STEPS TOWARDS AN INTELLIGENT VISION SYSTEM FOR DRIVER ASSISTANCE IN URBAN TRAFFIC

U.Franke, S.Görzig, F.Lindner*, D.Mehren*, F.Paetzold
Daimler-Benz Research, T728
D-70546 Stuttgart, Germany
{franke, goerzig, paetzold}@dbag.stg.DaimlerBenz.com
{lindner, mehren}@dbag.ulm.DaimlerBenz.com

Machine Vision, Image Understanding, Autonomous Driving

ABSTRACT

Most computer vision systems for vehicle guidance developed in the past were tailored to the relatively simple highway scenario.
Autonomous driving in the much more complex urban environment or driver assistance systems like intelligent Stop&Go are new challenges not only from the algorithmic but also from the system architecture point of view.
This contribution describes our current work on these topics. It includes appropriate vision algorithms as well as the control of different vision modules following the „active vision" paradigm.

1. INTRODUCTION

Most of the known autonomous demonstrators have been designed for highway traffic since this scenario is relatively simple: lanes are usually well marked and built with slowly changing curvature, traffic signs are large and well visible and vehicles must be considered as the only potential obstacles in front of the autonomous car. Consequently, special robust algorithms have been designed for these tasks.

Vision becomes much more difficult if one concerns driving on urban roads. The most important objects one has to recognize in this environment are:

- The course of the lane even if it is not given by well painted markings and does not show clothoidal geometry
- Small traffic signs and traffic lights in a highly colored environment
- Different additional traffic participants like bicyclists or pedestrians
- Still obstacles that limit the available free space e.g. parking cars.

Although the urban scenario is highly complex, it is very attractive for driver assistance. Imagine an intelligent Stop&Go system that is able to behave like a human driver: it does not only keep the distance to its leader constant, as a radar based system would do, but also stops at red traffic lights and stop signs, gives right of way to other vehicles if necessary and tries to avoid collisions with children running over the street.

In the following we describe our approach to build such a complex system without the necessity of a super computer in the trunk and present the vision algorithms currently under development for urban driving.

2. PERCEPTION CONTROL

In our well known Prometheus demonstrator each vision module was assigned to a subset of processors in a static configuration. There were about 200 processes running permanently on a network

* D.Mehren and R.Lindner are with the Daimler-Benz Research Center Ulm, Germany

of 70 transputers, even though the results of the vision modules were of no interest for vehicle guidance at that moment.

To use this architecture for computer vision in complex inner city traffic would afford even more resources. But why looking for new traffic signs or continuously determine the lane while the car has stopped at a red traffic light? While following a leading vehicle, why shouldn't only tasks be processed which are useful in that situation?

Such questions motivate the development of other strategies, which should be able to perform the perception tasks with a given number of computational nodes.

The pursued approach is to control image processing and system resources using a dynamic configuration system for computer vision. The central component is a permanently updated data base, which stores all data delivered by the available algorithms. This includes a vehicle fixed 3D environment map containing objects on a symbolic level.

Every object in the data base has a description of perception modules, which can be used to obtain more information about itself or to update old data. The object 'traffic light' for example can perform the methods 'track position' and 'get state'.

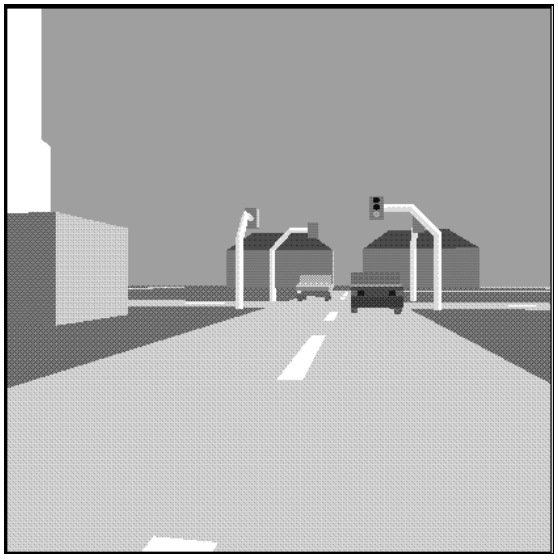

Fig. 2.1: Visualization of a virtual symbolic camera sensor in the simulator

The system analyses all objects and determines methods which are useful in the current situation. Subsequently, these methods are prioritized and the most important tasks are executed on free computational nodes of the used MIMD architecture. Therefore, the priority calculation has to include information like time-to-collision, the uncertainty of object data and the dependencies of objects among each other.

The results of the executed algorithms are finally added to the data base, closing the vision control loop.

