Image and Vision Computing 18 (2000) 367–376
www.elsevier.com/locate/imavis

An image processing system for driver assistance

U. Handmann[a,*], T. Kalinke[a], C. Tzomakas[a], M. Werner[b], W.v. Seelen[a]

[a]*Institut für Neuroinformatik, Ruhr-Universität Bochum, D-44780 Bochum, Germany*
[b]*ZN GmbH, Universitätsstraße 160, D-44801 Bochum, Germany*

Received 3 March 1999; received in revised form 28 August 1999; accepted 14 September 1999

Abstract

Systems for automated image analysis are useful for a variety of tasks. Their importance is still growing due to technological advances and increased social acceptance. Especially driver assistance systems have reached a high level of sophistication. Fully or partly autonomously guided vehicles, particularly for road traffic, require highly reliable algorithms due to the conditions imposed by natural environments. At the *Institut für Neuroinformatik*, methods for analyzing driving relevant scenes by computer vision are developed in cooperation with several partners from the automobile industry. We present a system extracting important information from an image taken by a CCD camera installed at the rear-view mirror in a car. The approach is divided into a sequential and a parallel phase of sensor and information processing. Three main tasks, namely initial segmentation (object detection), object tracking and object classification are realized by integration in the sequential phase and by fusion in the parallel phase. The main advantage of this approach is integrative coupling of different algorithms providing partly redundant information.

Keywords: Driver assistance; Machine vision; Data fusion

1. Introduction

Some systems presented in Refs. [1–3] show the principal feasibility of driver assistance systems based on computer vision. Although exclusively vision-based systems and algorithms are not yet powerful enough to solve all driving-relevant tasks, a large amount of different scenarios can be interpreted sufficiently. Additionally sensors like radar and lidar extend the range of sensor information available for building a reliable system. The main focus of our system lies in combining various methods for the analysis and interpretation of images and in the fusion of a large spectrum of sensor data for extracting the most reliable information for the final planning and for prediction of the behavior of other traffic participants.

The great variety of different scenarios as well as the high degree of reliability necessary for the given task require an encompassing and flexible system architecture. Reliability of the reached solution, the variety of geometric appearances of involved objects and environmental constraints of both deterministic as well as statistical nature necessitate a multitude of partial solutions based on different representations of the environment. Consequently, complexity and structure of the overall system have to be adaptable in order to accommodate additional methods without degeneration of already accomplished partial solutions. For this reason, even *simple* applications are encumbered by considerations concerning the overall system architecture. Basically, the overall system architecture can be divided into basic, fusion and integration algorithms. Basic methods are those providing specific partial solutions under given constraints. Results and application of the individual algorithms are not independent, resulting in an increase in redundancy making the overall system secure and reliable given a suitable coupling architecture. The necessary methods for fusion and integration ensure a flexible cooperation of the basic building blocks as well as the integrative evaluation of results. In a similar way, sequential data processing and dynamic components are necessary in order to build up an overall system and to give solutions to complex tasks.

2. Image processing system

The fusion of different sensor information and preprocessing results increases the performance of the system. The basic methods are specialized for a specific kind of sensor information. For this reason the choice of algorithm is highly dependent on the spatial characteristics of the

* Corresponding author. Tel.: +49-2343224201; fax: +49-2343214209.
E-mail addresses: uwe.handmann@neuroinformatik.ruhr-uni-bochum.de (U. Handmann), martin.werner@zn-gmbh.com (M. Werner).

0262-8856/00/$ - see front matter
PII: S0262-8856(99)00032-3

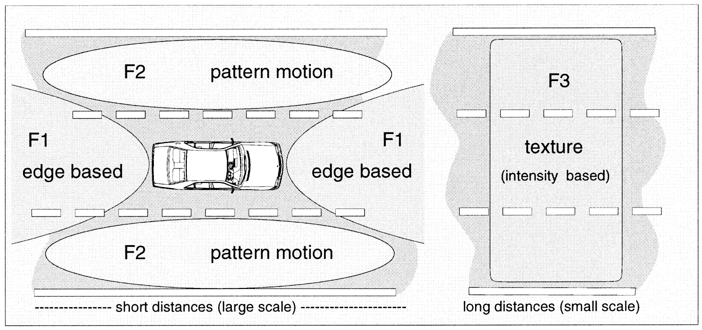

Fig. 1. Separation of the road in fields F1, F2 and F3 in which different algorithms can be applied optimally.

applications. As shown in Fig. 1, information typically differs depending on its spatial relation to the vehicle. In the area F1 contour-based methods are chosen. The sparse nature (edges) of the intensity information is sufficient due to the high resolution of the objects in the image. In addition, it speeds up computation time for real time applications. The feature we most often employ for this purpose is called local orientation coding [4]. In the field F2, we use motion detection algorithms segmenting overtaking and overtaken vehicles. In contrast to algorithms in different vision fields, we use a pattern tracking based algorithm, which ensures high stability. The long distance field F3 is analyzed by texture-based methods. The low spatial resolution makes edge-based processing infeasible. Nevertheless, the integrative characteristics of texture analysis provide good results by separating the objects from the background by use of their texture. In the area of preprocessing, a multitude of different methods for initial segmentation, object tracking, and object classification has been developed in the context of current research. A few tendencies are remarkable:

- Previous work was often based on the use of higher features, meaning the generation of a sequence of features beginning at the iconic (image-based) side and continuing to the symbolic side. There are two main reasons to do this. First, the historic rooting of image processing in material and surface inspection for quality control has led to the existence of theoretically well-founded and practically tested algorithms. Second, the symbolic features are commonly used for compact coding purposes, so that processed data amounts can be largely reduced for accommodating limited processing resources. The rapid evolution of processors has particularly alleviated the impact of this last constraint. In addition, it appears that particularly in the context of limited sensor resolution (i.e. in long-distance regions) algorithms can be employed that rely on statistical measures of extensive 'early' (in the chain of processing) feature sets.
- Often a formulation as an optimization problem can lead to implicitly robust solutions avoiding disadvantages of explicit methods (e.g. the correlation of model with image features, the correspondence problem). In this area, the increase in available computational power has contributed to scientific progress, as well.
- Particularly in natural environments, flexible algorithms possessing a certain learning capability (data-driven adaption) have desirable characteristics.

