parameters of the cameras are calibrated before the test drive. The cameras are fix with respect to the car. Thus, the cameras move together with the car. The angles between the car and the road surface will change due to static and dynamic factors. The loading of the car is a static factor. Acceleration, deceleration and steering are dynamic factors, which also cause the car to change its pitch and roll angles with respect to the road surface. To obtain these two angles we measure the distance between the car's chassis and wheels because the wheels are on the road surface. Four sensors are mounted between the chassis and wheels arms. The pitch and roll angles being computed, the SOI can be placed just above the road. The height of SOI is chosen to contain just the tallest vehicle.

In our SOI, no object is placed above other. Thus, on a satellite view of the 3D points in SOI, we are able to distinguish regions with high points density, representing and locating objects. Regions with low density are assumed to contain noisy points and are neglected. The satellite view of 3D points is analyzed to identify objects. In figure 5 such a view is shown.

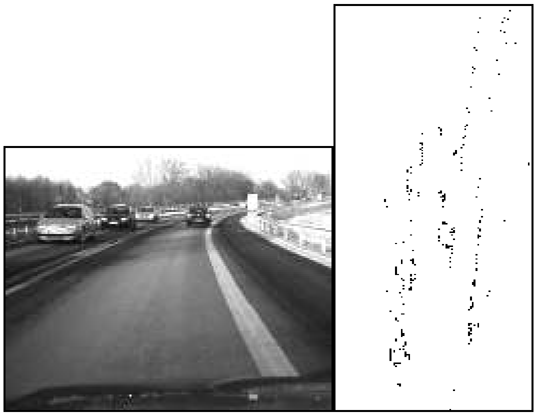


Figure 5: Left image and the satellite view of the 3D points

An important observation is that the 3D points are more and more rare as the distance grows. To overcome this phenomenon, we compress the satellite view of the space, depending on distance, in such a way that local density of points, in the new space, is kept constant. Regardless the distance to an object, in the compressed space, the region where that object is located will have the same points density.

The compression factor depends on distance (Z):

$$Scale(Z) = f \cdot \frac{1}{Z} \cdot k$$

Where: Z – distance

F – focal length of the cameras

k – is a manually chosen factor, depending on the richness of 3D reconstructed points with the current reconstruction method. For X and Z axis the values for k can be different.

The k factors are chosen to satisfy two conditions of the found objects:

- to not divide a real object into many smaller objects;
- to not unify many real objects into one bigger object.

The equations used to find the position (row, col) in the compressed space, of a point (X, Z) in the uncompressed space, are:

$$row = \log_{1+\frac{k}{f}} \frac{z}{z_{\min}}$$; $Z_{\min}$ = low distance limit of SOI

$$col = X \cdot Scale(Z)$$

The compressed space of the scene depicted in figure 6 is presented in the figure 8.:

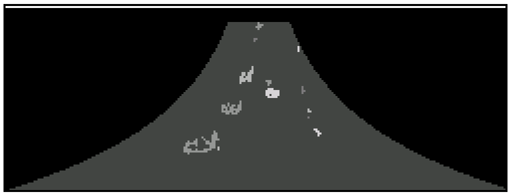


Figure 6. The compressed space and the identified objects

Also, in figure 6, objects were identified as dense regions.

For the resulted objects their limits along the Y-axis are found. In figure 7 the cuboids circumscribing objects are shown.

6. Object Tracking

Object tracking is used in order to obtain more stable results, and also to estimate the velocity of an object along the axes X and Z. The Y coordinate is tracked separately, using a simplified approach of simply averaging the current coordinate by the last detected coordinate.

The mathematical support of object tracking is the linear Kalman filter. The position of the object is considered to be in a uniform motion, with constant velocity. The position and speed parameters of the object along the axes X and Z at the moment k are components of the state vector $\mathbf{X}(k)$ that we try to evaluate through the tracking process.