The described approach has several advantages:

- The resources can be focused on relevant tasks for specific situations
- The symbolic level facilitates sensor fusion; different sensors can be used to perform a perception task
- The perception process can be described by higher-level instructions
- Modular basic image processing algorithms increase the robustness of the perception process

To test the configuration system and to develop and verify perception strategies, a traffic simulator was built. It generates typical traffic scenes, including cars, pedestrians, bikes, traffic lights and signs etc. (see fig. 2.1). A car with the steering behavior of an S-Class Mercedes interacts with the environment by virtual symbolic sensors and actuators.

These sensors allow to perform virtual perception tasks, which are triggered by the configuration system. A comparison between the objects in the dynamically updated local environment map of the configuration system and the objects in the simulator map shows directly the efficiency of the vision process and enables an evaluation of different strategies.

3. TRAFFIC SIGN RECOGNITION

Driving in urban traffic requires knowledge about the infrastructure's local information system (e.g. traffic signs, traffic lights and road markings) to comply with the traffic rules.

Our traffic sign recognition system has been originally designed for the highway scenario [Jan93, Est94, Rit95]. It can be divided into three steps:

1) a fast color segmentation based on a polynomial classifier,
2) a one pass scan on the color segmented image to find regions of interest, and
3) a hierarchical classification tree for the recognition task.

This approach is now being extended for the inner city scenario. This includes several new tasks such as:

- handling image distortion
- extending the field of view to an angle of 120° using 2 cameras
- using reduced resolution in the classifier
- adapting the color segmentation
- adding new types of traffic signs.

In inner cities one gets more time to recognize a single sign, because of the limited speed. Therefore, the extension of the viewing angle represents mostly a sensorial problem. Using wide angle objectives causes image distortions (fig. 3.1) and less resolution per traffic sign.

Fig. 3.1: Typical urban scene (color image)

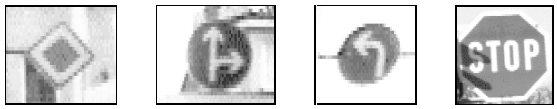

Fig. 3.2: Sample urban traffic signs

The additional traffic signs imply the inclusion of new colors in the segmentation task. This has been achieved by retraining the polynomial classifier. Some examples of the recently adapted inner city traffic signs are shown in fig. 3.2. In fig. 3.1, the detected signs are marked by rectangles.

For the recognition of road markings, in particular the direction arrows as shown in fig. 3.3, the successful approach has been modified. The color segmentation is replaced by an adaptive gray value segmentation. The following detection stage as well as the classifier have been reconfigured.

Fig. 3.3: Typical direction arrow

The size of the searched objects decreases due to the perspective view. Therefore the segmentation area is limited to the lower two thirds of the picture. For the regions in the segmented picture certain features like size, coverage or smoothness of contour are computed as part of the detection stage. These features are used to select regions of interest (fig. 3.4).

The classifier consists of an adapted Radial Basis module. If, like in the figure, it decides that straight arrow and straight-left arrow are both valid, the less restrictive decision is taken. The recognized arrow is shown in fig. 3.4 in white.

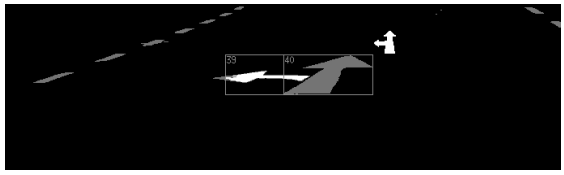

Fig. 3.4: Recognition of direction arrow

4. ROAD BOUNDARY ANALYSIS

Most conventional lane detection schemes are based on the principle introduced by Dickmanns [Dic86]. Using a spatiotemporal model, the relevant parameters are estimated from local measurements by means of Kalman filter techniques.

The analysis of the road topology for urban traffic environment has to cope with:

- a much wider variety of possible road geometries (intersections, junctions)
- roads that are not bounded by markings and
- frequent occlusions.

Therefore, a deeper understanding of the road situation is required for lateral vehicle control in urban traffic. Besides the region based approach, described in the previous chapter, an edge based scheme is under investigation.

Fig.4.1: Typical road image

First, the edges are extracted using a recursive Shen-filter and approximated by polygons. Fig. 4.2 shows the result obtained from fig. 4.1. For a fast selective access to the contours, a database is established that organizes the contours with respect to their length, orientation and position. Regions where obstacles, detected by the stereo vision system described in chapter 5, occlude or delimit the road surface are ruled out as possible positions of lane boundaries.