3. The basic algorithms

At the *Institut für Neuroinformatik*, algorithms providing partial solutions for object detection, tracking and classification have been incorporated into a driver assistance architecture. The following enumeration gives an overview over the applied methods:

- *Initial object detection:* local orientation coding [4,5], polygon approximation of contours [6], use of local symmetry [7], pattern motion analysis [7], texture analysis based on local image entropy [8], local variance analysis [9], local co-occurrence measures [7], shadow analysis [10,11], color analysis [12,13], and radar mapping [9].
- *Object tracking:* Hausdorff distance matching [10,14], parametric optimization [15,16], and cross entropy [17].
- *Object classification:* local orientation classifier [7], Hausdorff distance classifier [7,18], co-occurrence classifier [19], and parametric optimization classifier [15,16].

The algorithms can be classified as working on differential information (e.g. edges) or integral measurements (e.g. texture). For the application types initial object detection, tracking and classification, a description of the actually used algorithms for a real-time implementation is given.

3.1. Initial object detection

The main motivation of using multiple simple methods is that the design of a single method suitable for all conceivable scenarios seems to be impossible. Therefore, in order to provide reliable results and to ensure a fast and robust processing, a coupling of *specialists* is implemented.

The three methods used in the real time implementation, will now be described. An integration of a differential

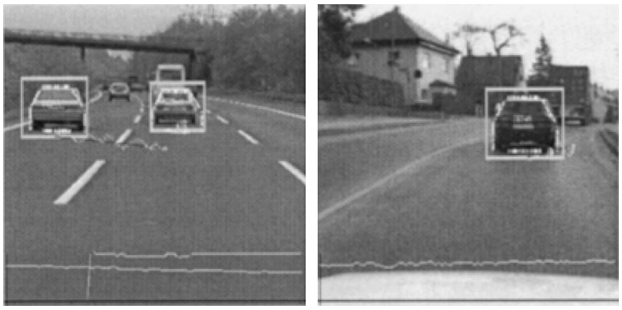

Fig. 2. Vision based object detection, object classification and object tracking.

algorithm (local orientation coding), an integrative algorithm (local image entropy) and a shadow analysis method (model-based) in the real-time system is done.

3.1.1. Local orientation coding

The 'raw' gray-scale (intensity) images are preprocessed by a method we call local orientation coding (LOC). The image features obtained by this preprocessing are bit patterns each representing a binary code for the directional gray-level variation in a pixel neighborhood. In a more formal fashion the operator is defined as

$$b'(n,m) = \sum_{i,j} k(i,j).u(b(n,m) - b(n+i, m+j) - t(i,j)),$$
$$(i,j) \in \text{neighborhood} \quad (1)$$

where $b(n,m)$ denotes the (gray scale) input image, $b'(n,m)$ the output representation, **k** a coefficient matrix, **t** a threshold matrix and $u(z)$ the unit step function. The output representation consists of labels, where each label corresponds to a specific orientation of the neighborhood. An adaption mechanism for the parameters **t** of the coding algorithm yields a high level of flexibility with respect to lighting conditions [4].

3.1.2. Local image entropy

In the local image entropy (LIE) method [8], an estimation of the information contents of a pixel based on its neighborhood is given.

The calculation is based on the information theory introduced by Shannon [20]. A part of an image can be interpreted as a signal $x(n,m)$ of k different states with the local entropy $E_{n,m}(x,y)$ determining the observer's uncertainty about this signal. This is a measure for information content. For every pixel, the normalized histogram of a centered neighborhood is calculated as an estimation of the probability distribution function $p_{n,m}(k)$

$$E_{n,m}(x,y) = -\sum_k p_{n,m}(k) \log p_{n,m}(k) \quad (2)$$

On the basis of the LIE, a saliency map is calculated for evaluating the separation of objects and background. The areas of objects and background are cut out by thresholding. A detection of objects and the free driving space on the lane can be done.

3.1.3. Shadow analysis

The detection of shadows is realized by thresholding the intensity image, applying some morphological processing and a region clustering algorithm stabilized over time. As already shown by Mori and Charkari [21] the shadow underneath a vehicle can be used for object detection. For this task the gray level of the road is analyzed in order to extract a threshold τ for shadows. Furthermore, we select those LOC features that expose horizontal orientation and correspond to a light-to-dark transition (scanning the image upwards) and group them in clusters. These clusters are subjected to further constraints (i.e. from the camera geometry [22]) and finally make up the initial hypotheses or regions of interest (ROI) [11].

3.2. Object tracking

Algorithms for object tracking are most important when a stabilization over time or a prediction of e.g. trajectories is required. As it can be seen in Fig. 1, the tracking algorithms are applied depending on the spatial resolution of the images. In the near-distance field the Hausdorff distance or order statistics are used as a measurement based on contour codes (LOC). Here we present the Hausdorff distance tracker that has been tested successfully on a large set of different image sequences. For further details of the approach using order statistics see Ref. [23]. In the long-distance field and in the case of tracking non-rigid objects, good supplementary results can be gained by texture-based cross entropy measures.