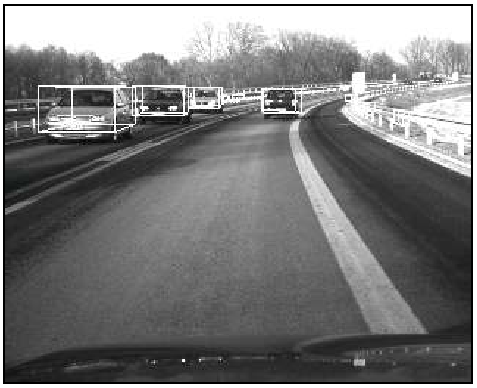


Figure 7: Perspective view of object cuboids painted over the image

The actual detection of the object will form the measurement vector $\mathbf{Y}(k)$, which consists only of the coordinates of the detected object.

$$\mathbf{X}(k) = \begin{bmatrix} x(k) \\ z(k) \\ v_x(k) \\ v_z(k) \end{bmatrix}$$

The evolution of the **X** vector is expressed by the linear equation:

$$\mathbf{X}(k) = \mathbf{A}(k) \times \mathbf{X}(k-1) \qquad (1)$$

where the state transition matrix $\mathbf{A}(k)$ is

$$\mathbf{A}(k) = \begin{bmatrix} 1 & 0 & \Delta t(k) & 0 \\ 0 & 1 & 0 & \Delta t(k) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The steps of tracking a single object are:

- **Prediction:** a new position of the object is computed using the last state vector and the transition matrix, through equation (1). The prediction of the Y coordinate is the last Y coordinate.
- **Measurement:** around the predicted position (*pX ,pY, pZ*) we search for objects resulted from grouping which have the distance to the prediction below a threshold. The distance is computed by equation (2), which gives different weights to displacements along the three coordinate axes, and take into consideration also the current object speed, which is seen as an indetermination factor.

$$D(x,y,z) = 2\left|\,|x - pX| - \left|\frac{v_x}{3}\right|\,\right| + 0.7|y - pY| + \qquad (2)$$
$$+ 0.5\left|\,|z - pZ| - \left|\frac{v_z}{3}\right|\,\right|$$

The objects that satisfy the vicinity condition are used to form an envelope whose position is computed and used as measurement, and the size of the envelope is used as the current measurement of the tracked object's size. By creating an envelope object out of the objects near a track we can join objects that were previously detected as separate. This merging becomes effective only if the separated objects have the same trajectory. This is ensured by the object-track association, when track compete for objects, and a false object joining won't last for too long.

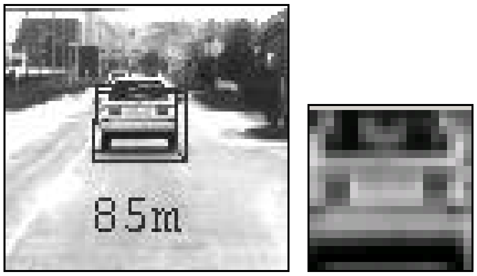
- **Update:** The measurement and the prediction are used to update the state vector **X** through the equations of the Kalman filter. The Y coordinate and the object's size are tracked by averaging the current measurement with the past measurements. If in the current frame there is no measurement that can be associated to the track, the prediction is used as output of the tracking system. The track is considered lost after a number of frames without measurement.

Tracking multiple objects adds a little bit more complexity to the algorithm presented above. We have to decide which detected object belongs to which track, or if a detected object starts a new track.

The association between detected objects and tracks is done using a modified nearest-neighbor method, using the distance expressed by equation (2). Each object is compared against each track. The objects are labeled employing the nearest track identity number, provided that there is at least one track that has a sufficient low distance to the object. The modification from the classical nearest-neighbor scheme is that we introduce an "age discount" in the distance comparison, and in this way we give priority to the older, more established tracks. This discounting mechanism is achieved by sorting the tracks in the reverse order of their age (the older ones first). If we compare an object to a track and the object already was labeled with the label of another track, we change the owner of the object only if the distance object-current track is lower than the distance to the older track minus a fixed quantity, the age discount.

