Towards a Human-like Vision System for Driver Assistance

Jannik Fritsch[◇], Thomas Michalke[⋆], Alexander Gepperth[◇]
Sven Bone[‡], Falko Waibel[‡], Marcus Kleinehagenbrock[‡], Jens Gayko[‡], Christian Goerick[◇]

[⋆]Darmstadt University of Technology
Institute for Automatic Control
D-64283 Darmstadt, Germany
thomas.michalke
@rtr.tu-darmstadt.de

[◇]Honda Research Institute Europe GmbH
D-63073 Offenbach, Germany
{jannik.fritsch,alexander.gepperth,
christian.goerick}
@honda-ri.de

[‡]Honda R&D Europe (Deutschland) GmbH
D-63073 Offenbach, Germany
{sven_bone,falko_waibel,
marcus_kleinehagenbrock,
jens_gayko}@de.hrdeu.com

***Abstract*— Several Advanced Driver Assistance Systems realizing elementary perception and analysis tasks have been introduced to market in recent years. For example, collision mitigation brake systems detect the distance and relative velocity of vehicles in front to assess the risk of a rear-end collision in a clearly defined following situation. In order to go beyond such elementary analysis tasks, today's research is focusing more and more on powerful perception systems for driver assistance. We believe computer vision will play a central role for achieving a full understanding of generic traffic situations. Besides individual processing algorithms, general vision architectures enabling integrated and more flexible processing are needed. Here we present the first instantiation of a vision architecture for driver assistance systems inspired by the human visual system that is based on task-dependent perception. Core element of our system is a state of the art attention system integrating bottom-up and top-down visual saliency. Combining this task-dependent tunable visual saliency with object recognition and tracking enables for instance warnings according to the context of the scene. We demonstrate the performance of our approach in a construction site setup, where a traffic jam ending within the site is a dangerous situation that the system has to identify in order to warn the driver.**

Keywords: System Architecture; Driver Assistance Systems; Vision System; Top-down / Bottom-up Saliency

I. INTRODUCTION

Today's Advanced Driver Assistance Systems (ADAS) support effectively the driver in clearly defined traffic situations like keeping the distance to the forward vehicle. For this purpose RADAR sensors, LIDAR sensors, and cameras are used to extract parameters of the scene, like, e.g., headway distances, relative velocities, and relative position of lane markers ahead. This approach resulted in specialized commercial products improving driving safety (e.g., the "Honda Collision Mitigation Brake System" [1], [2] to help the driver to avoid rear end collisions in case the forward vehicle brakes unexpectedly). Although traffic rules and road infrastructure like, e.g., lane markings restrict the complexity of what to sense while driving, perception systems of today's ADAS are capable of recognizing simple traffic situations only. Furthermore driving in normal traffic scenes can be done mainly in a rather reactive way by staying in the middle of the lane and keeping an appropriate distance.

For assisting the driver over the full range of driving tasks in all kinds of challenging situations and going beyond simple reactive behaviors, a more sophisticated task-dependent processing strategy is required. We see two major challenges for achieving this target:

- an adequate organization of perception using a generic vision system,
- a behavior planning system capable of predicting the driving situation and generating safe trajectories.

We focus in this paper on the first challenge. One possible way to solve this challenge is to realize a task-dependent perception using top-down links. In this paradigm, the same scene can be decomposed in different ways depending on the current task. A promising approach is to use an attention system that can be modulated in a task-oriented way, i.e., based on the current context. For example, while driving at high speed, the central field of the visual scene becomes more important than the surrounding. Furthermore only if the vision system attends fast enough to the relevant parts of the surrounding traffic and obstacles, it will be able to assist the driver in all dangerous situations.

Aiming towards such a task-dependent vision system, this paper describes a vision architecture that is being developed as perceptual front-end of an ADAS. The proposed system provides a framework that enables task-dependent tuning of visual processes via object-specific weighting of input features of the attention system. The system generates an appropriate reaction in dangerous situations (autonomous braking). Its architecture is inspired by findings of human visual system research and organizes the different functionalities in a similar way. For a first proof of concept, we focus on assisting the driver during a critical situation in a construction site. The system has been implemented using a software framework for component integration and is evaluated on a number of test streams. It achieves real-time performance on a prototype car which has been demonstrated live on a testing range.

The paper is organized as follows: We start in Section II by relating our work to research on visual attention systems and existing vision architectures for ADAS applications. Subsequently, Section III provides an overview of the system architecture and the individual components. For the analysis of the attention system, we evaluated the construction

site scenario to illustrate the performance of the top-down approach in a complex environment. The obtained results demonstrating the feasibility and benefits of top-down attention in a complex ADAS are described in Section IV. The paper concludes with a summary and an outlook in Section V.

II. RELATED WORK

In recent years several prototype vehicles being able to perform several driving tasks autonomously have been presented. Just recently this topic is gaining public interest as documented by the DARPA Urban Challenge [3] and the European Information Society 2010 *Intelligent Car Initiative* [4] as well as several European Projects like, e.g., Safespot [5] or PReVENT [6].

In terms of complete vision systems, one of the most prominent examples is a system developed in the group of E. Dickmanns [7]. It uses several active cameras mimicking the active nature of gaze control in the human visual system. However, the processing framework is not closely related to the human visual system. Without a tunable bottom-up attention system and with top-down aspects that are limited to a number of object-specific features for classification, no dynamic preselection of image regions is performed. Further research on complete architectures for intelligent vehicles has been presented by Franke [8] and Broggi [9] but these approaches focus mainly on a computational framework or the combination of several reactive systems. They lead to impressive results in specific scenarios and offer a good scalability in terms of computational aspects, but the challenge of functional integration and interaction is not yet fully solved.

