Proceedings of the
IEEE Intelligent Vehicles Symposium 2000
Dearborn (MI), USA • October 3-5, 2000

Real-time Stereo Vision for Urban Traffic Scene Understanding

U. Franke, A. Joos
DaimlerChrylser AG
D-70546 Stuttgart
HPC: T728
{uwe.franke, armin.joos}@DaimlerChrysler.com

Abstract

This paper presents a precise correlation-based stereo vision approach that allows real-time interpretation of traffic scenes and autonomous Stop&Go on a standard PC. The high speed is achieved by means of a multi-resolution analysis. It delivers the stereo disparities with sub-pixel accuracy and allows precise distance estimates. Traffic applications using this method are described.

1 Introduction

Within the UTA (Urban Traffic Assistance) project we have developed modules for understanding urban traffic scenes [1]. This includes the recognition of the infrastructure like traffic signs and lights, crosswalks, arrows and lanes as well as the detection of obstacles and the recognition of cars and pedestrians. Fig. 1 shows a typical inner city situation. The recognised objects are visualised on the screen: the leading vehicle, two pedestrians, the lane, the traffic light and the sign.

A key component of our demonstrator vehicle that has been firstly presented at the Intelligent Vehicles conference 1998 in Stuttgart is stereo vision. It allows the detection of arbitrary obstacles and the estimation of their relative motion at distances up to 50 meters.

One can distinguish between two different categories of stereo vision: area-based and feature-based approaches. Both of them have pros and cons:

- Feature-based systems are usually based on edges. They provide only sparse depth maps but can be implemented very efficiently. In indoor scenes, often vertical edges or corners are extracted in the stereo images and tracked over time [2]. A trinocular stereo vision system using vertical edges is described in [3].
- Area-based approaches, commonly based on correlation techniques, can generate dense depth maps but are computationally expensive and can have problems at occlusions. Several approaches have been implemented for vehicular application, such as a system by Subaru. They use a block matching algorithm based on 4x4 pixel blocks [4]. Another approach [5] uses band-pass filtered images in order to overcome the difficulties due to different image intensities of the left and right camera.

Fig. 1: View out of UTA II

We have developed two different stereo approaches, one feature based and one area-based. Both have in common that they do not require specialised hardware but are able to run in real-time on today's standard PC processors. They are sketched and

0-7803-6363-9/00/$10.00 © 2000 IEEE

compared in chapter two and three. Their applications to obstacle detection and tracking and analysis of free space in front of the car are given in chapter four.

2 Real-Time Stereo Analysis based on Local Features

When we started our work on stereo vision, correlation based stereo analysis was computationally too expensive for real-time analysis using standard processors. Since it was always our policy to avoid special hardware components, we developed a feature based approach. It uses a fast non-linear classification scheme to generate local features that are used for finding corresponding points. This scheme classifies each pixel according to the grey values of its four direct neighbours [6]. It is verified whether each neighbour is significant brighter, significant darker or has similar brightness compared to the considered central pixel. The similarity is controlled by thresholding the absolute difference of pixel pairs. This leads to $3^4=81$ different classes. In contrast to schemes that distinguish between positive and negative vertical edges only, this scheme is able to encode edges and corners at different orientations.

The correspondence analysis works on these feature images. The search for possibly corresponding pixels is reduced to a simple test whether two pixels belong to the same class. Since our cameras are mounted horizontally, only classes containing horizontal details are considered. Thanks to the epipolar constraint and the fact that the cameras are mounted with parallel optical axis, pixels with identical classes must be searched on corresponding image rows only.

It is obvious that this classification scheme cannot guarantee uniqueness of the correspondences. In case of ambiguities, the solution giving the smallest disparity i.e. the largest distance is chosen to overcome this problem. This prevents generation of phantom objects close to the camera caused by wrong correspondences e.g. in scenes with periodic structures. In addition, measurements that violate the ordering constraint are ignored.

Fig.2a shows the left image taken from our stereo camera system with a base width of 30 cm. The outcome of the correspondence analysis is a disparity image, which is the basis for all subsequent steps described in chapter 4. Fig. 2b visualises such an image. Of course, the result are noisy due to the extreme local operation. The advantage of this approach is its speed. On the currently used 400 MHz Pentium II processor this analysis is performed within 23 milliseconds on images of size 384x256 pixel.