The subsequent labeling process classifies the contours as 'possible lane boundaries' or 'clutter'. This is done by assessing the similarity of the contour features with the expected lane properties. The following attributes are employed to find possible lane boundaries: inverse perspective mapped coordinates, orientation, length, parallelism and collinearity. All contours with a similarity measure above a certain threshold are accepted as possible lane boundaries. The dark lines in fig. 4.2 present the result of this process.

Fig. 4.2: Extracted road-boundaries

Even though the maximum speed of the vehicle is low, its lateral position must be estimated at a rate of at least 10Hz to accomplish comfortable lateral guidance. The analysis of the entire road topology as described above is computationally too expensive. In order to overcome this problem, a fast boundary tracking scheme is initialized with the results of the analysis and delivers position and yaw angle of the vehicle at a rate of 25 Hz.

In this two-layer architecture, the tracker runs under control of the global analysis. It is repeatedly supervised and redirected to other boundaries if required by the current situation. Thus the benefits of a robust tracking approach are exploited while its deficiencies in the considered scenario are circumvented. In addition, the permanently running tracker allows the vision control system to delay the global analysis if tasks with higher priority have to be processed.

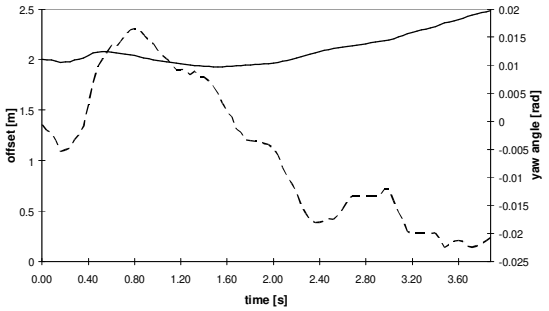

Fig. 4.3: Lateral distance to the right road boundary and yaw angle (dashed line)

The considered scene has been recorded during a manual drive. Fig. 4.3 shows the lateral distance of our vehicle relative to the kerb detected on the right side of the lane. In addition, the observed yaw angle is plotted in the diagram.

5. OBJECT DETECTION

Navigation in urban traffic requires 3D information for object detection and motion state estimation. The model based approaches used in many highway applications to detect objects i.e. leading cars are not appropriate since the system has to deal with a large number of different objects, some of them non-rigid, some of them unknown.

The most powerful method to derive 3D-information is stereo vision. Unfortunately, classical approaches are computationally very expensive. Since it is our philosophy to avoid special purpose hardware, we developed a feature based approach that is tailored to our specific needs and runs in real time on a simple PowerPC 604 [Fra96].

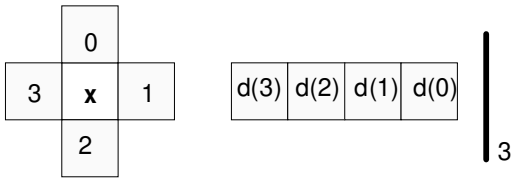

Fig 5.1: Encoding scheme of the structure classification based on a 4 pixel neighborhood

For the correspondance analysis each pixel is classified according to the grey values of its 4 direct neighbors. If these surrounding pixels are counted clockwise, a 4 digit word can be formed, as shown in fig.5.1. Each digit is assigned a difference value d(i) depending on its intensity g(i) compared to the brightness g(x) of the central pixel x. According to

$$d(i) = \begin{cases} 1 & \text{if } g(i) - g(x) > +T \\ 2 & \text{if } g(i) - g(x) < -T, \\ 0 & \text{else} \end{cases}$$

d(i) can get three different values meaning significant brighter, much darker or similar intensity. T is a difference threshold defining similarity.

This ternary word can have 81 different values. The decimal class index C of the considered pixel **x** is then given by:

$$C = \sum_{i=0}^{3} 3^i \cdot d(i).$$

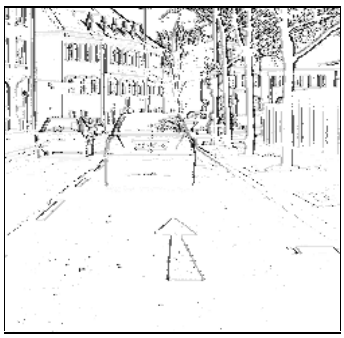

Fig. 5.2: Classified pixel

Fig.4.1 shows the left image of a stereo pair that is used in this chapter for illustration. The result of the structure classification using a difference threshold of T=20 ($g(i) \in [0,255]$) is shown in fig.5.2. For better visualization, the class image C(x,y) has been multiplied by a factor of 3 and inverted.

The camera pair is mounted with parallel camera axis, lens distortions are neglected at the moment. Correspondances are established between pixels with identical classes. Possible ambiguities are solved by favoring small disparities and exploitation of the ordering constraint. The outcome is a disparity image, which is the basis for all subsequent steps.