With regard to vision systems developed for ADAS, there have been few attempts to incorporate aspects of the human visual system into complete systems. With respect to attention processing, a saliency-based traffic sign detection and recognition system was proposed by Ouerhani [10]. A more biologically inspired approach has been presented by Färber [11]. This publication as well as the recently started German Transregional Collaborative Research Centre 'Cognitive Automobiles' [12] address mainly human inspired behavior planning whereas our work currently focuses more on the task-dependent perception aspects.

III. SYSTEM

In the following, a rough overview of the implemented vision system structure for driver assistance is given. Subsequently, crucial system parts are described in more detail.

A. Overview

The overall architecture concept to realize task-based visual processing is depicted in Fig. 1. It contains a distinction between a 'what' and a 'where' processing path, somewhat similar to the human visual system where the ventral and dorsal pathway are typically associated with these two functions. Among other things, the 'where' pathway in the human brain is believed to perform the localization and coarse tracking of a small number of objects that are relevant for the current task. This tracking is performed by the human visual system without focusing the eye gaze on individual objects to be tracked [13], i.e., tracking does not require high resolution. In contrast, the 'what' pathway considers the detailed analysis of a single spot in the image. In the human visual system this is intimately bound to the current eye gaze, as the human eye possesses a high resolution in the central 2-3° (foveal retina area) of the visual field only.

In our vision system the eye gaze is performed virtually as the camera mounted in the car has a constant resolution in the complete field of view. Changing the eye gaze is therefore equivalent to shifting the processing to another spot of the input image. This spot is analyzed in the 'what' pathway in full resolution while the whole image is analyzed in the 'where' path in lower resolution. Processing in these two pathways is believed to occur in parallel in the human brain, but their intertwinings are as yet not known in too much detail. We here adopt the idea of continuously tracking a small number of objects in each image of the incoming visual stream to coarsely represent the current scene and at the same time acquiring more detailed information on one additional object. We therefore have two analysis processes running in parallel in our system, indicated by the two circular arrows in Fig. 1.

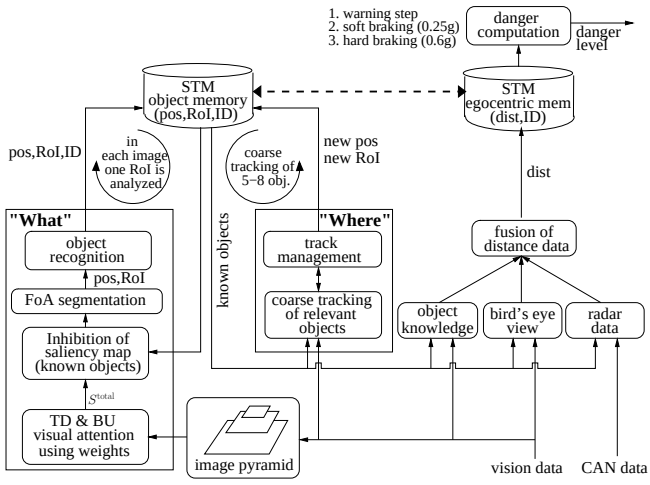

Fig. 1. Architecture concept of vision-based driver assistance system.

The detailed organization of the two processing streams in our architecture concept is as follows: The input image is analyzed in the 'what' path (depicted left in Fig. 1) for salient locations using a variety of visual features including orientation, intensity, color, and motion. This visual attention combines Bottom-Up (BU) and Top-Down (TD) pathways and is described in more detail in Section III-B. The resulting saliency map S^{total} is modulated by suppressing image regions that contain known objects, i.e., that have been detected earlier. The system stores all detected objects in a so-called Short Term Memory (STM) that provides the position information of known objects as top-down link. The suppression of saliency areas is also known as Inhibition of Return (IoR) in the human visual system [14]. The performance gain of

using this IoR approach and the influence on the STM will be shown in Section IV. A simple maximum search is used on the resulting saliency map to find the currently most salient point in the scene, the Focus of Attention (FoA). At this position the Region of Interest (RoI) is determined by region growing on the overall saliency map using the FoA as seed. In the final step of the 'what' cycle, the resulting RoI as well as its position (pos) are fed to the fast feedforward object recognition system (see Section III-C).

After object recognition, the image region, its position, and the object label (pos, RoI, ID) are stored in the STM in order to be tracked coarsely in subsequent images in the 'where' path. Before insertion, it is checked whether the new object can be associated to a known object based on its position, size, and label; if a matching object is found, the object already stored in the STM is updated. One iteration is concluded by calculating for all objects in the STM their distance (dist) based on fusing measurements from radar, depth from familiar object size (i.e., object knowledge, [15]) and from bird's eye view [16] using an Extended Kalman Filter (see Section III-D). The distance information is stored in a separate egocentric representation that is directly suitable for calculating the current danger level and generating a warning message if necessary.

All objects contained in the STM are constantly tracked in the 'where' path based on an appearance-based tracker that uses a second order motion model for prediction and a local correlation step for the refinement of the new object positions. In each iteration the position is updated in the STM and a new template RoI is stored. In case the prediction does not match (no good correlation found) the object is deleted from the STM and therefore its position will not be inhibited in the 'what' pathway anymore. Consequently, the attention will be focused on the missing object in one of the next images if the object is still present and salient. This way, all objects being recognized and behaving as predicted are coarsely tracked while the 'what' attention is always focused on new objects and objects behaving unexpectedly.