Two facts might be a problem in some applications. First, the disparity image is computed with pixel accuracy only. This problem can simply be overcome by post-processing. Secondly, the described algorithm uses a threshold to measure similarity. Although the value of this threshold turns out to be uncritical, it is responsible for mismatches of structures of low contrast.

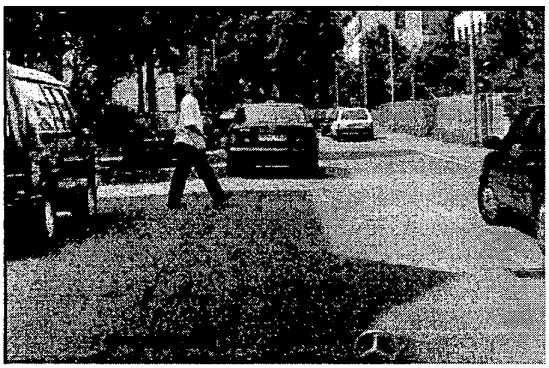

Fig.2a: Left image of the stereo camera system

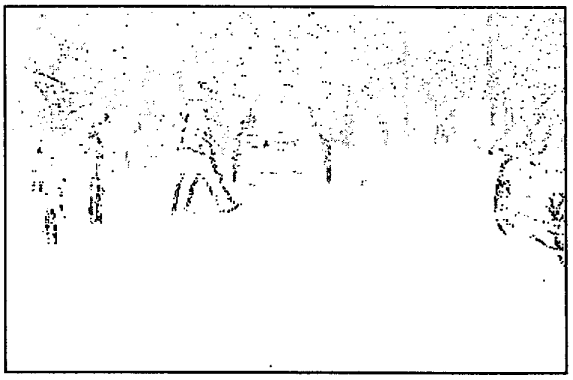

Fig 2b: Grey value encoded disparities obtained by the sketched feature based approach

3 Real-Time Stereo Analysis based on Correlation

The rapidly increasing computational power allows to realise area based techniques with real-time performance nowadays. For applications that require high precision 3D information and can not accept the noise level of the above scheme, we developed an alternative correlation-based approach. Nevertheless, the maximum processing time that we tolerate is 100 msec per image pair.

In order to reach this challenging goal, we use the sum-of-squared (SSD) or sum-of-absolute (SAD) differences criterion instead of expensive cross correlation to find the optimal fit. In order to avoid wrong results due to different mean and variance of the image pairs, we control gain and shutter of our cameras.

However, the demand for real-time performance is still a hard problem. Full brute-force correlation of 9x9 pixel windows requires about 9 seconds for images of size 384x256, if the maximum disparity is set to 80 pixel. With an optimised recursive implementation we achieved typical values of 1.2 seconds.

To speed up the computation, we use a multi-resolution approach in combination with an interest operator. The idea is to find correspondences on a coarse level that can be recursively refined. First, a gaussian pyramid is constructed for the left and right stereo images (see fig. 3). Areas with sufficient contrast are extracted by means of a fast horizontal edge extraction scheme, but any other interest operator that extracts points of significant horizontal variance can be used, too.

Pixel with sufficient gradient are marked, from which a binary pyramid is constructed, as shown in fig. 3. A pixel (i,j) at level n is marked if one of its 4 corresponding pixels at level n-1 is set. A non-maximum suppression can be applied to the gradient image in order to further speed up the processing. In this case, we find about 1100 attractive points at pyramid level zero (original image level), 700 at level one, 400 at level 2 and about 150 at level 3 on an average. Only those correlation windows with the central pixel marked in these interest images are considered during the disparity estimation procedure.

