Eyetracker-Based Driver State and Intention Recognition for Realtime Control and Configuration of Human-Machine Interfaces in Vehicles – 121

T. Poitschke, F. Wallhoff, G. Rigoll
Institute for Human-Machine Communication
Department of Electrical Engineering and Information Technology
Technical University of Munich
Email: {poitschke,wallhoff,rigoll}@tum.de

S. Bardins, E. Schneider, Th. Brandt
Department of Neurology
Faculty of Medicine
Ludwig-Maximilians-University of Munich
Email: {sbardins,eschneider}@nefo.med.uni-muenchen.de
thomas.brandt@med.uni-muenchen.de

Abstract— **In the first funding period this project was focused on a novel gaze-based human-machine interface that was intended to avoid driver overload in automotive uses. A major short-term goal was the extension of an existing head-mounted eyetracker to a low-latency remote system that was to be integrated into an automotive environment. Based on the gaze data and other modalities like speech as well as facial expression, algorithms for automated driver state and intention recognition were envisioned together with a cognitive system that autonomously adapts the information flow between vehicle and driver to the drivers state, thereby avoiding driver overload. Various approaches of early signal and late semantic fusion were planned to be applied in the integration of multimodal user state recognition.**

In this progress report, we will summarize the gathered results concerning various novel approaches for remote eye tracking, implemented (automotive) demonstrators, and cross-cluster cooperations with other CoTeSys projects.

Due to the numerous activities and requests within the cluster in the first project year, the application scenario and project title for the ongoing research activities are redefined. Accordingly, we have started to tailor the eye-tracking system to the needs of the humanoid robot scenario and the observation of human workers in the cognitive factory. In production environments important and safety related instructions and symbols can be projected into a workers primary field of view using sophisticated projection techniques based on the knowledge of the users gaze direction. The long-term vision is the integration of remote gaze recognition also into humanoids. Thus a humanoid robot like Johnnie will be able to perceive and interpret a humans area/focus of interest and therefore, by recognizing look ahead fixations, collisions and other critical situations can be avoided.

I. Motivation

In cars of today drivers are overstrain-prone by an unfiltered parallel information flow from diverse and independent in-car systems (e.g. infotainment, navigation, cellular phone). A reduction of these dangers can only be achieved by driver state and intention recognition and a subsequent context-aware coordination of the information flow. To achieve this goal, an appropriate sensor technology is needed in future vehicles in order to draw conclusions on the driver's current mental state from different objective parameters. The integration of an image-based eye tracking sensor into the driver's working environment is one possible approach. For example, its signal can be used for driver state supervision. The derived data can be applied to future cognitive in-car systems that detect intentions from look-ahead fixations and mental workload as well as vigilance from parameters like saccade distribution, eyelid closure, and pupil size. Further, these data can be used for a human-centered gaze-based human-machine interaction. E.g., adapted to the current line of sight, the area, position, and duration of information presentation will be optimized to avoid driver overload. Further, human gaze can be used as a powerful input and control device for graphical user interfaces, to increase the useful interaction bandwidth across the interface. By the combination of an automotive Head-Up Display (HUD) and an eye tracking system, the possibility of contact-analog information visualization arises. A contact-analog HUD would improve the driver-vehicle interaction by focusing the driver's attention to relevant objects in the environment, by augmenting the visual information with environment-related content, and by displaying relevant information at the current target of gaze. By merging the displayed content with the environment, a lowered transferring effort arises. Since this display requires a calibration to the user's position and gaze, and a real-time adjustment of the HUD content, low-latency eye tracking algorithms are also required.

II. Goals

The main goal of the project was the development of a remote eye tracker that can be used for low-latency real-time gaze-based human-vehicle interaction, e.g., control of a contact-analog HUD, for driver state supervision, and gaze-control of human-machine interfaces.

In a first step, we therefore extended the EyeSeeCam system that has been developed by the Neurology of the LMU, to be used as a remote gaze tracker. In doing so, the newly developed system can now detect gaze in space from the eye-in-head orientation and the 3D positions of the EyeSeeCam goggle markers. This approach quickly enabled us to perform gaze-based experiments in cooperation with project #127 [6]. In addition to this adoption, the main goal is the development

Fig. 1. EyeSeeCam head-mounted mobile eye tracker of the LMU Neurology

of a contact-less remote gaze tracker. Based on this sensor's information about gaze direction it was initially planned to implement an automotive human-machine interface in order to achieve an efficient, ergonomic, and seamless driver integration.

A. Targeted Demonstrators

In the first project phase, our mainly targeted demonstrator was the *MuCAR* (see Fig. 8 on page 7). The plan was to integrate the results of the development of the eye tracker sensor and of a new human-machine interface into the *MuCAR*.

B. Baseline System

In the starting phase of the project we could revert on various preliminary work of the involved research partners. The LMU Neurology contributed with expertise in the area of head-mounted eye tracking, and especially with the eye tracking system EyeSeeCam, which provides both temporally and spatially highly dissolved eye tracking data (see Fig. 1). In a satellite project funded by the Bayerische Forschungsstiftung the development of EyeSeeCam has been finalized to the point that manufacturing of the prototype is now possible. EyeSeeCam can now be replicated upon request.

A (prototypical) remote eye and head (pose) tracking system, developed at the Institute for Human-Machine Communication (MMK), was used in the beginning as a basis for further implementations of capable remote eye and gaze tracking approaches. Also, the MMK has a broad spectrum of preliminary work within the research area of automotive infotainment systems, and provided its research facilities such as the navigation lab and a car mock-up for the integration, testing and evaluation of the eye tracking system and the developed human-machine interfaces.