However, it should be avoided that objects that can be tracked are stored in the STM forever, as this would mean that the system cannot correct a wrong object label. This is achieved by deleting an object from the STM after N frames, i.e., objects have a lifetime of N frames. This is equivalent to limiting the capacity of the STM to N objects in scenes with more than N objects. Note that the rather simple tracking method is sufficient for many applications in the automotive domain where most objects are rigid (e.g., a car) and therefore the main appearance changes are caused by small translations and scalings. One notable exception are pedestrians, for which a specialized detection and tracking will be needed.

The novelty of our architecture lies in the introduction of top-down aspects (like, e.g., task-dependent tunable attention generation via sets of weights and, in parallel, inhibiting known object positions predicted by tracking) resulting in the ability to cope with highly dynamic traffic scenes using limited computational resources. The top-down tunable attention system is a key aspect of our ADAS, since such preprocessing leads to a considerable reduction of scene complexity by restricting further processing steps to image regions that are interesting according to the current system task. This saves not only computational resources but we implicitly reduce the number of false positives as, e.g., the object recognition only gets RoIs that are likely to be a car based on their current saliency profile.

B. Attention Sub-System

A rough sketch of the visual attention sub-system is depicted in Fig. 2, for a more detailed description please see [17]. Our attention sub-system consists of a number of features that are extracted from the image on 5 scales derived from a Gaussian image pyramid starting from 256×256 pixels. The pyramid was calculated by low pass filtering and dyadic downsampling. On the left of Fig. 2 the different feature modalities calculated on the image pyramid are depicted. The lower right half of Fig. 2 shows the bottom-up processing of the different features to obtain conspicuity maps C^{BU} that are combined to form the bottom-up saliency S^{BU}. In the upper right half the top-down processing is shown where for each feature an object-specific excitation (E) and inhibition (I) map is calculated and combined into the conspicuity map $C^{\text{TD}}_{O_j}$. All conspicuity maps are combined into the object-specific top-down saliency map $S^{\text{TD}}_{O_j}$ and a nonlinear operator is applied to cut off negative values. The overall saliency map S^{Total} is calculated by linearly combining the normalized bottom-up S^{BU} and top-down $S^{\text{TD}}_{O_j}$ saliency maps depending on the current task of the ADAS using parameter λ. With increasing λ, the top-down saliency contributes more to the final saliency map, leading to a focus of attention on specific objects. The overall saliency map is passed on to the FoA generation.

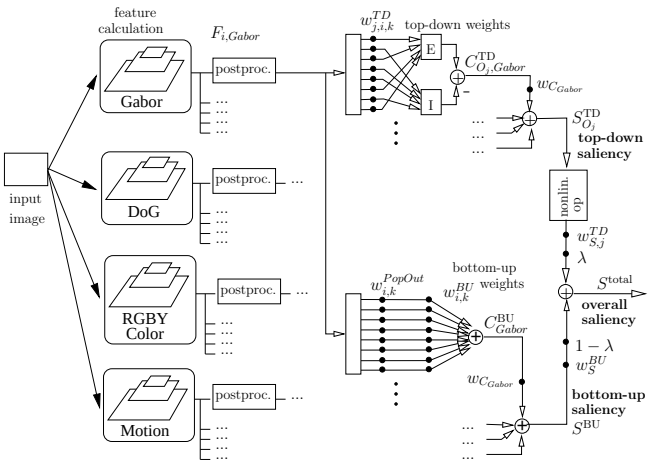

Fig. 2. Simplified sketch of our visual attention sub-system (showing for each feature only one output).

As feature modalities we currently use *odd* and *even Gabor filters* [18] in 4 orientations and *Difference of Gaussians filters* (DoG), both with additional on-center/off-center separation. For example, bright lane markings on a dark road

would trigger an on-center response, while a dark car on a bright background would trigger an off-center response. The separation of a filter in on-center (called on-off in the following) and off-center selectivity (off-on) as emphasized in [19] is realized by separating the filter response into its positive and negative part, which is equivalent to the computationally more demanding usage of two different filter kernels. Additional features are the biologically motivated *RGBY color* space as color opponent and double color opponent [19] and *motion* from differential images.

The feature map responses are passed through a post-processing step that consists of normalizing, squaring, and nonlinear noise suppression by a sigmoidal function. In addition to combining these features to obtain a bottom-up saliency map as it is done typically [20], [21], we also compute top-down saliency maps using object-specific feature map weights. The object-specific weights are inspired by [22], [23] in the way the weights are obtained: During a supervised training stage, the feature map activations of an object are compared to the feature map activations in its surrounding. From this comparison, the relative importance of a feature (its signal-to-noise ratio) can be determined. For each trained object O_j and feature channel $F_{i,k}$ we therefore get a top-down weight $w^{\text{TD}}_{j,i,k}$ that is proportional to how well the feature channel i of feature modality k is able to discriminate the object j from its surrounding. More specifically, the average activation in the object region is related to the average activation in the surround on each feature map $F_{i,k}$ taken only $N_{obj|surr}$ pixels above the threshold $\kappa = K_{conj} Max(F_{i,k})$ with $K_{conj} = (0, 1]$ into account:

$$w^{\text{TD}}_{j,i,k} = \frac{\frac{\sum\limits_{x,y \in obj} (F_{i,k} > \kappa)}{N_{obj}}}{\frac{\sum\limits_{x,y \in surr} (F_{i,k} > \kappa)}{N_{surr}}} \tag{1}$$