3.2.1. Hausdorff distance

The geometric comparison of shapes is a fundamental tool for model-based object recognition. Most of the methods used in object recognition systems employ a similarity measure between model features and image features [24]. The Hausdorff distance measures the divergence of a set of features from a reference set of features [25]. In the given application these sets mostly describe object contours. The comparison of similar object contours yields small distance values, whereas objects with different contours yield larger distances.

The directed Hausdorff distance h of one point set A to a point set B is the maximum of the minimum distances of each point of set A to the points of set B. The final Hausdorff distance H is simply the maximum of the two directed distances:

$$\begin{aligned} h(A,B) &= \max_{p\in A}(\min_{q\in B}(\|p-q\|)) \\ H(A,B) &= \max(h(A,B), h(B,A)) \end{aligned} \quad (3)$$

The partial Hausdorff distance performs a ranking of these

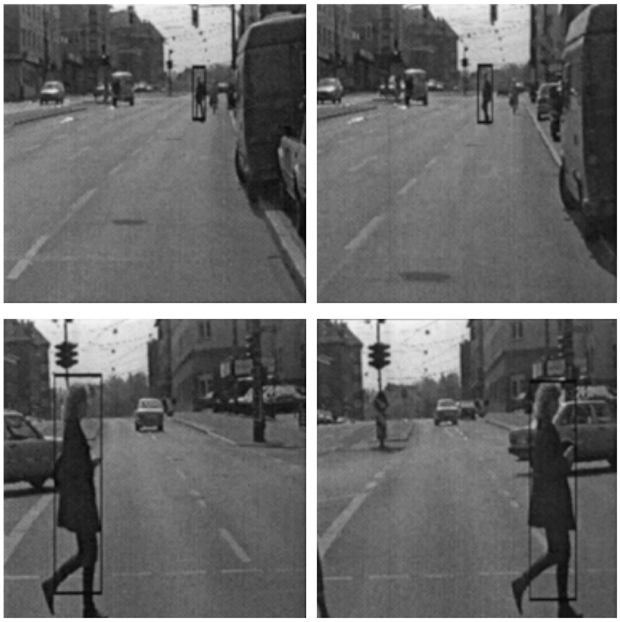

Fig. 3. Tracking of pedestrians based on the cross entropy.

minimum distances and considers a fraction of them instead of the maximum.

Unlike classical correlation methods, the Hausdorff distance uses Min–Max operations instead of multiplications, so it is more efficient. The partial Hausdorff distance is robust against partially occluded objects and outliers that may arise at the contours due to noise or insufficient feature extraction.

The partial Hausdorff distance can examine object hypotheses in a complex scene. This method was tested successfully with highway traffic scenes (Fig. 2). It was able to recognize vehicles on highways and track them over time. Two degrees of freedom (translation and scaling) were considered in our model.

3.2.2. Cross entropy

One of the simpler descriptions of textures is obtained by intensity histograms (first-order statistics). Especially, non-rigid objects like pedestrians and two-wheeled vehicles, which possess an additional rotational degree of freedom, compared to other road-users can be tracked using the cross entropy (Fig. 3).

As described by Kalinke and von Seelen [17], a matching process can be performed by comparison of two probability distributions based on the Kullback–Leibler divergence [26]. In the given application a model distribution D at time step $(t-1)$ is compared to several hypotheses at time t in the space of search (translation, scale, and deformation).

3.3. Neural classifiers for vehicles

For the task of classification, different methods are applied. Feature-based and model-based solutions have been developed. The LOC-classifier is a computationally fast method used for a first classification of a given ROI. It is intended for separating possible objects from the background. It is independent from the resolution of the objects due to normalization in size. In a second stage, two classifiers with higher computational costs perform a more reliable classification. The Hausdorff distance classifier processes objects in the near field with high spatial resolution, thus enhancing the ROI image coordinates. For details

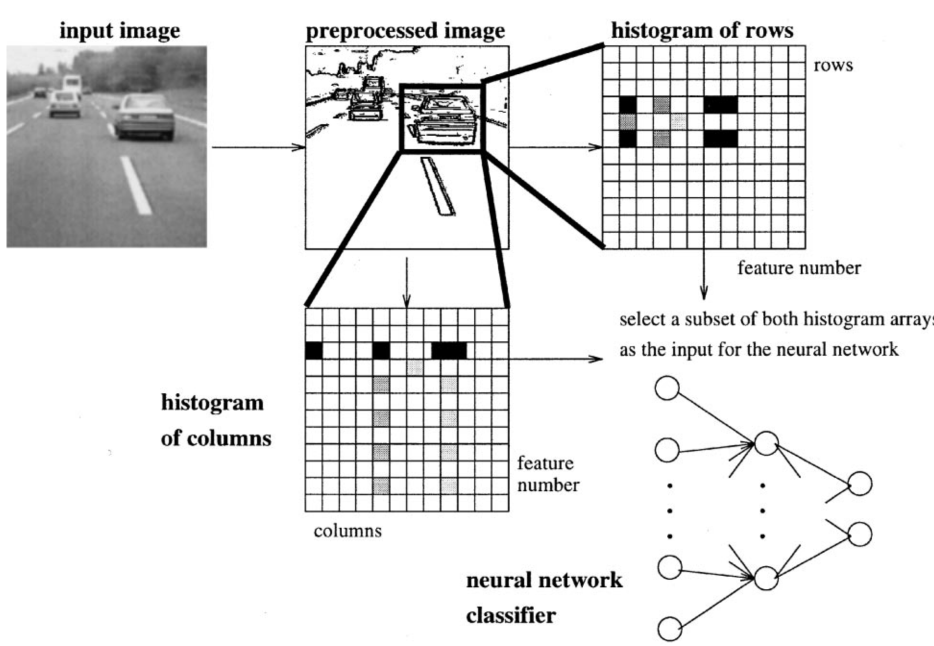

Fig. 4. The LOC-classifier process.