III. Goal Changes and Redefinition

The first project proposal originally focused on an in-car scenario – especially on the *MuCAR* demonstrator – in order to observe the gaze of a human driver. Due to the numerous activities and requests within the cluster in the first project year the application scenarios for the ongoing research activities are redefined. Since we could identify a high cross-cluster demand for gaze tracking, we decided to concentrate on other application areas within the cluster and its demonstrators. Because of the demand to close the cognitive loop, we decided to cooperate with several cluster projects to achieve this goal. Thus, we had to abandon the automotive scenario and concentrate on future research topics within the cluster, where gaze data can supplement experiments on the investigation of human performance and their cognitive processes in complex assembly tasks in the *Cognitive Factory*. Also, the demonstrator *Johnnie* has a high demand for a true gaze-based human-robot interaction, where we focus on the impact of gaze on natural interaction patterns. The long-term vision now is the integration of the remote eye tracking technology into the active vision systems of the humanoid CoTeSys demonstrators *Johnnie* and *Lola* in order to enable them to detect the direction of human gaze and thereby seamlessly and proactively interact with humans. Also we want to integrate our developed systems in the *Cognitive Factory* demonstrators.

A. Cooperations in the first funding period

- In the cooperation with project #127 *Cognitive Modeling in Ergonomics*, we provided gaze information for the analysis of human gaze behavior during traffic situations and especially for the analysis of the perception of traffic signs [6].
- *ACIPE* (#159): Here, we focused on delivering gaze data for contact-analog AR visualizations for the *Cognitive Factory* demonstrator in RA-F. Also, we will integrate the gathered results from the automotive gaze control demonstrator to the implemented *Cognitive Factory* demonstrator.
- Within the Task Force Nonverbal Communmication (CA-4), gaze has been identified to be an integral aspect throughout a number of projects within the cluster. For this Task force we provide a broad knowledge basis concerning eye/gaze tracking and analysis of eye movements.
- *Cognitive Factory and Humanoid Robots* (#147): The eye tracker is being extended by modules that allow the synchronized integration of external tracking devices (Polhemus, Intersense, etc.). This setup was used and is further planned to be used in projects #147 and #148 and its experiments that examine the fundamentals of eye-hand-coordination in joint human-human and human-machine interaction.
- *Head-mounted Stereo Camera* (#106) this provides direct input to the humanoid demonstrator *Johnnie* in RA-F.

These cooperations already show the strong interdisciplinary character of the project that spans not only faculties but also universities.

B. Planned Cooperations

For the next CoTeSys funding period, the following cooperations and integrations into working groups are planned amongst others, to fulfill the CoTeSys requirements and to strengthen the whole cluster:

- CA-1 Cognitive Vision
- Planned active role in Task Force *Vision for Action* (CA-6). The long-term goal of these activities is to provide an additional „saliency" map with the location of eyes of human co-operators in human-machine interaction scenarios.
- Integration into project #106: Enhancement of demonstrator platform *Johnnie* by remote gaze detector.
- Support of the studies and evaluation concerning real test persons in *ACIPE*.
- Intention recognition studies supplementing the joint human-robot cooperation in *JAHIR* (#207), which fills an identified gap in RA-D, enhancing a natural joint action scenario. An interface to the proposed system architecture in the *Cognitive Facetory* will be established using the *Internet Communication Engine* (ICE).
- *Attention Based Saccade Control* in prospective project group #327 #334 #105
- *Memory-based mechanisms in the cognitive control of uni- and multi-modal attentional selection* (#148). In RA-A, gaze detection will also cover additional neurobiological and -cognitive aspects.

The redefined goals of project #121 will be pursued in the potential new project #323 (*Real-Eye – Remote and Low-Latency Eye Tracking*). The planned cooperations will further deepen the inter-disciplinarity of the project and they will close the *perception-coginition-action* loop..

IV. Results

A. Remote gaze tracking system based on EyeSeeCam

The head-mounted eye tracker EyeSeeCam [4], that was also used as a basis for the head mounted camera (CoTeSys project #106) can already compute the gaze direction in the head coordinate system. A low-latency system is required, that allows to do head free measurements of user gaze in space. Based on the available eye tracker, two gaze tracking systems have been developed that are able to determine gaze in space. These systems have been developed with the *MuCAR* demonstrator and with the cooperation with project #127 in mind.

1) Head tracking with remote stereo cameras: As a first step toward a remote eyetracker 3 LEDs were attached to the eye tracker goggles of Fig. 1. By mounting one LED in the middle of the eye tracker goggles and two laterally, position and orientation of the goggles can be computed.

The LED positions in 3D space are calculated by using a stereo-camera system that runs synchronously to the head-mounted eye tracker cameras. The stereo-camera system consists of two digital cameras (Pointgrey, Firefly MV) operating at 120 frames per second. The camera mount is placed in front of the user (e.g. on a computer monitor). In a subsequent step the stereo cameras will not only track the head position but also the eye positions, thereby rendering the goggles unnecessary.

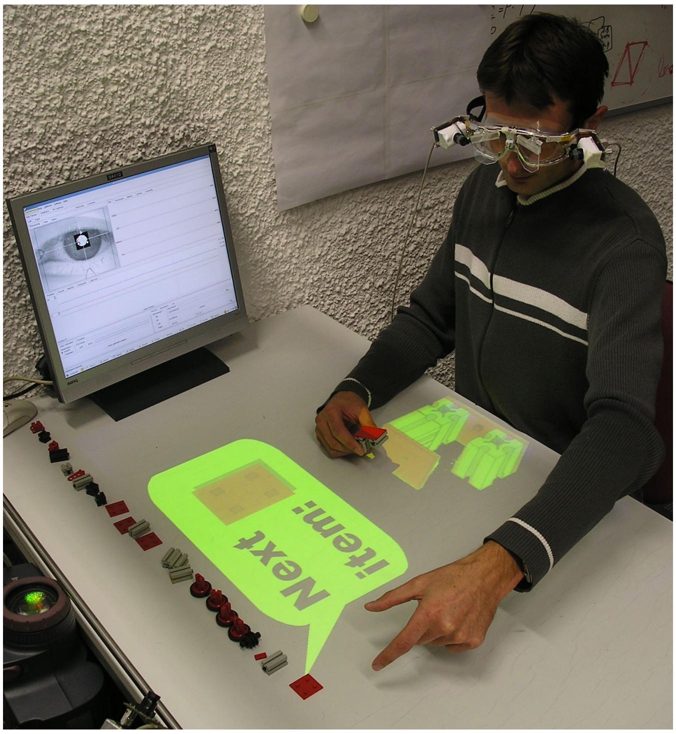

Fig. 2. Remote gaze tracking system in action. The user wears eye tracker goggles and fixates a target on the table. The eye tracker position in 3D space is determined by a stereo camera system. One of the two cameras of the stereo system can be seen in the lower left lower corner. The gaze-in-space data will be further evaluated in the cognitive architecture of project #325