In order to emphasize matching features and suppress irrelevant features, separate maps for *excitation* E and *inhibition* I are constructed. Their combination leads to object-specific conspicuity maps $C^{\text{TD}}_{O_j,k}$:

$$E^{\text{TD}}_{O_j,k} = \sum_{i=1}^{N_k} w^{\text{TD}}_{j,i,k} F_{i,k} \qquad \forall\, w^{\text{TD}}_{j,i,k} \geq 1.0 \tag{2}$$

$$I^{\text{TD}}_{O_j,k} = \sum_{i=1}^{N_k} \frac{1}{w^{\text{TD}}_{j,i,k}} F_{i,k} \qquad \forall\, w^{\text{TD}}_{j,i,k} < 1.0 \tag{3}$$

$$C^{\text{TD}}_{O_j,k} = E^{\text{TD}}_{O_j,k} - I^{\text{TD}}_{O_j,k} \tag{4}$$

It is important to note that the performance gain of this approach compared to standard attention systems lies in the explicit inhibition of non-target regions. The conspicuity maps $C^{\text{TD}}_{O_j,k}$ are combined to an object-specific top-down saliency map $S^{\text{TD}}_{O_j}$ by modality specific weights w_{C_k} (conspicuity weights) that are proportional to the confidence one can assign to the modality k in the current scene. This can be done dynamically depending on, e.g., the current weather or lighting conditions. The TD saliency results from a weighted sum of 8 different conspicuity maps (even Gabor on-off/off-on, odd Gabor on-off/off-on, DoG on-off/off-on, RGBY color opponent, RGBY double color opponent):

$$S^{\text{TD}}_{O_j} = \sum_{k=1}^{8} w_{C_k} C^{\text{TD}}_{O_j,k} \quad \text{with } C^{\text{TD}}_{O_j,k} = \sum_{i=1}^{N_k} w^{\text{TD}}_{i,j,k} F_{i,j,k} \tag{5}$$

In addition, we also calculate a biased bottom-up saliency map by combining all feature maps weighted with their specific bottom-up weights $w^{\text{BU}}_{i,k}$ resulting in the weighted sum of 8 modalities (Gabor and DoG as for TD, RGBY double color opponent, motion):

$$S^{\text{BU}} = \sum_{k=1}^{8} w_{C_k} C^{\text{BU}}_{k} \quad \text{with } C^{\text{BU}}_{k} = \sum_{i=1}^{N_k} w^{\text{PopOut}}_{i,k} w^{\text{BU}}_{i,k} F_{i,k} \tag{6}$$

As $w^{\text{BU}}_{i,k}$ we choose a set of weights that shows good performance for most situations in the car environment. In the object-unspecific bottom-up path no inhibition takes place (i.e., feature maps are only added up), since its purpose is to evaluate the general unspecific saliency of a scene. The individual bottom-up feature maps are additionally preprocessed by a pop-out operator that globally amplifies maps with a small number of maxima and attenuates maps with many maxima [20]. The pop-out operator multiplies the feature maps with a dynamic factor $w^{\text{PopOut}}_{i,k}$ computed at runtime (see Eq. (7)). The factor is inversely proportional to the number of pixels that are near the maximum of the feature map. Additionally, $w^{\text{PopOut}}_{i,k}$ is decreased by a factor of 2 for each higher (i.e., smaller) scale level s in the image pyramid. As higher levels tend to contain more pixels fulfilling the threshold $\xi = 0.9 \cdot Max(F_{i,k})$ in the denominator of Eq. (7), this weight increase maintains the comparability of scales:

$$w^{\text{PopOut}}_{i,k} = \sqrt{\frac{2^s}{\sum\limits_{\forall x,y \text{ with } F_{i,k}(x,y) > \xi} F_{i,k}(x,y)}} \text{ for } s = [0, 4] \tag{7}$$

By applying this operator, the bottom-up path is designed to amplify feature maps that show few maxima, i.e., that are sparse. In consequence, feature maps containing image regions that pop out are boosted. It is of crucial importance that the top-down feature maps do not pass a similar pop-out step, since by tuning the top-down weights, we aim at finding objects based on feature conjunctions. The individual feature map responses for the searched objects might only reach medium values, whereas the combination of all relevant maps leads to a strong response in the resulting saliency map. Through this explicit differentiation we achieve an increased performance compared to other top-down attention systems.

For weighting the feature maps we currently use TD weight sets for signal boards and cars ($w^{\text{TD}}_{sigboard,i,k}$ and $w^{\text{TD}}_{car,i,k}$) that were calculated in a supervised training step using Eq. (1). It is envisioned in later versions of our ADAS to calculate these weights dynamically at runtime to track and even learn new objects.

C. Object recognition

For object recognition we use a view-based approach, where we perform classification only on the image patch provided by the FoA segmentation. Note that object recognition operates on the original image resolution of 800×600 pixels, i.e., the RoI position and size provided by the saliency system are transformed appropriately.

The object recognition module is based on a biologically motivated processing architecture proposed in [24]. It uses a strategy similar to the hierarchical processing in the 'what' pathway of the human visual system by creating a classification hierarchy. Unsupervised learning is used for the lower levels of the hierarchy to determine general features that are suitable for representing arbitrary objects robustly with regard to local invariance transformations like local shift and small rotations. Only at the highest level of the hierarchy object-specific learning is carried out, i.e., only this layer has to be trained for different objects. This architecture can be applied to the difficult case of segmentation-free recognition that we have to deal with as the saliency segmentation only provides an approximate RoI with rectangular shape and no object-specific segmentation.