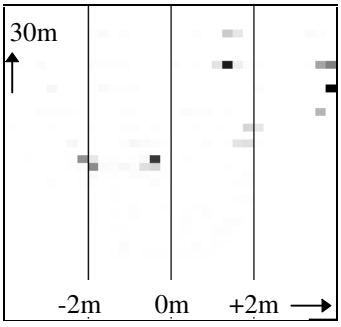

Fig. 5.3: Depth map, peaks are caused by the car and the trees

One way to illustrate the result is a 2D depth map. The map shown in fig. 5.3. covers an area of 30m in length and 8m in width. It is used to detect objects, that are tracked subsequently. From the position of the objects relative to the camera system their motion states are estimated by means of standard Kalman Filters and sent to the environment map mentioned in chapter 2.

In the current version, an unlimited number of objects can be considered. Sometimes the right and left part of a vehicle are initially tracked as two distinct objects. These objects are merged on

a higher „object-level“ if their relative position and motion fulfill reasonable conditions.

Fig. 4.1 already shown in chapter 4 is the first image of a sequence of 4 seconds duration. Fig. 5.4 shows the measured and filtered distances to the leading vehicle over the considered period of time as well as the relative speed. It can be seen that the speed is correctly estimated after 0.3 seconds.

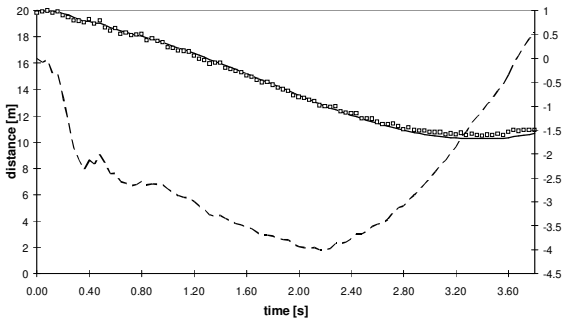

Fig. 5.4: filtered distance, distance measurements (squares) and relative speed (dashed line)

Structures on the road can easily be extracted by means of stereo vision if the road can be assumed to be flat. Fig. 5.5 shows pixel that have been classified as road pixel. This information can be used to support the recognition of the road boundaries and direction arrows described in chapter 4 and 3, respectively. In addition, camera height and tilt angle are estimated repeatedly using these points.

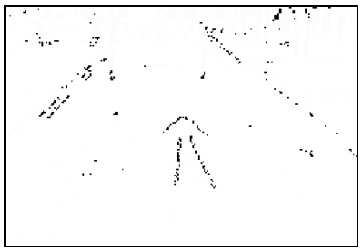

Fig. 5.5: Classified road pixel

7. SUMMARY

This contribution describes first steps towards an intelligent vision system for urban traffic. Objects are detected and tracked by means of stereo vision, road boundaries and direction arrows are recognized as well as traffic signs.

Currently, these modules are running separately on a parallel MIMD-machine which consists of 4 PowerPCs connected with serial high speed links, each link offering a sustained bandwidth of more than 20 Mbytes/sec. It is the heart of UTA (Urban Traffic Assistant), a Mercedes-Benz S-class vehicle used for the practical investigations.

The next step will be to integrate the vision modules in the presented perception control. It will allow to focus the attention on important objects in the scene and to optimize the performance of the system with respect to the restricted computational resources. Although designed for vision in city traffic, this control scheme is not restricted to vision tasks in this particular environment.

REFERENCES

[Dic86] E.D.Dickmanns, A.Zapp: "A curvature-based scheme for improving road vehicle guidance by computer vision", Proc. SPIE Conference on Mobile Robots, Vol. 727, 1986, S. 161-16

[Est94] S.Estable et al.: „A real-time traffic sign recognition system“, Intelligent Vehicles ‘94, Paris, 24.-26. Okt.. 1994, S.213-218

[Fra96] U.Franke, I.Kutzbach:“Fast Stereo based Object Detection for Stop&Go Traffic“, Intelligent Vehicles ‘96, Tokyo, pp.339-344

[Jan93] R.Janssen, W.Ritter, S.Ott, F.Stein.: „Hybrid Approach for Traffic Sign Recognition“, Intelligent Vehicle Symposium’93, Tokyo, pp. 390-397

[Rit95] W.Ritter, F.Stein. R.Janssen: „Traffic Sign Recognition Using Colour Information“, Math. Comput. Modelling, Vol. 22, No. 4-7, pp. 149-161

[Ulm94] B.Ulmer: „VITA II - Active collision avoidance in real traffic“, Intelligent Vehicles ‘94, Paris, 24.-26. Okt.. 1994, S.1-6