For every object that cannot be assigned to an existing track and that fulfills some specific conditions, a new track is initialized. A new track is started for a single object, which has a reasonable size. There is no object joining in the initialization phase of a track. In this way we avoid initializing tracks to noise objects, and thus amplifying the noise. Tracks are aborted if the association process fails for a predefined number of frames

A tracking validation process based on the image of the object is employed in order to ensure that there is no track switching from one object to another. If an object is tracked for several frames and therefore its size is well established an image of the object is taken using its projection in the left image (figure 8, a), normalized to a 20x20 pixel size and stored (figure 8, b). In subsequent frames this image is matched against a 40x40 pixel normalized search area around the tracked object. If the matching fails for more frames, the track is aborted.



a) b)

Figure 8: A tracked object a) and its normalized image b)

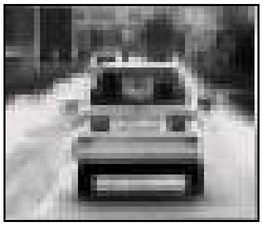


Figure 9: The search area used for validation

7. Results

The detection system has been deployed on a standard 1 GHz Pentium® III personal computer, and the whole processing cycle takes less than 100 ms processing time, therefore securing a 10 fps detection rate. This makes the system suitable for real-time applications. The system has been tested in various traffic scenarios, both offline (using stored sequences) and online (on-board processing), and acted well in both conditions. Tests covered as much traffic conditions as possible: urban, as in figure 10, highway, as in figure 11, or country road (figure 12). In all situations the obstacles were reliably detected and tracked, and their position, size and velocity measured. The detection has proven to have a maximum working range of about 90 m, with maximum of reliability in the range 10-60 m. The position measurement error is, naturally, higher than one can obtain from a radar system, but it is very low for a vision system: less than 10 cm of error at 10 m, about 30 cm of error at 45 m and about 2 m of error at 95 m.

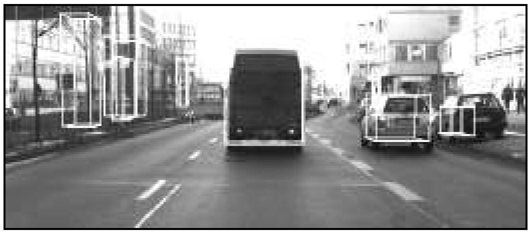


Figure 10: Detection results in urban traffic

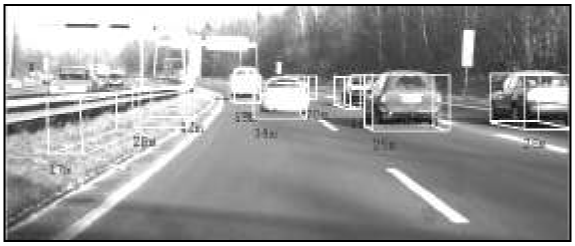


Figure 11: Detection results in highway traffic

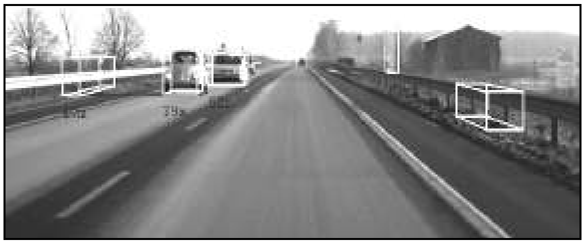


Figure 12: Tracking of incoming traffic on country road

The edge performance of tracking is best shown in figure 14, where we have incoming traffic, and relative speeds of more than 200 km/h. Nevertheless, the incoming objects are correctly tracked and their speeds are estimated. In this situation our vehicle is moving at 100 km/h and the incoming vehicles are moving at 120 km/h. The highest errors in speed estimation happen for the objects outside the road (barriers, trees, bushes) which are very poorly delimited one from another and therefore it is very difficult to estimate the motion from one frame to another.