Training is done by presenting several thousand color RoI images with changing backgrounds for back views of cars and signal boards (see also [25]). The learning algorithm automatically extracts the relevant object structure and neglects the clutter in the surround. The output of the classifier is the identity of the recognized object and a confidence value where a threshold is used to reject object hypotheses with low confidence. The threshold is chosen so that only a small number of false positives can occur for cars, as a wrong car detection could lead to a false emergency braking. If a car is not recognized due to the high threshold, it is stored in the STM as unknown and tracked for N frames before it is removed from the STM. Subsequently, if the car is still a salient object, a new FoA will be generated and recognition is performed again. As now the car may be closer due to the ego-motion of our vehicle, the image patch may be larger and therefore may have a higher confidence resulting in a correct recognition.

D. Depth cues

The current ADAS uses three independent depth sources (see Fig. 3) that are combined using weak fusion (see [26]). Weak fusion combines the depth sources based on the reliability of the specific cues. It is realized here using an Extended Kalman Filter (EKF) that combines at each time step the cues via dynamic weights depending on the static sensor variances (calculated offline) and the available depth sources. Note that not every cue is available in each time step. The EKF uses a second order process model for its prediction step that models the relevant kinematics in the car domain (velocity and acceleration). The resulting depth values are assigned to detected objects in the image. The following depth sources are currently used for fusion in the EKF:

Depth from radar is obtained from a commercial standard vehicle equipment sensor, which delivers sparse pointwise measurements with high confidence (for an example see Fig. 3a).

Depth from bird's eye view is realized using inverse perspective world to image mapping (see [16]) based on a pin hole camera model under the flat plain assumption (i.e., all objects in the image are assumed to have zero height). Inverse perspective mapping (i.e., the inverse usage of the 3D-world position X_W, Y_W, Z_W to (u,v)-image mapping $[u, v] = f(X_W, Y_W, Z_W = 0, \text{camera parameters})$) is used to assure a dense bird's eye view image along with low computational demands. A vertical grow algorithm with dynamic thresholds searches for lanes and obstacles based on discontinuities in the bird's eye view and assigns a distance value to found obstacles (see Fig. 3b).

Depth from object knowledge calculates the distance of an object Z_{obj} (see Eq. (8)) using knowledge about the RoI area the object covers on the camera's image chip (width W_{pixel} and height H_{pixel}), the width and height of the object in the real world drawn from experience (W_{real} and H_{real}) as well as the intrinsic parameters of the camera sensor (α_u = focal length/pixel width and α_v = focal length/pixel height):

$$Z_{obj,W} \approx \frac{W_{real}\ \alpha_u}{W_{pixel}} \quad \text{and} \quad Z_{obj,H} \approx \frac{H_{real}\ \alpha_v}{H_{pixel}} \qquad (8)$$

The intrinsic camera parameters were calculated in a camera calibration procedure. A prerequisite for depth from object knowledge is a reliable segmentation algorithm. Currently we use histogram based segmentation on an image region that is pre-segmented by our region growing algorithm working on the saliency (see Fig. 3c).

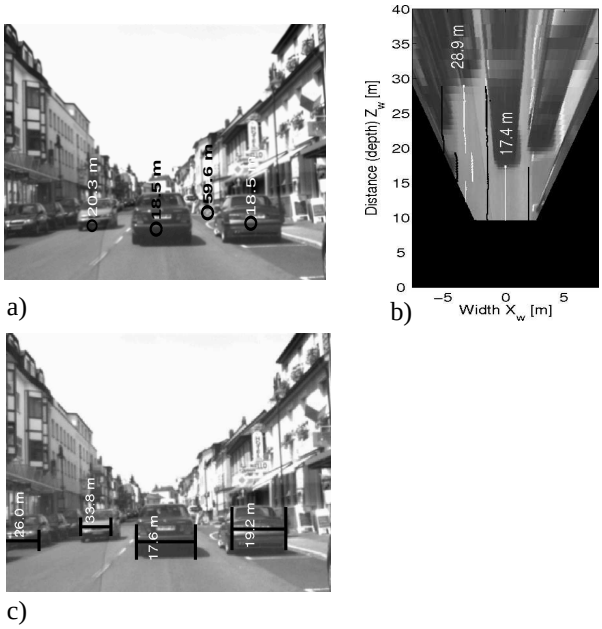

Fig. 3. Used depth cues: Depth from (a) radar, (b) bird's eye view, (c) object knowledge.

IV. EXPERIMENTS AND RESULTS

A. Evaluation of depth fusion

Figure 4 shows the EKF-based fusion of depth measurements for a car that drives in front of our prototype vehicle through an inner city (see Fig. 3a). For the EKF we used the sensor variances $\sigma_{radar} = 0.3$, $\sigma_{birds} = 2.8$, and $\sigma_{obj} = 2.7$ as well as the process variance $\sigma_{process} = 0.023$ for the prediction step. Note that the usage of two additional monocular depth cues of high variance fused with the low variance radar cue ensures the availability of depth values even if the interesting objects are outside of the radar beam.

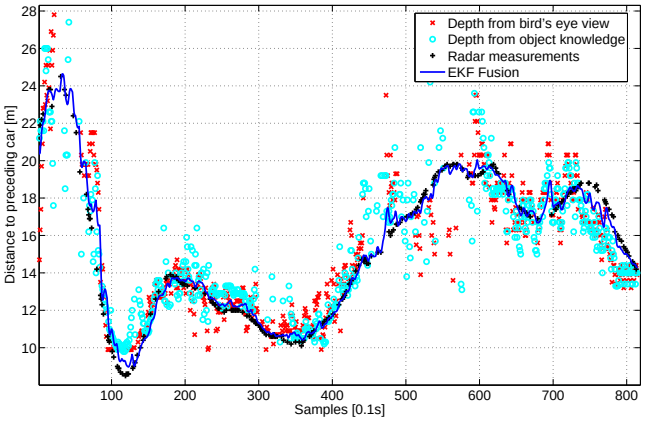

Fig. 4. Depth from bird's eye view, object knowledge, radar and fusion with EKF.