2) Head tracking with a head-mounted scene camera: In a newly established cooperation with project #127 there was a demand to track the head with no additional stereo cameras but with a head-mounted scene camera. We have therefore attached an extra scene camera to the middle of the eye tracker device. This camera is pointing into the same direction as the wearer's face. The wearer is situated in front of a LCD monitor, which has infrared LEDs attached to every corner. By tracking these LEDs with the scene camera, the position of the goggles can be computed from the projection of the rectangle formed by the LED points.

3) Calibration: Calibration-free operation is one of the aims of the project, however, in the current eye tracker versions this is not yet implemented and calibration is still necessary. The calibration of the eye tracker is based on subsequent eye fixations in five defined gaze directions. For this purpose, a small laser was attached on the goggles as close as possible to the subject's eyes [11]. The calibration of the head-tracker was done with the Matlab software *Camera Calibration Toolbox for Matlab* [2], [7]. The stereo camera system needs to be calibrated once at system setup.

4) Hardware and software setup: The image processing algorithms detect the position of the pupil and compute the user's gaze direction measured in the eye tracker coordinate system [4]. In the second step the algorithms detect the position of the three head-LEDs in 3D-space relative to one of the head-tracker cameras. For more accuracy a non-linear

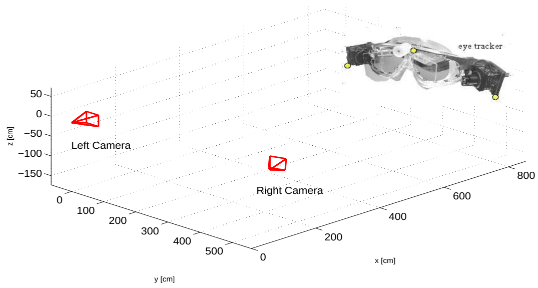

Fig. 3. Head tracker with 3D-coordinates system. The origin of 3D coordinate system is placed in the middle of the left camera. Three LEDs , which are placed on the eye tracker goggles determine the position and orientation of head in 3D space.

lens distortion correction was used [7]. Finally the user's gaze direction in 3D-space was computed.

Numerous interfaces have been implemented that ensure seamless integration with other CoTeSys projects. In a first step the data about the gaze direction in 3D space is delivered to requesting devices by a network socket server and also by a DA converter (USB-DUX). For applications with deterministic latency demands the data is transmitted over the serial port. In a second step it is planed to make the interfaces compatible with the ICE-based cognitive architecture of proposal #325 (*A Conceptual Framework for Cognitive Architectures*).

5) Resolution, accuracy, latency: The direction of gaze-in-head could be determined with an accuracy of $0.71°$ and a resolution of $0.02°$ [4]. It has to be stated that in humans the natural inaccuracy for refixations is on the order of $1°$ [5]. The resolution of the eye tracker goggles in the stereo camera setup was 0.007 mm $0.0032°$ for position and orientation, respectively. In order to determine the precision of the system the user had to fixate four points in space from two different positions. The accuracy of the detection of that four points in 3D space was $0.158°$. Since the stereo camera analysis runs in the same user space the latency of the whole system at 120 Hz is the same as in the head-mounted setup: 10 ms.

B. Remote video-based eye-tracking

In a first step, the remote eye tracking systems have been designed with the head-up display applications in mind. The real-time control of such a display demands for a high-speed and low-latency tracking system. Therefore, the newly designed systems used the same cameras as in the EyeSeeCam system of project #106 with Region Of Interest readout (ROI) for the driver monitoring. Two remote eye-tracking systems have been developed.

1) Eye tracking by conic reconstruction: In the first system the gaze-vector is computed by reconstructing the pupil contours in 3D space, using an algorithm developed in cooperation with CoTeSys project #106 [1].

Since modern eye tracking systems always need some sort of calibration, it would be a big improvement if an eye tracking system could operate free of calibration. One approach to address this problem is reconstructing the pupil in 3D space and assuming the gaze direction parallel to the pupil normal. If the position of the camera rig is known, this approach works without the need for a calibration procedure, because the position and orientation of the pupil ellipse is explicitly known.

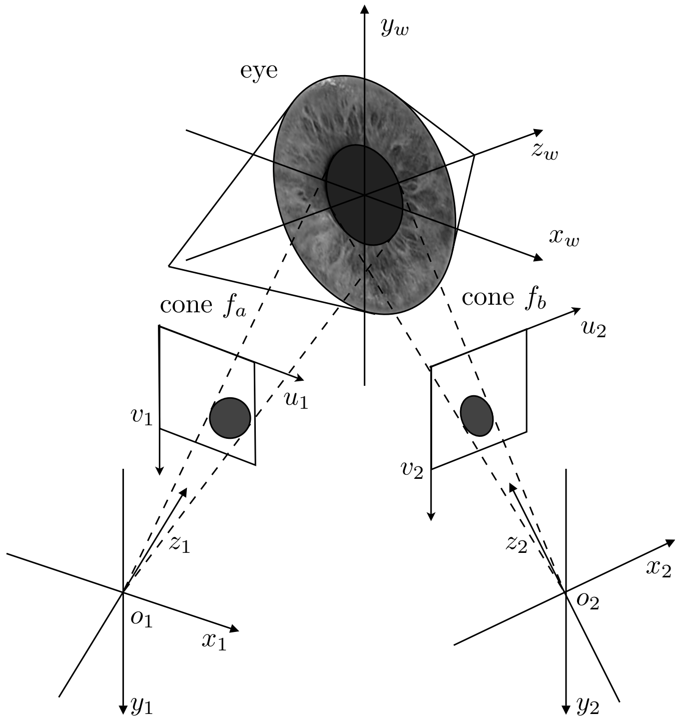

Fig. 4. Coordinate systems and two images of the pupil

a) Stereo reconstruction: The image of a circular pupil in the image plane is an ellipse. By laying a cone f_a through the camera center and the pupil image alone, one cannot determine the 3D position and orientation of the pupil, because many pupils are projected onto the same conic in the image plane. The situation changes if there is another projection of the same pupil on a second image plane. Now we can ray another cone f_b through the second camera center and pupil image. The intersection between the two cones then defines the pupil.