Fig. 5. Hausdorff distance classifier: each region is compared with two models.

of the co-occurrence classifier applied to the long-distance field, see Ref. [19].

3.3.1. LOC-classifier

With the given local orientation coding [4] described in Section 3.1.1, a classification of vehicles is realized (Fig. 4). The classifier has to cope with partial occlusions, varying illumination conditions, tilt of an object, differently resolved structures depending on the distance of the object under consideration, noise and perturbation induced by the recording and processing equipment, different viewpoints and different kinds of vehicles with different shapes and colors. Additionally, the classifier should be able to generalize from relatively few training examples to the necessary features characterizing a vehicle.

Therefore, a neural network has been chosen for solving the classification task. It is a feed-forward neural network with one hidden layer trained by the error back-propagation algorithm [27]. These networks are known to be universal estimators for any continuous-valued function [28]. Furthermore, it is shown that these structures can, with some small modifications, approximate a posteriori probabilities in the sense of a Bayesian classifier [29].

The inputs for the classifier are certain subsets of the histograms. The output is the class of the region. The complete system has been implemented and extensively tested on the Mercedes Benz VITA II test vehicle [30]. Different classes of vehicles have been trained. For a further evaluation of the system see Ref. [31].

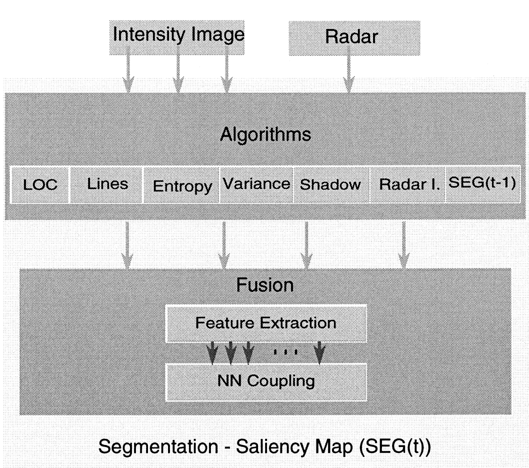

Fig. 6. The coupling model of the fusion process.

3.3.2. Hausdorff distance classifier

The geometric property of the Hausdorff distance leads to the idea of classifying various vehicles into separate classes according to the imposed dissimilarity measure. Because of the need for defining a reference contour for each class, we deal here with a model-based approach. The design of accurate models (prototypes) is of great importance for our task. At a first step, the Hausdorff distance is used for the classification of cars and trucks. Due to the fact that rear views of cars differ significantly from rear views of trucks, one can expect that the design of generic models for each class can accomplish the separation of the objects of both classes.

The classification works as follows: each region is compared with two models, a car model and a truck model (Fig. 5). The features of the region and the models have been extracted using the LOC. For more robust results the horizontal features are separated from the verticals for both the region and the models.

The Hausdorff distance is computed for each model over all possible translations inside the region and a certain range of scales and deformations. The fractions of the features of the forward (model to image) and the backward (image to model) match that are consistent with a given distance threshold constitute the criteria for its classification for each model. These values are learned by a multi-layer perceptron (MLP) network using the back-propagation algorithm.

4. The concept of fusion

Data fusion is one of the main goals to be achieved if a large amount of stability and reliability is necessary like in driver assistance systems. On the one hand, a gain in robustness is reached by creating higher redundancy so that poor or missing results of one data stream do not affect the overall result decisively [19]. On the other hand, the varying types of objects and background constellations demand a large spectrum of data to be processed to solve the given task. Three different types of neural coupling mechanisms are introduced [9]. The high flexibility and the possibility of extension and adaptivity of the retraining processes have led to the choice of neural networks.

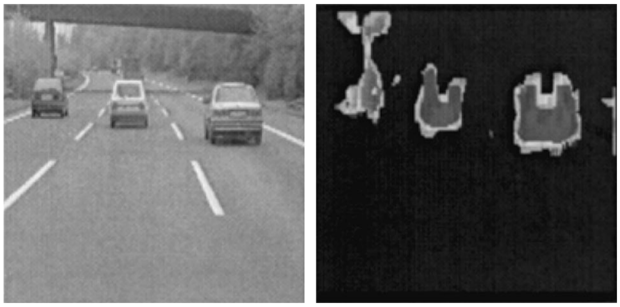

Fig. 7. Image and saliency map.

The aim of fusion in computer vision is to get an improvement over special solutions and single methods by a coupling net (parallel branch). The modular coupling of single processing steps generates redundancy necessary for object recognition. By this, greater flexibility and robustness of the image processing modules and higher adaptation of the modules regarding to the problems should be achieved. In Fig. 6 a principal fusion process for segmentation is shown. A prototype of this process has been implemented by Handmann et al. [9]. Computer vision modules, generating lines (polygon approximation of the contour) [6], local orientation coding [4], local image entropy [32], and local variance analysis [9] are coupled in a neural network. In the present implementation a fusion process on the feature level is selected (MLP with a 16-5-1 structure) for generating a saliency map (Fig. 7). A feedback over time is realized, additional sensor information (radar) are integrated.

For the detection of pedestrians a higher-level fusion process is realized (Fig. 8). We propose a temporal fusion of walking model matches using the Hausdorff distance and the LIE. For the final classification, measurement of independent motion is incorporated. This leads to the concept of integration.

5. The concept of integration

The system for the object-related analysis is shown in Fig. 9. The concept of integration (sequential branch) of separate steps to a reliable working system is mainly based on feed-

Fig. 8. Image and pedestrian hypotheses.