B. Experimental Setup for System Evaluation

Scenario: In order to evaluate the proposed system in a challenging situation, we concentrate on typical construction sites on highways. This situation is quite frequent and a traffic jam ending exactly within a construction site is a highly dangerous situation: due to the S-curve in many construction sites, the driver will notice a braking or stopping car quite late as the signal boards limit the field of view (see Fig. 5a). Our ADAS implementation uses a 3-phase danger handling scheme depending on the distance and relative speed of a recognized obstacle. For example, when the vehicle drives around 40 km/h and a static obstacle is detected in front at less than 33 meters, in the first warning phase a visual and acoustic warning is issued and the brakes are prepared. If the dangerous situation is not resolved by the human driver, the second phase triggers the belt pretentioners and the brakes are engaged with a deceleration of 0.25 g followed by hard braking of 0.6 g in the third phase.

Technical setup: For the experiments we used a Honda Legend prototype car equipped with a mvBlueFox CCD color camera from Matrix Vision delivering images of 800×600 pixels at 10 Hz. The image data as well as the radar and vehicle state data from the CAN bus can be recorded. The recorded data is used during offline evaluation. For online prcessing all data is transmitted via Ethernet to two laptops (2 GHz Core Duo) running our RTBOS integration middleware [27] on top of Linux. The individual RTBOS components are implemented in C using an optimized image processing library based on the Intel IPP [28].

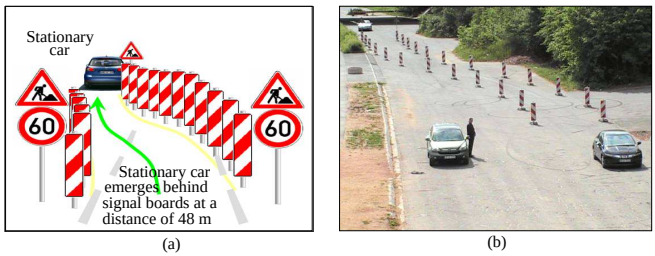

Fig. 5. Scenario: (a) Schematic sketch of the construction site scenario. Stationary car is visible from 48 meters on. (b) Real scenario.

Test data for training and evaluation: In order to gain sufficient training data and for evaluating the actual system performance, we set up an exemplary construction site on a private driving range where we recorded data and performed the actual online tests.

C. Influence of Parameters on Detection Performance

All results described in the following are obtained by averaging over 10 recorded streams in order to lessen statistical outliers. As performance metric we will use the detection performance as this is a good indicator for the efficiency of the saliency system in analyzing complex visual scenes under time constraints. As in each time step of the system running at 10 Hz one FoA is analyzed in the 'what' pathway and potentially added to the STM, we will use frames (equivalent to $\frac{1}{10}$ second) as time unit.

In the first step the object detection distance is evaluated depending on STM size N and the TD parameter λ (setting the amount of TD influence) while using a TD weight set trained on cars. Figure 6 shows the distance to the stationary car when the first FoA hits the car, which is defined by hand-labeled groundtruth on the recorded streams. It can be seen that the larger the TD influence (search task: find cars)

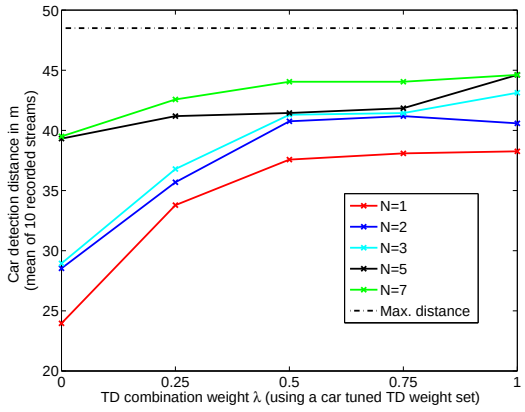

Fig. 6. Stationary car detection distance depending on λ=0, 0.25, 0.5, 0.75, and 1 as well as the STM size N=1,2,3,5, and 7 when using groundtruth for detecting a hit

expressed by λ, the earlier the car is detected. Similarly, the more objects are stored in the STM (object number N), the earlier the car is detected as a large part of the visual scene is already contained as (unknown) objects in the STM and therefore inhibited in the saliency map. It can also be deduced that with growing N the influence of TD is reduced since the scene coverage increases.

Including the task of object recognition in the evaluation, Fig. 7 shows the distance to the stationary car when the first FoA hits the target and this RoI is recognized as car by the object classifier. Since the used classification threshold was set high to obtain a low false-positive error rate at the cost of a high false-negative error rate, the distance when the car is detected is smaller than in the evaluation with groundtruth. Differing from Fig. 6, at large values of N (see Fig. 7 for N=7) the detection distance worsens again. The reason for this effect is that our system is not using object segmentation algorithms but performs segmentation directly on the saliency image which can lead to enlarged patches suppressing the surround of the found objects as well. In this way, the borders of the car might be suppressed by adjacent signal board patches leading to incomplete car FoAs that are not sufficient for correct classification by the used object classificator.