Depending on the application, the correlation

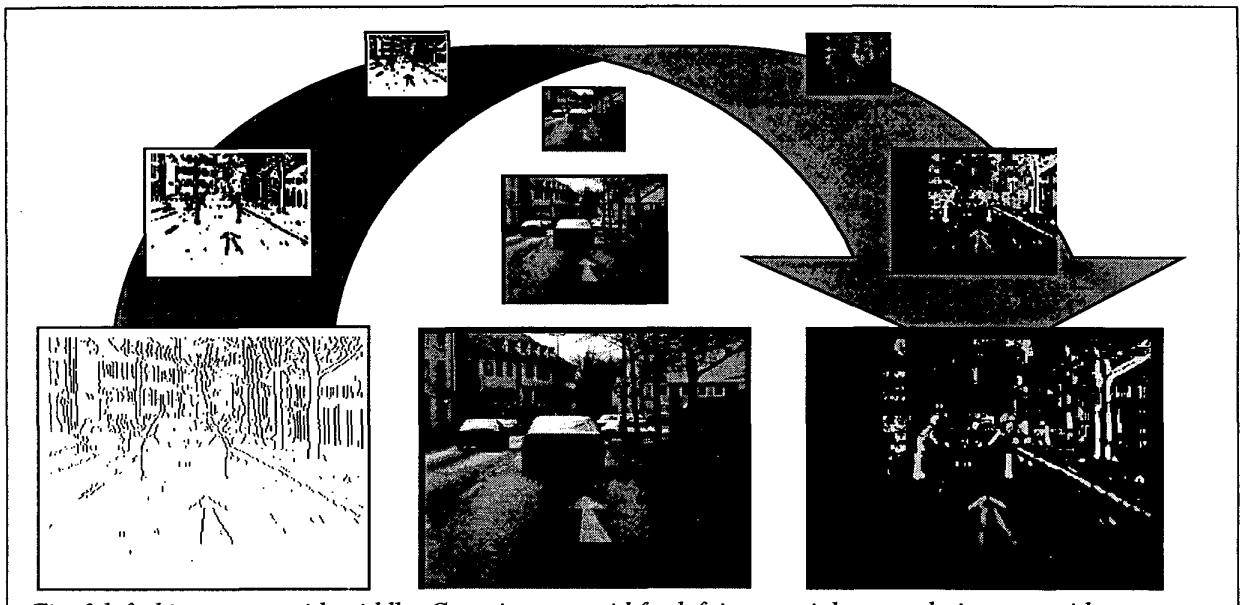

Fig. 3 left: binary pyramid, middle: Gaussian pyramid for left image, right: correlation pyramid

process starts at level two or three of the pyramid. If D is the maximum searched disparity at level zero, it reduces to D/2**n at level n. At level 2 this corresponds to a saving of computational burden of about 90% compared to a direct computation at level zero. Furthermore, smaller correlation windows can be used at higher levels which again accelerates the computation.

The result of this correlation is then transferred to the next lower level. Here, only a fine adjustment has to be performed within a small horizontal search area of +/- 1 pixel. This process is repeated until the final level is reached. At this level, subpixel accuracy is achieved by fitting a parabolic curve through the computed correlation coefficients.

The price we have to pay for this fast algorithm is that mismatches in the first computed level propagate down the pyramid and lead to serious errors. Since the quality of the found match cannot be judged by the measured SSD or SAD, we compute the normalised cross correlation coefficient for the best matches at the first (i.e. highest) correlation level and eliminate bad matches from further investigations. In addition, a left-right check can be applied to the disparity image obtained at the highest pyramid level. In case of ambiguities, the best match or the match with the smaller disparity can be selected. The latter strategy avoids the erroneous detection of close obstacles caused by periodic structures.

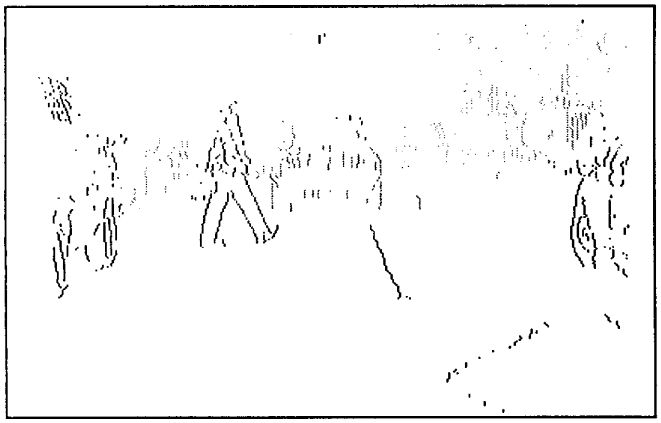

Fig. 4: Grey value encoded disparity image generated by the correlation approach. Non-maximum suppression has been applied to the interest image to speed up processing. Distance is inverse proportional to the darkness.

If we start at level 2 (resolution 91x64 pixel), the total analysis including pyramid construction runs at about 90 milliseconds on a 400 MHz Pentium. If we abandon the multi-resolution approach, about 450 milliseconds are necessary to yield comparable results. This corresponds to a speed up factor of 5. Starting at higher levels causes problems in our field of applications, since relevant structures can be lost.