By reconstructing the original pupil ellipse from both projections, we can get position, size and orientation of the pupil in 3D space. A closed form solution to this problem has been proposed previously [3].

Figure 4 shows the image planes, the camera centers (o_1, o_2), the cones (f_a, f_b) and their intersection, and the pupil. The pupil is in the plane defined by the vectors x_w and y_w. The projections of the pupils are in the normalized image planes, defined by the vectors u_1 and v_1 as well as u_2 and v_2. If the original pupil is assumed as an ellipse (or a circle), its projection results in an ellipse again, because projective transformations applied to conic sections always result in conic sections again. The projected ellipses are defined by the two conics below:

$$\mathbf{x}^T\mathbf{A}_1\mathbf{x} = 0 \tag{1}$$

$$\mathbf{x}^T\mathbf{A}_2\mathbf{x} = 0 \tag{2}$$

Given the raw ellipse data, which is the lengths of both the semi-axis a and b, the position in the image plane t_x and t_y as well as the angle ϕ between the x-axis of the image plane and the first main axis of the ellipse, we can obtain the conic matrix by applying an affine transformation $\mathbf{S}$ to a conic $\mathbf{H}$ in normal form as follows:

$$\mathbf{S} = \begin{pmatrix} \cos\phi & -\sin\phi & -t_x\cos\phi + t_y\sin\phi \\ \sin\phi & \cos\phi & -t_x\sin\phi - t_y\cos\phi \\ 0 & 0 & 1 \end{pmatrix} \tag{3}$$

$$\mathbf{H} = \begin{pmatrix} \frac{1}{a^2} & 0 & 0 \\ 0 & \frac{1}{b^2} & 0 \\ 0 & 0 & -1 \end{pmatrix} \tag{4}$$

$$\mathbf{A} = \mathbf{S}^T\mathbf{H}\mathbf{S} \tag{5}$$

The relation between the coordinate system c_i of camera i and the world coordinate system c_w is:

$$\mathbf{x}_i = \mathbf{R}_i\mathbf{x}_w + \mathbf{t}_i \qquad i = 1, 2 \tag{6}$$

For points in the pupil plane $\mathbf{x} = \left(x_w, y_w, 0\right)^T$ this can be written as:

$$\mathbf{x}_i = \mathbf{G}_i\mathbf{u}_w \qquad i = 1, 2 \tag{7}$$

with $\mathbf{u}_w = \begin{pmatrix} x_w & y_w & 1 \end{pmatrix}^T$ being homogenous coordinates on the pupil plane and $\mathbf{G}_i$ being a 3×3 matrix consisting of the first two columns of $\mathbf{R}_i$ and the translation vector $\mathbf{t}_i$.

$$\mathbf{G}_i = \begin{pmatrix} \mathbf{r}_{i1} & \mathbf{r}_{i2} & \mathbf{t}_i \end{pmatrix} \tag{8}$$

With $u_i = \frac{x_i}{z_i}$ and $v_i = \frac{y_i}{z_i}$ follows:

$$z_i\mathbf{u}_i = \mathbf{G}_i\mathbf{u}_w \qquad i = 1, 2. \tag{9}$$

An ellipse in the pupil plane is defined by

$$\mathbf{Q} = \begin{pmatrix} \frac{1}{a^2} & 0 & 0 \\ 0 & \frac{1}{b^2} & 0 \\ 0 & 0 & -1 \end{pmatrix} \tag{10}$$

$$\text{and} \tag{11}$$

$$\mathbf{u}_w^T\mathbf{Q}\mathbf{u}_w = 0 \tag{12}$$

and its projections by

$$\mathbf{u}_i^T\mathbf{A}_i\mathbf{u}_i = 0 \qquad i = 1, 2 \tag{13}$$

Inserting (9) in (13) yields:

$$\mathbf{u}_w^T\mathbf{G}_i^T\mathbf{A}_i\mathbf{G}_i\mathbf{u}_w = 0 \qquad i = 1, 2 \tag{14}$$

As equation (12) und (14) define the same conic $\mathbf{Q}$ we can write:

$$\mathbf{G}_i^T\mathbf{A}_i\mathbf{G}_i = k_i\mathbf{Q} \tag{15}$$

Thereby k_1 and k_2 are unknown scale factors, because $\mathbf{x}^T\mathbf{A}\mathbf{x} = 0$ and $\mathbf{x}^T k\mathbf{A}\mathbf{x} = 0$ describe the same conic.

The basic constraint is then given by:

$$\mathbf{G}_1^T\mathbf{A}_1\mathbf{G}_1 = k_1\mathbf{Q} \tag{16}$$

$$\mathbf{G}_2^T\mathbf{A}_2\mathbf{G}_2 = k_2\mathbf{Q} \tag{17}$$

with

$$\mathbf{G}_1 = \begin{pmatrix} \mathbf{r}_{11} & \mathbf{r}_{12} & \mathbf{t}_1 \end{pmatrix} \tag{18}$$

$$\mathbf{G}_2 = \begin{pmatrix} \mathbf{r}_{21} & \mathbf{r}_{22} & \mathbf{t}_2 \end{pmatrix} \tag{19}$$

Furthermore the relation between the coordinate systems c_1 and c_2 is known by calibration:

$$\mathbf{R}_2 = \mathbf{R}\mathbf{R}_1 \tag{20}$$

$$\mathbf{t}_2 = \mathbf{R}\mathbf{t}_1 + \mathbf{t} \tag{21}$$

The equations (16) and (17) provide 12 constraints, because they consist of two real symmetric 3×3 matrices with 6 parameters each.