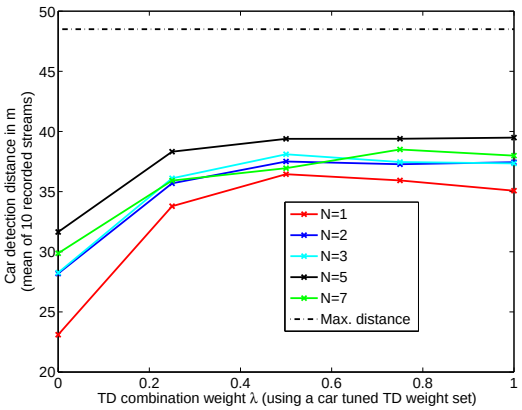

Fig. 7. Stationary car detection distance depending on λ=0, 0.25, 0.5, 0.75, and 1 as well as the STM size N=1,2,3,5, and 7 when using the classifier for detecting a hit.

Based on Fig. 7 the best choice of λ for detecting cars would be 1, which equals pure TD search mode. However, such a parameterization is not appropriate because this leads to a reduced capability of detecting other objects that are only prominent in the BU saliency (see Fig. 8). Here we see that with growing λ the average detection distance of signal boards (the only other object class besides cars in our evaluation) drops. Stated differently, the system ignores all other objects while searching for cars in pure TD mode (λ = 1), which might lead to dangerous situations. The default λ was hence set to 0.5 for the online tests.

In the previous evaluations we assumed that the scene contains more than N objects and used a fixed STM size which is equivalent to storing any object for N frames

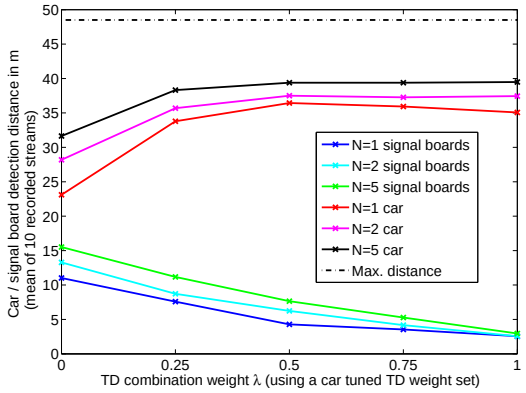

Fig. 8. Detection distance depending on λ=0, 0.25, 0.5, 0.75, and 1. Average detection distance of signal boards and the stationary car using the object classifier for an STM size of N=1,2, and 5.

independent of, e.g., whether it is was correctly recognized. We now introduce an object specific Time To Live (TTL) defining for how many frames an object is stored in the STM before it is removed. In this way, unknown objects can be tracked for only a short time before a new recognition attempt is carried out if the image region is still salient. Figure 9 shows how the choice of the TTL influences the system performance. For an object unspecific TTL of 5 frames the curve is identical to Fig. 8 for N=5. For the object-specific case we chose $\text{TTL}_{sigB} = 6$ frames for signal boards, $\text{TTL}_{cars} = 20$ frames, and $\text{TTL}_{unknown} = 3$ frames, leading for the construction site streams on average to N=5 objects in the STM. Note that the low value of $\text{TTL}_{unknown}$ and the high value of TTL_{cars} are linked to setting the object recognition threshold high, i.e., it is very likely to get an unknown which is a false negative car but rather unlikely to get a car that is a false positive.

A clear gain in detection performance can be seen when using object dependent TTL values which is due to the fact that FoAs which hit the car very early are often too small

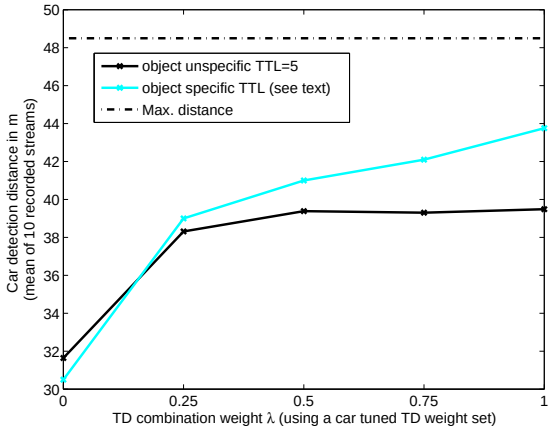

Fig. 9. Stationary car detection distance depending on λ=0, 0.25, 0.5, 0.75, and 1 while using object unspecific and object specific TTL values.

for a reliable classification. These unknown scene parts are suppressed only for 3 frames before the classifier gets a second chance to detect the car. This object-specific TTL parameterization was used during the online tests described below.

D. Evaluation of System Performance

We evaluated the warning generation offline in detail on 10 recorded construction site streams used also for evaluation in the previous Section IV-C. In all streams, the ADAS was able to recognize and track the car from a distance between 42 and 32 meters, while the car was fully visible at a distance of about 48 meters.

During documented online system tests in the setting depicted in Fig. 5 with our prototype vehicle driving 40 km/h our system detected in 57 of 60 cases the stationary car in time and issued the 3 warning phases as expected including autonomous braking. In the remaining cases, either the object recognition detected a signal board as car and the braking was performed too early or the FoA generation did not deliver a good car RoI position so that the fusion of the car RoI with radar data failed and no warning/braking was performed at all. Note that in our vision-based proof-of-concept system we rely completely on vision and do not make use of an additional radar-based emergency braking that would be needed in real traffic as backup for situations in which our vision system fails.

V. SUMMARY

In this contribution we have presented our approach to an Advanced Driver Assistance System where we make use of a human-like attention system for controlling the processing focus. Through tuning the attention system to interesting objects the system analyzes only relevant parts of the scene. By performing an EKF-based fusion of different depth sources, good depth estimates are obtained for all objects on the street. The overall architecture is organized in a brain-like fashion by separating the identification of new objects from the continuous tracking of previously detected objects. In experiments we have shown how the overall detection performance is affected by different strengths of TD-weighting as well as the Time To Live of an object. By applying object specific Time To Live values our system is capable of detecting cars in cluttered scenes even from a great distance. For a live demonstration we chose a construction site setup where the system has successfully performed braking online, providing a proof-of-concept. Ongoing efforts aim at incorporating information about the road into the overall architecture concept in order to provide the basis for more context-dependent processing strategies like, e.g., focusing attention on objects on or near the road.