A disparity image derived by this scheme is shown in fig. 4 with non-maximum suppression. Since a larger neighbourhood is taken into account during processing, the result looks less noisy than the feature based solution. In fact, only very few mismatches remain.

4 Understanding Traffic Scenes using the Depth Information

The obtained disparity or depth image delivers rich information for the subsequent processing steps in our UTA II vehicle. This includes obstacle detection and tracking, but also free space analysis and obstacle classification. Since the results of both sketched stereo algorithms are disparity images, the further processing is independent of the used approach.

4.1 Obstacle Detection

Driving on roads, we regard all objects above ground level as potential obstacles. If the cameras are mounted H meters above ground and looking downwards with a tilt angle α, all image points with a disparity d given by

$$d = x_l - x_r = \frac{B}{H} f_x \cdot [\frac{y}{f_y} \cdot \cos(\alpha) + \sin(\alpha)]$$

lie on the road. Here, B is the base line and f the focal length measured in pixel in horizontal (x) and vertical (y) direction. The y-axis is looking

downward, the center of the coordinate system is the optical axis.

The projection of all features above the road plane, i.e. those with disparities larger than given by the above equation, yields a two-dimensional depth map. In this histogram, obstacles show up as peaks.

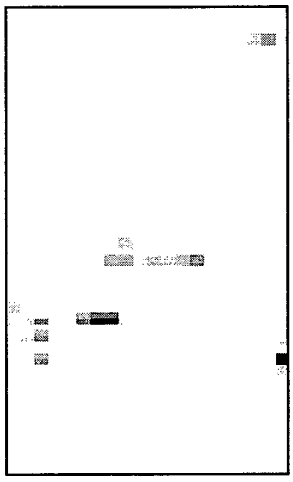

Fig.5: Depth map from bird eye's view (see text)

The map shown in fig.5 has been obtained for the situation shown in fig.2. It covers an area of 40m in length and 6m in width. The hits in the histogram are clearly caused by the cars parking left and right, the car in front and the pedestrian. The shown depth map has been obtained by the correlation method. This map is used to detect objects that are tracked subsequently. In each loop, already tracked objects are deleted in this depth map prior to the detection.

The detection step delivers a rough estimate of the object width (2.1 m for the car, 0.9m for the pedestrian). A rectangular box is fitted to the cluster of feature points that contributed to the extracted area in the depth map. This cluster is tracked from frame to frame in the depth image. For the estimation of the obstacle distance, the disparities of the object's feature points are averaged.

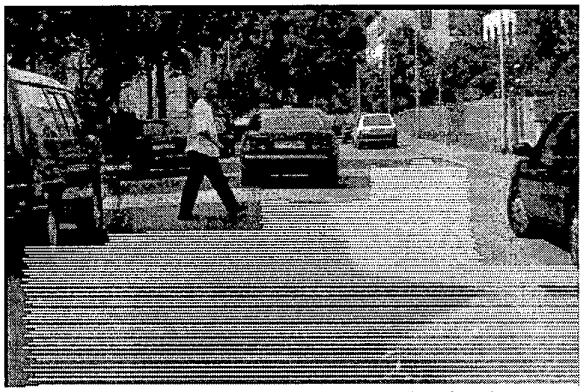

Fig.6: Free space determined from the depth information

From the position of the objects relative to the camera system their motion states i.e. speed and acceleration in longitudinal as well as lateral direction are estimated by means of Kalman filters. For the longitudinal state estimation we assume that the jerk, i.e. the deviation of the acceleration, of the tracked objects is small. This is expressed in the following state model with distance **d**, Speed **v** and acceleration **a**:

$$\begin{bmatrix} d \\ v_l \\ a_l \end{bmatrix}_{k+1} = \begin{bmatrix} 1 & T & T^2/2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} d \\ v_l \\ a_l \end{bmatrix}_k - T \cdot \begin{bmatrix} v_e \\ 0 \\ 0 \end{bmatrix},$$

The index **l** denotes the states of the lead vehicle, the index **e** denotes the ego vehicle. **T** is the cycle time. The longitudinal motion parameters are the inputs for a distance controller. A comparable filter estimates the lateral motion of the leader, taking into account the yaw rate of our own vehicle.