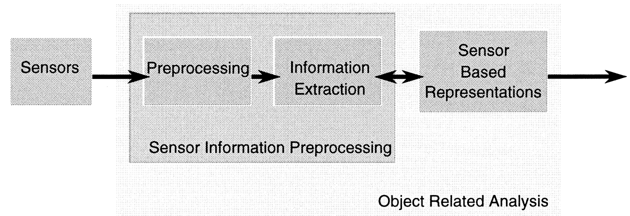

Fig. 9. Structure of the object-related analysis.

back of results for determination of the expected value. In the following subsections, the sensor information processing and the sensor-based representational part are described.

5.1. Sensor information processing

In this section the sensor information processing shown in Fig. 10 is described. The intensity image and radar signals are used as sensor input. The results of basic preprocessing algorithms are fed into a neural fusion architecture yielding the initial object detection that provides hypotheses of possible location of vehicles. The very fast LOC-classifier reduces the set of hypotheses. An internal stabilization over time ensures further robustness. In order to confirm the hypothesis, an object tracking is performed where the object size and type decides whether the Hausdorff tracking or the cross entropy tracking have to be used. The estimates for scale, position, and confidence are fed into the main stream and to the modular classifier. A neural network fed with object size, Hausdorff and co-occurrence classification results determines what type of object has been tracked. The object tracking is performed at every time step. The initial object detection can work at a slower time rate. Finally, the classification provides results on larger time scales due to the fact that a tracked object with high confidence values will not change its class. Hence object tracking is the most important task next to the detection process. To ensure stable tracking over time, a Kalman filter is utilized.

The main feedback stream gathers all the results of the single tasks. The type of information is changing from an iconic (preprocessing) to a symbolic (classification) description. A global data representation is built. Here the integration of the different processing steps is accomplished. Task-dependent pixel-oriented saliency maps (sensor based representations) are implemented.

5.2. Sensor-based representations

In the sensor-based representational part of the object-related analysis the data are combined consistently for each sensor. Representations in general can be subdivided into functional modules. They perform the consistent integration of the processed sensor data over time. Each representation has data integration and a knowledge integration module. An internal memory and internal dynamics have to be organized (Fig. 11).

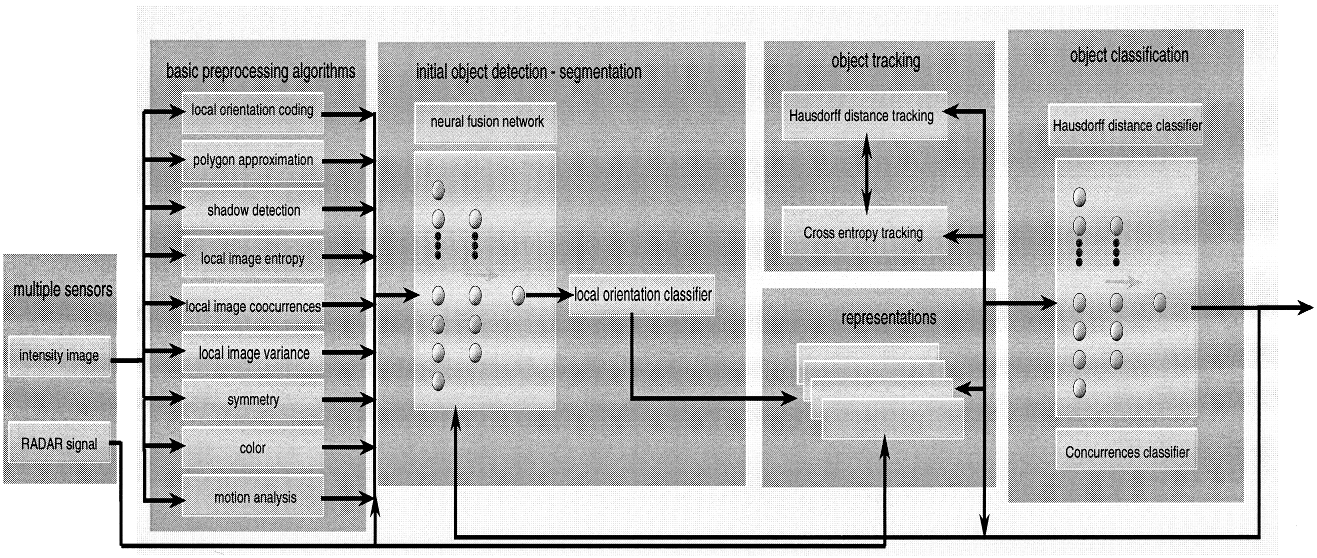

Fig. 10. Sensor information processing. In the preprocessing part an extraction of relevant features from multiple sensors is done. In the segmentation part a detection of initial object hypotheses is realized. In the object tracking and classification parts a verification of the hypotheses are performed. In the sensor-based representations task-dependent saliency maps are implemented to verify the output of the segmentation, the classification and the tracking part of the system.

To be able to compare the results from different sensors, each representation has to describe a common database on a comparable time scale (data integration). In this part, the results of the information processing are evaluated in sensor coordinates according to consistency and discrepancy or ambiguity of information. The results of the sensor information processing stage are stabilized in movement-sensitive representations by introducing a third dimension, the time dimension. In this sense, a ROI is accepted as a valid hypothesis only if it has a consistent history. This is implemented by spatio-temporal accumulation using different representations with predefined sensitivities (Fig. 12). The sensitivities are functions of objects' supposed relative velocity and distance to the observer. In order to apply a time stabilization to these regions and decide whether they are valid or not, a prediction of their position in the knowledge integration part is realized. A competition between the different representations and a winner-takes-all mechanism ensures reliable object detections.