8. Conclusions

We have presented a stereovision-based obstacle detection system that reconstructs and works on 3D points corresponding to the object edges, in a large variety of traffic scenarios, and under real-time constraints. The system is suitable to vehicle environment perception and to be integrated in a driving assistance application. The functions of this system can be greatly extended in the future. An intelligent correlation function should be developed, one that can disambiguate, not reject, repetitive patterns and reconstruct points from horizontal edges. Because the stereovision module reconstructs any feature in sight, it means that it reconstructs also the road features, and therefore it can form the basis for a 3D lane detection algorithm. Moreover, because any type of object is detected this algorithm can form the basis for any type of specific object detection system, such as vehicle detection, pedestrian detection, or even traffic sign detection. The classification routines can be performed directly on our detected objects, with the advantage of reduced search space and additional helpful information such as distance, size and speed, which can also reduce the class hypotheses.

References

[1] D. M. Gavrila, "Pedestrian Detection from a Moving Vehicle", *Proc. of European Conference on Computer Vision,* pp. 37-49, Dublin, Ireland, 2000

[2] I. Ulrich and I. Nourbakhsh, "Appearance-Based Obstacle Detection with Monocular Color Vision", *Proc. of the AAAI National Conference on Artificial Intelligence, Austin, TX, July/August 2000*

[3]. T. Kalinke, C. Tzomakas, and W. von Seelen, "A Texture based Object Detection and an Adaptive Model-based Classification", in *Procs. IEEE Intelligent Vehicles Symposium'98*, (Stuttgart, Germany), pp. 341–346, Oct. 1998.

[4] G. Marola, "Using Symmetry for Detecting and Locating Objects in a Picture", *Computer Vision Graphics and Image Processing*, vol. 46, pp. 179–195, May 1989.

[5] T. Zielke, M. Brauckmann, andW. von Seelen, "Intensity and Edge-based Symmetry Detection with an Application to Car-Following", *CVGIP: Image Understanding*, vol. 58, pp. 177–190, 1993.

[6] A. Kuehnle, "Symmetry-based vehicle location for AHS", in *Procs. SPIE Transportation Sensors and Controls: Collision Avoidance, Traffic Management, and ITS,* vol. 2902, (Orlando, FL), pp. 19–27, Nov. 1998.

[7] M. Bertozzi and A. Broggi, "GOLD: a Parallel Real-Time Stereo Vision System for Generic Obstacle and Lane Detection", *IEEE Trans. on Image Processing*, 7(1):62-81, January 1998,

[8] U. Franke, D. M. Gavrila, S. Görzig, F. Lindner, F. Paetzhold and C. Wöhler, "Autonomous Driving Approaches Downtown", *IEEE Intelligent Systems*, vol.13, nr.6, pp. 40-48, 1998.

[9] T. A. Williamson, "A high-performance stereo vision system for obstacle detection", September 25, 1998, *CMU-RI-TR-98-24.* Robotics Institute Carnegie Mellon University. Pittsburg, PA 15123.

[10] M. Bertozzi, A. Broggi, A. Fascioli, and S. Nichele, "Stereo Vision-based Vehicle Detection", in *Procs. IEEE Intelligent Vehicles Symposium 2000*, pages 39-44, Detroit, USA, October 2000.

[11] E. Trucco and A. Verri, Introductory Techniques for 3-D Computer Vision, *Prentice-Hall*, 1998

[12] Founded German research initiative INVENT – Intelligenter Verkehr und nutzergerechte Technik, http://www.invent-online.de

[13] Founded projects of the European Commision, Frame program 5, *IST Information Society Technologies,* http://www.cordis.lu/ist/projects/projects.htm

[14] KOMMISSION DER EUROPÄISCHEN GEMEINSCHAFTEN: "WEISSBUCH, Die europäische Verkehrspolitik bis 2010: Weichenstellungen für die Zukunft", *KOM(2001)_370,* Brüssel, September 2001

[15] M. Stanzel: „Unfallforschung zum Zusammenwirken aktiver und passiver Sicherheit im Pkw", *Fahrerassistenzsysteme und Aktive Sicherheit, Haus der Technik, Es-sen*, 20.11.2002

[16] W. Specks, R. Schmidt ,P. Schulenberg: „Elektronikkonzepte für zukünftige Fahrerassistenzsysteme", *VDA-Technischer Kongress*, 26.-27. März 2001 Bad Homburg, pp. 127-138