As we only have 10 independent unknowns (three each in $\mathbf{R}_1$ and $\mathbf{t}_1$ as well as k_1, k_2, a and b) the system is overdetermined.

It has been shown that $\mathbf{R}_1$ and $\mathbf{t}_1$ can be solved for independently [3]. With $\mathbf{X}^{2\times2}$ being the upper left submatrix of $\mathbf{X}$ we can write:

$$(\mathbf{R}_1^T\mathbf{A}_1\mathbf{R}_1)^{2\times2} = k_1\mathbf{Q}^{2\times2} \tag{22}$$

$$(\mathbf{R}_2^T\mathbf{A}_2\mathbf{R}_2)^{2\times2} = k_2\mathbf{Q}^{2\times2} \tag{23}$$

Substituting equation (20) into equation (23) we can write after eliminating $\mathbf{Q}^{2\times2}$:

$$[\mathbf{R}_1^T(\mathbf{A}_1 - k\mathbf{R}^T\mathbf{A}_2\mathbf{R})\mathbf{R}_1]^{2\times2} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \text{ with } k = \frac{k_1}{k_2} \tag{24}$$

If the 2×2 upper left submatrix of a 3×3 matrix is equal to a zero matrix, this means that its determinant is equal to zero. Therefore (24) gives:

$$\det(\mathbf{A}_1 - k\mathbf{R}^T\mathbf{A}_2\mathbf{R}) = 0 \tag{25}$$

which is equivalent to

$$\det\left[(\mathbf{R}^T\mathbf{A}_2\mathbf{R})^{-1}\mathbf{A}_1 - k\mathbf{E}\right] = 0. \tag{26}$$

This means that k is the eigenvalue of the matrix $(\mathbf{R}^T\mathbf{A}_2\mathbf{R})^{-1}\mathbf{A}_1$. By calculating k and denoting $\mathbf{C} = \mathbf{A}_1 - k\mathbf{R}^T\mathbf{A}_2\mathbf{R}$, equation (24) can be written as:

$$(\mathbf{R}_1^T\mathbf{C}\mathbf{R}_1)^{2\times2} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \tag{27}$$

Equation (27) only provides two independent equations because the matrices are 2×2 symmetric and $\det\mathbf{C} = 0$ has already been used for the solution of k.

Now $\mathbf{R}_1$ can be solved for: One of the eigenvalues of the matrix $\mathbf{C}$ is zero, because $\det(\mathbf{C}) = 0$. Given the two non-zero eigenvalues λ_1 and λ_2 with the corresponding eigenvalues $\mathbf{s}_1$ and $\mathbf{s}_2$ the third column of $\mathbf{R}_1$ can be calculated by:

$$\mathbf{r}_{13} = \pm\text{norm}\left(\sqrt{|\lambda_1|}\mathbf{s}_1 \pm \sqrt{|\lambda_2|}\mathbf{s}_2\right) \tag{28}$$

Here $\text{norm}(\mathbf{x})$ means that $\mathbf{x}$ is normalized to length one. The above case differentiations result in four different possible solutions for $\mathbf{r}_{13}$.

This is due to the fact that two intersecting cones have two shared ellipses. As can be seen from from Fig. 5 ambiguities can be resolved by considering two cases: In the first case (Fig. 5 $\{1\}$) both cameras are on the same side of the ellipse and in the second case (Fig. 5 $\{2\}$) one camera is on the front side and the other camera is on the back side of the

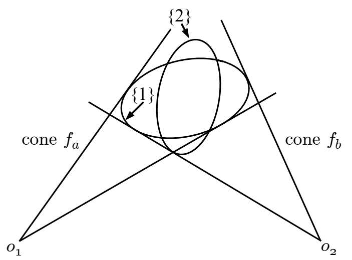

Fig. 5. Intersections of two cones

ellipse. As there is no way to watch a pupil from both sides and we generally want the gaze-vector pointing away from the eyeball, we can rule out three of the solutions by ensuring that the z-components of both $\mathbf{r}_{13}$ and $\mathbf{r}_{23} = \mathbf{R}\mathbf{r}_{13}$ are positive. After having picked the correct $\mathbf{r}_{13}$ the remaining columns of $\mathbf{R}_1$ can be obtained by calculating the eigenvectors $\mathbf{r}_{11}$ and $\mathbf{r}_{12}$ of $\mathbf{H} = \mathbf{r}_{13}^T\mathbf{r}_{13}\mathbf{A}_1$.

In the last step, all remaining parameters can be solved:

$$\mathbf{R}_2 = \mathbf{R}\mathbf{R}_1 \tag{29}$$

$$\mathbf{t}_1 = \begin{pmatrix} \mathbf{r}_{11}^T\mathbf{A}_1^T \\ \mathbf{r}_{12}^T\mathbf{A}_1^T \\ \mathbf{r}_{21}^T\mathbf{A}_2^T\mathbf{R} \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ 0 \\ -\mathbf{r}_{21}^T\mathbf{A}_2^T\mathbf{t} \end{pmatrix} \tag{30}$$

$$k_1 = -\mathbf{t}_1^T\mathbf{A}_1\mathbf{t}_1 \tag{31}$$

$$k_2 = \frac{k}{k_1} \tag{32}$$

$$a^2 = \frac{k_1}{\mathbf{r}_{11}^T\mathbf{A}_1\mathbf{r}_{11}} \tag{33}$$

$$b^2 = \frac{k_1}{\mathbf{r}_{12}^T\mathbf{A}_1\mathbf{r}_{12}} \tag{34}$$

b) Calibration: One prerequisite is, that the position and orientation of the second camera with respect to the first camera must be known. To get this information, the Camera calibration toolbox for MATLAB has been used. The calibration process involves printing a checkerboard pattern, and taking several pictures of the pattern with both cameras from different angles and distances. An iterative algorithm based on [12] is then used to calculate the intrinsic and extrinsic parameters of both cameras as well as the extrinsic parameters of the stereo rig.

c) Hardware and software setup: The system uses calibrated stereo cameras that observe a subject's face sitting about 100 cm away. The scene is illuminated by two infrared LEDs positioned near the optical axis of each camera. This produces the so called red-eye effect, where incoming light is reflected by the blood-rich retina in the direction of the light source which is near the camera. The monochrome cameras have a high sensitivity in the near IR spectrum, so the pupils appear as bright, white spots, just like the red-eye effect in amateur photography.