This prediction can also be useful for scene interpretation, since preceding vehicles can be discriminated from oncoming ones. The prediction requires knowledge about the road trajectory. When the road boundaries can be localized in the image (e.g. either from GPS/road map information or by a vision-based approach [1]), then the trajectory of oncoming vehicles can be estimated, since it lies approximately parallel to the road boundaries. A trapezoid road model is assumed for small and medium distances. Detected objects are assumed having a constant relative velocity v_r (regarding only translatory motion without rotational components) within the time interval of two successive frames. Within this interval Δt, the running distance of the object is given by $\Delta s = v_r \Delta t$ in the real world. Using the perspective geometry and assuming zero tilt for reasons of simplicity, the motion of the object in the image plane is calculated.

The translation in the vertical direction of an oncoming vehicle in the next frame depends on its translational

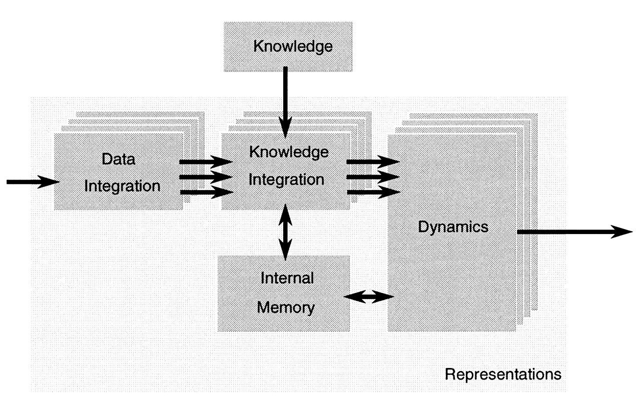

Fig. 11. Structure of the representations.

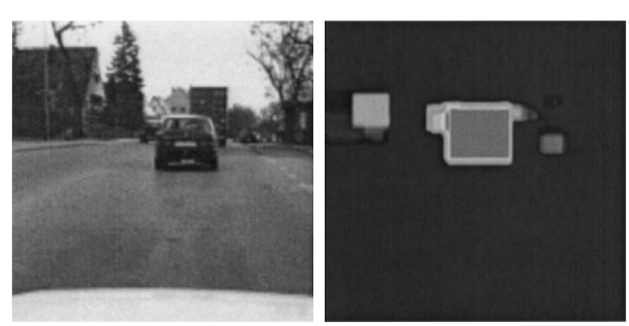

Fig. 12. Image and representation.

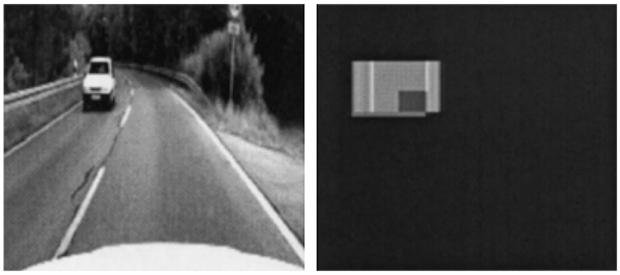

Fig. 13. Image and representation: prediction and detection of oncoming objects.

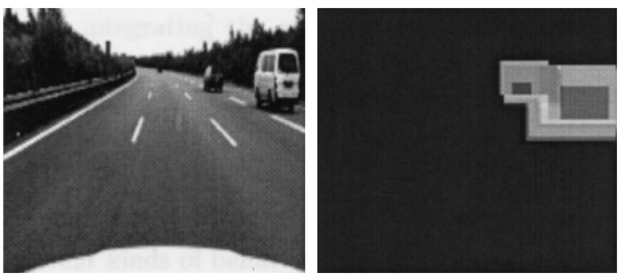

Fig. 14. Image and representation: prediction and detection of objects to be overtaken.

velocity and its current position (height) in the image. Its lateral translation can be estimated as the ratio of the trajectory slope and the vertical translation. Thus, for each detected region its position is predicted for a finite number of frames and the predicted ROIs are registered in an accumulator field in a similar way as in the case of preceding vehicles. The predicted ROIs are verified along with newer regions at each frame separately, and the total activity (volume) of a region within the time window is the criterion for the detection of oncoming objects (Fig. 13).

The translation principle can be easily adapted for the prediction of oncoming vehicles on the right (overtaking task, Fig. 14). Finally, possible contradictions between representations are solved by a competing winner-takes-all mechanism so that a reliable object detection is provided.

6. Real-time implementation

Due to the increase of computational power and the development of reliable algorithms, fusion and integration of basic methods for specific problems can be performed to realize an overall stable system. The stability and robustness over single algorithms is largely increased. Because the overall computational demands are still quite large when using actual standard hardware, a spin-off was realized. Hence if real-time operation is necessary the processing has to be limited to a selection of algorithms due to limited computational resources.

In the real-time implementation the initial object detection is restricted to a shadow- and a LIE-analysis including a LOC-classification. The objects are tracked by the Hausdorff tracker and classified by the Hausdorff classifier in order to use just one preprocessed feature map. On a standard DEC Alpha (500 MHz), the system needs 10 ms for the initial segmentation including a time stabilization, the LOC-classification needs 2 ms for every ROI, the tracking is performed in about 5 ms per object and finally the classification takes about 10 ms per object. As mentioned before, the classification does not need to be calculated for every