Camera height and pitch angle are not constant during driving. Fortunately, the relevant camera parameters can be efficiently estimated using the extracted road surface points. Least squares techniques or Kalman filtering can be used to minimise the sum of squared residuals between expected and found disparities. The lane recognition benefits from this fact.

4.2 Free space analysis

Active collision avoidance is the ultimate goal of driver assistance. A careful evaluation of the depth map allows to extract free space on the road that could be used for a jink. Since this evaluation does not depend on the object grouping described above, it can be more detailed and does not depend on the heuristics used to detect and track obstacles. Averaging and thresholding the depth map with a distance dependent threshold result in a binary depth map, from which all visible points on the road plane

can be determined. Fig. 6 shows the found free space overlaid on the original image.

Alternatively, the driving corridor can be estimated from the depth map, if no other lane boundaries are present. In [7] this depth map together with the tracked lead vehicle is used to optimise the path UTA II should drive in order to avoid the collision with parking cars or suddenly occuring obstacles.

4.3 Obstacle Recognition

In traffic scenes some objects are of special interest: vehicles and pedestrians. The 3D analysis supports their recognition in three ways:

1. It detects these objects much more efficiently and reliably than most monocular approaches can do.
2. Detected objects can be pre-classified by means of their precisely measured width and height.
3. Subsequently applied classification schemes can work on image regions that have been scaled to standard size.

Potential vehicles are scaled to 32x32 pixel, pedestrians to 64x32 pixel. Fig. 7 shows such images. A neural network classification scheme using spatial receptive fields is used to recognise these objects. This approach is extremely fast, the average time for scaling and classification is less than 1 milliseconds.

This approach is sufficient for the recognition of cars and trucks. However, the recognition of pedestrians turns out to be more difficult. If the network is not sure about it's decision, further classification stages are triggered for that reason. Moving pedestrians are checked by means of a time-delay neural network that looks for the typical gate-patterns of walking pedestrians. Still pedestrians can be recognised by means of a shape based approach. These schemes are described in [8] in detail.

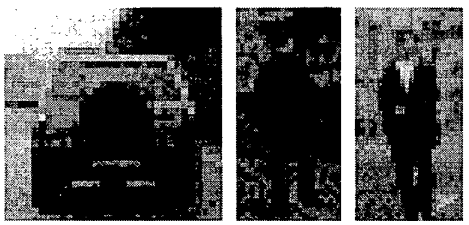

Fig. 7: Cropped and scaled image regions for subsequent classification

5 Summary

Stereo vision is a powerful approach for the interpretation of complex scenes. The described algorithm generates disparity images with sub-pixel accuracy in real-time on a standard PC. Thus, no specialized hardware is necessary any longer to reach this important goal. The scheme has proven reliability in daily traffic even under bad weather conditions including rain and snow.

References

[1] U.Franke, D.Gavrila, S.Görzig, F.Lindner, F.Paetzold, C.Wöhler: „Autonomous Driving Goes Downtown", IEEE Intelligent Systems, Vol.13, No.6, Nov./Dec.. 1998, S.40-48

[2] O.Faugeras: „Three-Dimensional Computer Vision", MIT Press, 1993

[3] L.Kaminski et. al.: „A sub-pixel stereo vision system for cost-effective intelligent vehicle applications", Intelligent Vehicles '95, 25./26. Sept. 1995, Detroit, pp.7-12

[4] K.Saneyoshi: „3-D image recognition system by means of stereoscopy combined with ordinary image processing", Intelligent Vehicles '94, 24.-26. Oct. 1994, Paris, pp.13-18

[5] L.Matthies et. al.: „Obstacle detection for unmanned ground vehicles: a progress report", Intelligent Vehicles '95, 25./26. Sept. 1995, Detroit, pp.66-71

[6] U.Franke, I.Kutzbach: „Fast Stereo based Object Detection for Stop&Go", Intelligent Vehicles '96, Tokyo, 19./20.Sept.1996, S. 339-344

[7] F.Paetzod, U.Franke, W.v.Seelen:"Lane Recognition in Urban Environment using Optimal Control Theory", Proc. IEEE Intelligent Vehicles, October 2000, submitted

[8] U.Franke, D.Gavrila, A.Gern, S.Goerzig, R.Janssen, F.Paetzold and C.Wöhler: „From door to door – principles and applications of computer Vision for driver assistant systems", *in Intelligent Vehicle Technologies: Theorie and Applications*, Arnold, 2000