First, a face tracking algorithm is applied to both input images to find the approximate eye positions. In the area around the assumed eye position a simple thresholding algorithm is applied to roughly find the pupil centers. From there on, the edge points of the pupil are extracted by segmenting the pupil and integrating over the intensity until there is a drop when the darker iris is reached. An ellipse fitting algorithm is applied to the resulting edge points, to get the ellipse parameters of the projected pupil. Figure 6 shows a sample output of the image processing algorithm.

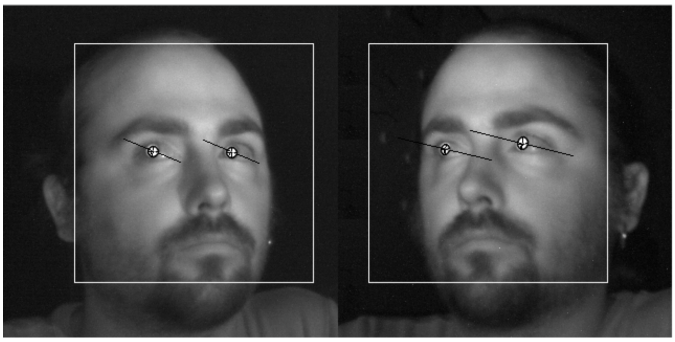

Fig. 6. Image processing output

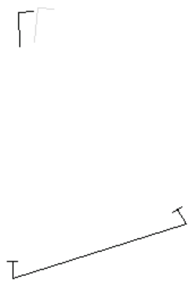

Fig. 7. Scene view from above

Using this ellipse reconstruction algorithm, the pose of the original pupil surface is then computed in a closed mathematical framework [1], [9] in 3D space relative to the camera coordinate systems. For visualization, a view on the scene from above is rendered as depicted in figure 7.

2) Discussion: The stereo reconstruction algorithm has proven useful for the implementation of a novel eye tracker with unprecedented functionality. Based on these results, eye trackers can be designed without the need for a lengthy fixation-based calibration procedure. The algorithm's closed-form solution guarantees real-time performance even with high sampling rates. One drawback, however, is that the ellipse parameters of the conics reconstruction algorithm are not well defined for gaze orientations around the primary position. In the current setup the resolution is 4° (RMS) which is due to the small pupil projections that lead to weakly defined ellipse parameters. This leads to considerable gaze vector variabilities that are not sufficient for proper eye tracking. Future work will address this problem by improving the image processing used to extract the pupil contours. Furthermore, methods of sensor fusion will be investigated, i.e., the ability to operate free of calibration will be combined with existing, but less variable tracking algorithms. In the next step the eye tracking will be integrated into the active vision system of project #106 [1]. There, pivotable cameras with longer focal length will increase the projected pupil sizes. The active cameras will be guided by a wide angle face detection system.

3) Eye tracking by Purkinje reflexes: Here, only the range around the two eyes is read out from the sensor and thus the frame rate is increased from 60 fps (complete picture) to 200 fps (ROI). On the average, one can count on approximately 40 frames per eyelid closure. Thus, it is also possible to differentiate between different eyelid closure speeds as well as fixation phases and saccades of the pupil. Furthermore it is also possible to differentiate reliably between fixation phases and saccades of the pupil. These data can supply a reliable indications for the driver's vigilance.

Since we wanted to design the system as exactly as possible, we decided to use a lens with 40 mm of focal length and we mounted the camera system onto the front column of the car. This position nearly under the vehicle roof guarantees the lowest possible distance to the driver's eyes and thus the highest possible local accuracy with fixed sensor and optics. Further, the system is more robust, since the view is not impaired by steering movements of the driver, contrary to a position behind or beside the steering wheel. Also, reflections from the LED sources on eyeglass lenses rarely occur using this camera perspective.

The pupil is illuminated frame-by-frame by 2 different, alternating LED sources. One is close of the optical axis and causes the red-eye effect, when the other source far from the optical leaves dark pupils. Thus, the corresponding difference image only shows two bright spots at the location of the pupils and is invariant from the environment light. For the suppression of artifacts and noise, a filter chain consisting of a temporal low-pass, a spatial Gauss filter and various morphologic filters is applied. Finally, the pupils are extracted from the optimized difference image with simple heuristic methods.

After the system start and after losing the tracked pupils in a frame, the pupil search is initiated using the above mentioned techniques. This guarantees, that the pupils are recognized under most diverse environment lighting conditions. A window (ROI) around the two detected pupils is immediately read out of the sensor. The rough position of the pupil is determined using a dynamic threshold value and a following Canny edge detector determines its border. Finally, an ellipse fitting algorithm computes the coordinates of that ellipse, whose pixels have the lowest possible square error to the given edge pixels. These operations are applied on the interpolated picture, whereby sub pixel accuracy can be achieved.

While the frames of the full image search are differently lit up (alternating bright/darkness = on-axis/off-axis), the lighting for the standard "ROI tracking" is kept constantly to the highest possible tracking/frame rate of 200 Hz.

Apart from the pupil position the eye tracker also delivers the coordinates of two LED reflections on the cornea, which will be used in further work packages for the computation of the eye gaze. Here we will use an approach according to [8], which will be modified and extended.