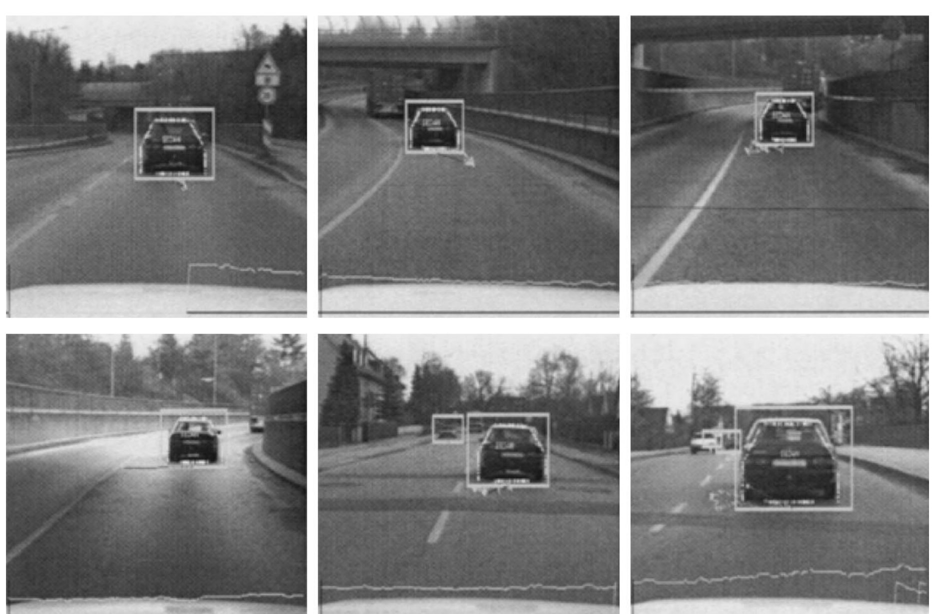

Fig. 15. Object detection, object tracking and object classification on a sequence of 1000 frames under various lighting conditions, different relative velocities and a large range of distances.

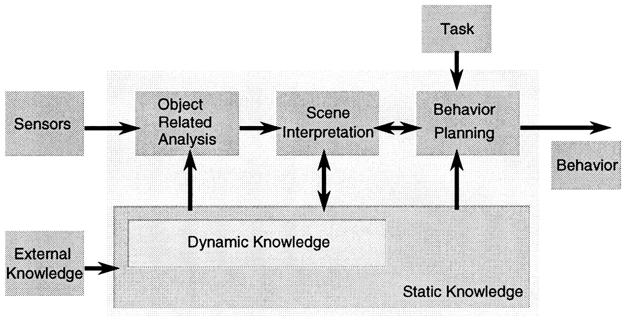

Fig. 16. Architecture for a driver assistance system.

frame. The implementation of the sensor-based representations requires 2 ms per frame.

This system is capable of obeying the real time requirements, but the processing cannot cope with all different scenarios and we have restricted this application to extra-urban roads and motorways. In Fig. 15, the results for a sequence of 1000 frames with various lighting conditions are shown (regions of interest, class, scaling, and the position of the vehicle). Nevertheless, if the performance of the hardware components will increase, the presented overall system will be able to cope with most of the scenarios even in more complex situations.

7. Conclusion and discussion

For driver assistance systems, the organization of different kinds of behavior according to given tasks is necessary. In this paper we presented an image processing system integrating the results and experiences of a long period of research in computer vision. An object-related analysis based on a vision sensor was described. Objects are perceived by a segmentation, a tracking and a classification task. The object hypotheses are used to build movement-sensitive representations to get detailed information about objects in front of the car. The results are shown in Fig. 15.

To organize different kinds of behavior we propose an architecture (Fig. 16) where the extracted information of the presented image processing system can be used [33]. In this architecture, the information about the actual state of the environment is perceived by the system's sensors. The data collected by each sensor have to be processed and interpreted to gain the desired information for the actual task [34]. This is done by the object-related analysis. The object-related analysis has to provide the scene interpretation with information. In the scene interpretation, the different results have to be interpreted and integrated to achieve consistent results, and the behavior-relevant information has to be presented to the behavior planning. The behavior planning is the final element that has to evaluate which action should be taken to achieve the current task based on the actual information from the scene interpretation and the actual knowledge. It also has to decide whether the current decision or advice is reliable and can be proposed to the driver (Zhuang et al. [35] use a fuzzy controller for this element). The actual behavior planning should influence the scene interpretation to produce the optimal amount of information needed.

References

[1] A. Broggi, A massively parallel approach to real-time vision-based road markings detection, Proceedings of the Intelligent Vehicles'95 Symposium, Detroit, USA (1995) 84–85.

[2] E.D. Dickmanns, et al., seeing passenger car VaMoRs-P, Proceedings of the Intelligent Vehicles'94 Symposium, Paris, France (1994) 68–73.

[3] D. Pomerleau, RALPH: rapidly adapting lateral position handler, Proceedings of the Intelligent Vehicle'95 Symposium, Detroit, USA (1995) 506–511.

[4] C. Goerick, D. Noll, M. Werner, Artificial neural networks in real time car detection and tracking applications, Pattern Recognition Letters (1995).

[5] C. Goerick, Local orientation coding and adaptive thresholding for real time early vision. Internal Report IRINI 94-05, Institut für Neuroinformatik, Ruhr-Universität Bochum, D-44780 Bochum, Germany, June 1994.

[6] ELTEC Elektronik GmbH, Mainz. THINEDGE-processor for contour matching, Hardware Manual, Rev. 1A, 1991.

[7] W. von Seelen et al., Image processing of dynamic scenes. Internal Report IRINI 97-14, Institut für Neuroinformatik, Ruhr-Universität Bochum, D-44780 Bochum, Germany, July 1997.

[8] T. Kalinke, W. von Seelen, Entropie als Maß des lokalen Informationsgehalts in Bildern zur Realisierung einer Aufmerksamkeitssteuerung, Mustererkennung, Springer, Heidelberg, (1996) 627–634.