REFERENCES

[1] K. Kodaka, M. Otabe, Y. Urai, and H. Koike, "Rear-end collision velocity reduction system," in *Proc. 2003 SAE World Congress*, Detroit, 2003.

[2] K. Kodaka and J. Gayko, "Intelligent systems for active and passive safety - collision mitigation brake system," in *Proc. of the ATA_EL conference 2004*, Parma, June 2004.

[3] *DARPA Urban Challenge*. [Online]. Available: http://www.darpa.mil/grandchallenge/

[4] "European commission information society 'Intelligent Car initiative'," 2007. [Online]. Available: http://ec.europa.eu/information_society/activities/intelligentcar/

[5] "European project SAFESPOT," 2007. [Online]. Available: http://www.safespot-eu.org

[6] "European project PReVENT," 2007. [Online]. Available: http://www.prevent-ip.org/

[7] E. Dickmanns, "Three-Stage Visual Perception for Vertebrate-type Dynamic Machine Vision," in *Engineering of Intelligent Systems (EIS)*, Madeira, Feb 2004.

[8] U. Franke, D. Gavrila, A. Gern, S. Görzig, R. Janssen, F. Paetzold, and C. Wöhler, "From door to door - principes and applications of computer vision for driver assistant systems," in *Intelligent Vehicle Technologies*, L. Vlacic, M. Parent, and F. Harashima, Eds. Oxford: Butterworth Heinemann, 2001.

[9] A. Broggi, M. Bertozzi, G. Conte, and A. Fascioli, "Argo prototype vehicle," in *Intelligent Vehicle Technologies*, L. Vlacic, M. Parent, and F. Harashima, Eds. Oxford: Butterworth Heinemann, 2001.

[10] N. Ouerhani, "Visual attention: From bio-inspired modeling to real-time implementation," Ph.D. dissertation, Université de Neuchâtel, Institute de Microtechnique, 2003.

[11] G. Färber, "Biological aspects in technical sensor systems," in *Proc. Advanced Microsystems for Automotive Applications*, Berlin, Mar 2005, pp. 3–22.

[12] C. Stiller, G. Färber, and S. Kammel, "Cooperative cognitive automobiles," in *IEEE Intelligent Vehicles Symposium*, 2007, pp. 215–220.

[13] P. Cavanagh and G. Alvarez, "Tracking multiple targets with multifocal attention," *Trends in Cognitive Sciences*, vol. 9, pp. 350–355, 2005.

[14] R. M. Klein, "Inhibition of return," *Trends in Cognitive Science*, vol. 4, no. 4, pp. 138–145, April 2000.

[15] S. Palmer, *Vision Science: Photons to Phenomenology*. MIT Press, 1999.

[16] M. Bertozzi, A. Broggi, and A. Fascioli, "Stereo inverse perspective mapping: Theory and applications," *Image and Vision Computing*, vol. 8, no. 16, pp. 585–590, 1998.

[17] T. Michalke, J. Fritsch, and C. Goerick, "Enhancing robustness of a saliency-based attention system for driver assistance," in *The 6th Int. Conf. on Computer Vision Systems (ICVS'08)*, Santorini, Greece, 2008.

[18] R. Trapp, "Stereoskopische korrespondenzbestimmung mit impliziter detektion von okklusionen," Ph.D. dissertation, University of Paderborn Germany, 1998.

[19] S. Frintrop, "Vocus: A visual attention system for object detection and goal-directed search," Ph.D. dissertation, University of Bonn Germany, 2006.

[20] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 11, pp. 1254–1259, 1998.

[21] C. Koch and S. Ullman, "Shifts in selective visual attention: towards the underlying neural circuitry," *Human Neurobiology*, vol. 4, no. 4, pp. 219–227, 1985.

[22] S. Frintrop, G. Backer, and E. Rome, "Goal-directed search with a top-down modulated computational attention system," in *Lecture Notes in Computer Science*, 2005, pp. 117–124.

[23] V. Navalpakkam and L. Itti, "Optimal cue selection strategy," in *Advances in Neural Information Processing Systems, Vol. 19*. Cambridge, MA: MIT Press, 2006, pp. 1–8.

[24] H. Wersing and E. Körner, "Learning optimized features for hierarchical models of invariant object recognition," *Neural Computation*, vol. 15, no. 2, pp. 1559–1588, 2003.

[25] A. Gepperth, B. Mersch, J. Fritsch, and C. Goerick, "Color object recognition in real-world scenes," in *ICANN 2007, part II*, ser. Lecture Notes in Computer Science, J. de Sa, Ed. Springer Verlag Berlin Heidelberg New York, 2007, no. 4669.

[26] M. Landy, L. Maloney, E. Johnsten, and M. Young, "Measurement and modeling of depth cue combinations: in defense of weak fusion," *Vision Research*, vol. 35, no. 3, pp. 389–412, 1995.

[27] A. Ceravola, F. Joublin, M. Dunn, J. Eggert, and C. Goerick, "Integrated research and development environment for real-time distributed embodied intelligent systems," in *Proc. Int. Conf. on Robots and Intelligent Systems*, 2006, pp. 1631–1637.

[28] Intel, "Integrated Performance Primitives," 2006, http://www.intel.com/cd/software/products/asmo-na/eng/perflib/ipp/302910.htm.