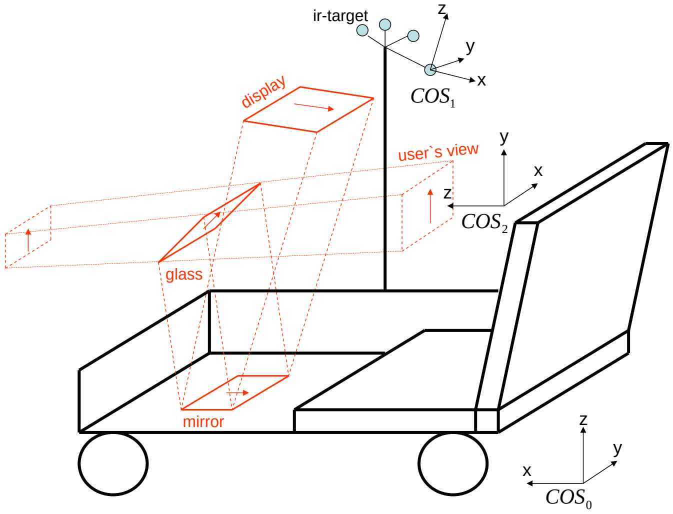

Fig. 8. *MuCAR* schematic overview

C. Implemented automotive demonstrators

In accordance with last year's original proposal we initially have pursued the goal of combining the remote eye tracker technology with a contact-analog head-up display in the context of automotive demonstrators. In the course of the project it became clear that such a driver assistance system does not integrate well with the CoTeSys demands for autonomous driving, although its functionality is based on cognitive (driver) modelling. We therefore changed the project goals (see section III) into the direction of gaze-based interaction facilities for the cognitive factory and humanoid demonstrators. In the subsequent sections we nevertheless report on the initial project activities in the automotive domain.

1) Calibration of the contact-analog HUD: A laboratory prototype was set up that consists of three main parts (see Fig. 8). An Augmented Reality system (ARS) is mounted on a simulation environment. This model consists of a driver‘s seat and a rudimentary steering wheel. The model‘s position and orientation (COS_1) are tracked by an infrared-tracking system (IRTS) based on the IRTS‘s world coordinate system (COS_0). A monitor mounted on top of the car‘s model displays the virtual content, which is mirrored by a reflector at the bottom of the model. Content displayed by the monitor becomes visible within a glass in front of the driver. Using this combiner (e.g., a windscreen) empowers the combination of virtual content and real environment. To overlay virtual content in a contact-analog way with the real environment, the user‘s direction of sight is tracked by our eye tracking system (COS_2, see section IV-A). Based on this, virtual content is displayed by the ARS (see Fig. 9). Moving the user‘s view results in moving virtual content. Thus, a contact-analog perception of the displayed content is induced.

However, the technical complexity for this kind of visualization is much higher than with a non-contact-analog HUD. On

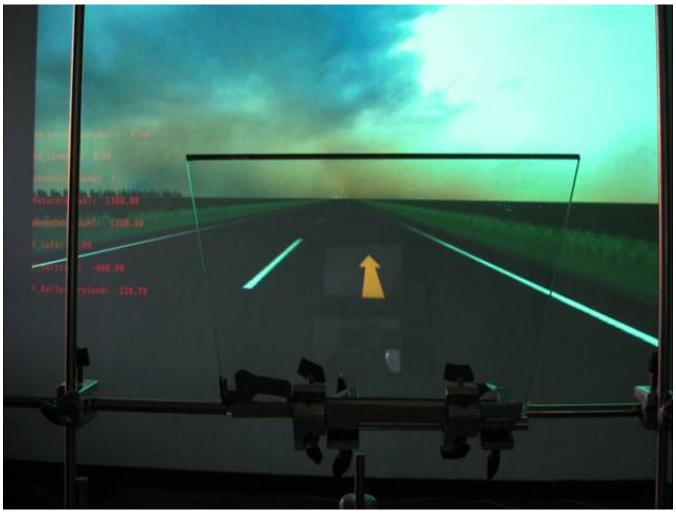

Fig. 9. Driver‘s view through the windscreen/HUD.

the one hand, an adjustment of the synchronization between the displayed HUD content and the real environment must constantly take place. On the other hand, the perspective representation of the HUD contents must be adapted to the optical system consisting of the driver's eyes as well as the actual display, in order to be noticed at the correct place. Therefore, this optical system must be calibrated very exactly. The correct representation of a virtual object on the HUD depends on the positions of the driver (and/or its eyes), the display and the real object relative to each other.

For the calibration procedure, we adapted an approach from the research field of Augmented Reality (AR) to the requirements of the automotive HUD, which is used for the calibration of optical see-through head-mounted displays. This approach uses the point correspondences between real points (in our case represented by a body for the IRTS) and its representative seen through the HUD. During the calibration procedure, both the real 3D-coordinates as well as the corresponding 2D-display coordinates of the respective points are logged. The 2D-coordinates are obtained by clicking the appropriate position in the HUD. These results from the central projection of the 3D-scene on the 2D-display plane. Afterwards, the corresponding projection matrix P can be derived from these values using several coordinate transformations and appropriate numeric operations. P contains the intrinsic and extrinsic parameters of the optical system (e.g., position of the virtual camera center, focal length, etc.).

During the calibration procedure, the user is to keep its head and/or the eye position and thus the center of the virtual camera system as constant as possible. Further, this was supported by a special mounting plate at the head restraint. The head/eye position is noted and evaluated during the entire calibration procedure using the head mounted eye tracking system. Also, these values are used as reference values in the further process. If the user changes now his head and thus his eye position, the new position values are used as further offset on the determined calibration values. Thus, the display system can be adapted on the new parameters of the virtual camera system.

Based on these values, the used AR software determines the position of the virtual HUD contents in the rendered picture. This is displayed on the monitor on the top of the mock-up and reflected into the windshield, where the user notices it as HUD contents.

2) Gaze control of automotive user interfaces: The eye tracking system is used for the registration of the driver's view on several displaying areas and regions of interest (e.g., different control elements) within the car. For a first study, we implemented simply structured dialogs for the instrument cluster. Here, we present a GUI for selection tasks concerning simple "‘YES/NO"’ decisions, e.g., accepting an incoming phone call or accepting a newly computed navigation guidance in case of traffic jams. Currently we are working on a user study aiming a proof-of-concept, and confirming the power of gaze as interaction device. On the basis of the derived data, we will expand the functional range of the system (e.g., implementation of more complex menu structures), and integrate the system into other demonstration scenarios like the *ACIPE Cognitive Factory* demonstrator.