[9] U. Handmann, G. Lorenz, T. Schnitger, W. von Seelen, Fusion of different sensors and algorithms for segmentation, IV'98, IEEE International Conference on Intelligent Vehicles 1998, Stuttgart, Germany, 1998 (1998) 499–504.

[10] U. Handmann, T. Kalinke, C. Tzomakas, M. Werner, W. von Seelen, Computer vision for driver assistance systems, Proceedings of SPIE, Orlando, 1998, SPIE Session Enhanced and Synthetic Vision 3364 (1998) 136–147.

[11] C. Tzomakas, W. von Seelen, Vehicle detection in traffic scenes using shadows. Internal Report IRINI 98-06, Institut für Neuroinformatik, Ruhr-Universität Bochum, D-44780 Bochum, Germany, August 1998.

[12] U. Handmann, G. Lorenz, W. von Seelen, Fusion von Basisalgorithmen zur Segmentierung von Straßenverkehrsszenen, Mustererkennung, Springer, Heidelberg, 1998, pp. 101–108.

[13] G. Lorenz, Erzeugung eines adaptiven Objektformmodells unter Ausnutzung des Farbverteilungskontrastes, in: Proceedings 9, Aachener Kolloquium Signaltheorie Bild- und Sprachsignale, pp. 179–182, RWTH Aachen, 1997.

[14] M. Werner, Objektverfolgung und Objekterkennung mittels der partiellen Hausdorff-Distanz, Number 474, in: Fortschrittberichte, Reihe 10, VDI-Verlag, Düsseldorf, 1999. Dissertation, Ruhr-Universität Bochum.

[15] D. Noll, Ein Optimierungsansatz zur Objekterkennung, Number 454, in: Fortschrittberichte, Reihe 10. VDI-Verlag, Düsseldorf, 1996, Dissertation, Ruhr-Universität Bochum.

[16] D. Noll, W. von Seelen, Object recognition by deterministic annealing, Image and Vision Computing 15 (1997) 855–860.

[17] T. Kalinke, W. von Seelen, Kullbach-Leibler Distanz als Maß zur Erkennung nicht rigider Objekte, Mustererkennung, Springer, Heidelberg, 1997, pp. 501–508.

[18] U. Handmann, T. Kalinke, C. Tzomakas, M. Werner, W. von Seelen, An image processing system for driver assistance, IV'98, IEEE

International Conference on Intelligent Vehicles 1998, Stuttgart, Germany, 1998 (1998) 481–486.
[19] U. Handmann, T. Kalinke, Fusion of texture and contour based methods for object recognition, in: ITSC'97, IEEE Conference on Intelligent Transportation Systems 1997, Boston, 1997, IEEE Session 35: Intelligent Vehicles: Vision (3).
[20] C.E. Shannon, A mathematical theory of communication, Bell Systems Technical Journal 27 (1948) 379–423 see also pp. 623–656.
[21] H. Mori, N.M. Charkari, Shadow and rhythm as sign patterns of obstacle detection, International Symposium on Industrial Electronics (1993) 271–277.
[22] T. Kalinke, C. Tzomakas, Objekthypothesen, in: Verkehrsszenen unter Nutzung der Kamerageometrie. Internal Report IRINI 97-07, Institut für Neuroinformatik, Ruhr-Universität Bochum, D-44780 Bochum, Germany, March 1997.
[23] M. Werner, W. von Seelen, Using order statistics for object tracking, in: ITSC'97, IEEE Conference on Intelligent Transportation Systems 1997, Boston, 1997, Session 30: Intelligent Vehicles: Vision (2).
[24] D.W. Paglieroni, Distance transforms: properties and machine vision applications, CVGIP 54 (1) (1991) 56–74.
[25] D.P. Huttenlocher, Comparing images using the Hausdorff distance, IEEE Transactions on PAMI 15 (9) (1993).
[26] S. Kullback, R.A. Leibler, On information and sufficiency, Ann. Math. Stat. 22 (1951) 79–86.
[27] J.A. Hertz, A.S. Krogh, R.G. Palmer, Introduction to the Theory of Neural Computation, Addison Wesley, Reading, MA, 1991.
[28] K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, Neural Networks 2 (1989) 359–366.
[29] M. Finke, K.-R. Müller, Estimating a posteriori probabilities using stochastic network models, Proceedings of the Summer School on Neural Networks, Bolder, CO (1993) 276–281.
[30] M.E. Brauckmann, C. Goerick, J. Groß, T. Zielke, Towards all around automatic visual obstacle sensing for cars, Proceedings of the Intelligent Vehicles'94 Symposium, Paris, France (1994) 79–84.
[31] S. Bohrer, T. Zielke, V. Freiburg, An intergrated obstacle detection framework for intelligent cruise control on motorways, Proceedings of the Intelligent Vehicles Symposium, Detroit (1995) 276–281.
[32] C. Curio, J. Edelbrunner, T. Kalinke, C. Tzomakas, W. von Seelen, Walking pedestrian recognition, ITSC'99, IEEE Conference on Intelligent Transportation Systems 1999, Tokyo (1999).
[33] U. Handmann, I. Leefken, C. Tzomakas, A flexible architecture for intelligent cruise control, ITSC'99, IEEE Conference on Intelligent Transportation Systems, Tokyo, 1999 (1999).
[34] U. Handmann, I. Leefken, C. Tzomakas, A flexible architecture for driver assistance, SPIE's International Symposium on Intelligent Systems and Advanced Manufacturing, Boston 1999 (1999) Mobile Robots and Autonomous Systems.
[35] Qiang Zhuang, Martin Kreutz, Jens Gayko, Optimization of a fuzzy system using evolutionary algorithms, Proceedings of the Fuzzy-Neuro Systems 98, München, Germany 1998, pp. 178–185.