Also, we determine the currently focused displaying area (e.g., instrument cluster, central information display, etc.) for a spatially targeted information output. Thus, we are able to target the graphical dialog to the display next to the user's field of view.

3) Driver intention recognition based on analysis of the driver's eye movements and manual interaction patterns: The experimental setup consists of our vehicle mock-up combined with the head-mounted eye tracking system. Currently we are working on the analysis of eye movement patterns during the manual interaction with hardware control elements (e.g., operating elements for the air condition control) and a touchscreen. Here we want to identify characteristic gaze movement patterns that allow a prediction of what the user wants to do next. Also, we want to integrate an appropriate hand/finger tracking which allows the evaluation of the user's hand trajectory. Since the user has to move his hand to reach a needed operating element (e.g., a button or a thumb wheel), we can also find indications for its intentions. Therefore, we are working on the integration of both the magnetic Polhemus hand tracking system and the hand gesture tracking system proposed in [10].

V. Outlook

Gaze direction is a relevant correlate of the direction of spatial attention and intention. The first project proposal originally focused on an in-car scenario observing the gaze of a human driver. We will now abandon this goal in favour of a new vision: The integration of remote gaze recognition into demonstrators like Johnnie (see Fig. 10) and the cognitive factory. This will equip the demonstrators with the capability to autonomously draw conclusions about the users activities,

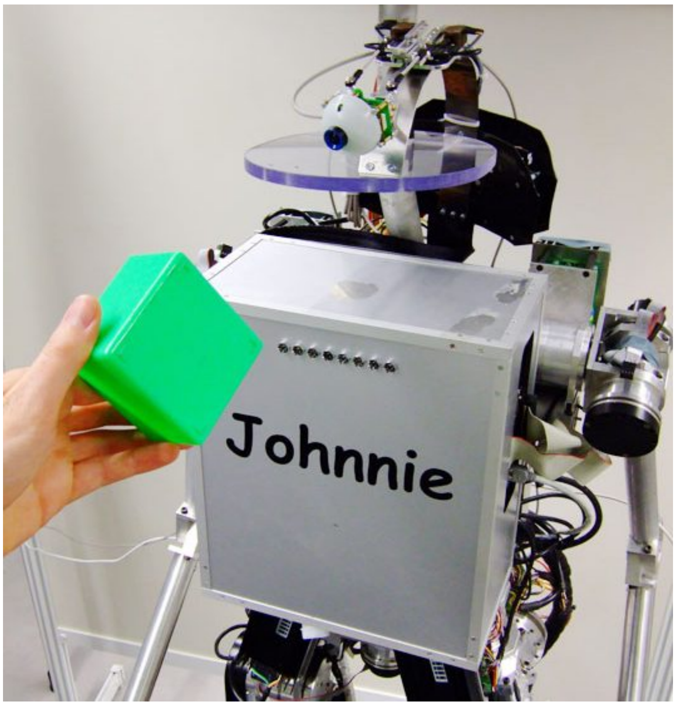

Fig. 10. Humanoid (JOHNNIE) with a mockup of the active vision system of project #106. In this example, JOHNNIE's "eye" pursues a green box. It is planned to equip the active vision system with eye tracking abilities that enable robots to pursue the eyes of cooperating workers and to infer their gaze direction in human-machine interaction scenarios.

states, and intentions, which is mandatory for seamless human-robot interaction and non-verbal communication scenarios. In addition to the pure information about eye-position and gaze, an understanding of the underlying cognitive process, respectively the intentions, is envisioned.

Therefore, the second major project goal will be the investigation of specific methods and algorithms for automated intention derivation and gaze-control. The idea is to distinguish between task-relevant and task-irrelevant gaze directions, e.g., by a cluster-based and segmentation-based analysis of the eye scan path. Based on precise and contact-less measured data containing relevant eye parameters, i.e. eyelid closure, iris and pupil size, and gaze direction, a humans vigilance and mental workload will be identified, which are fundamental for a closed cognition loop. Together with gathered knowledge about the cognitive mechanisms, intentions predicting the subsequent human activity will be calculated from the trajectory of eye-movements and fixations.

VI. Involved Researchers

The following researchers contributed to the work fulfilled in the first funding period:

- S. Kohlbecher (Diploma Thesis): *Blickrichtungserkennung durch Stereorekonstruktion der Pupillenkonturen*
- E. Bay (Bachelor Thesis): *Remote high resolution high speed eye detection*
- Yu Ying (Diploma Thesis): *Eye-Tracking and Calibration for Perspective HUD Projections*
- Bin Shen (Master Thesis): *Remote Infrared-based stereo head and pose tracking*
- T. Hook (Interdisciplinary Project): *Kalibrierung eines kontaktanalogen HUDs*
- S. Bardins: phd candidate
- T. Poitschke: phd candidate
- J. Vockeroth: phd candidate
- E. Schneider: senior researcher
- F. Wallhoff: senior researcher

A. Publications

The following publications have been generated by the involved researchers during the first funding period:

1) M. Ablassmeier, T. Poitschke, F. Wallhoff, K. Bengler, and G. Rigoll. *Eye gaze studies comparing head-up and head-down displays in vehicles.* In Proc. of ICME 2007, Beijing, China, July 2-5, 2007, pp. 2250-2252. CD ROM., 2007.
2) M. Ablassmeier, T. Poitschke, S. Reifinger, and G. Rigoll. *Context-Aware Information Agents for the Automotive Domain Using Bayesian Networks.* In Proc. of 12th International Conference on Human-Computer Interaction HCI Interantional 2007, 22.-27.07.2007, Beijing, P.R. China. CD-ROM., 2007.
3) W. Einhäuser, F. Schumann, S. Bardins, K. Bartl, G. Böning, E. Schneider, P. König, *Human eye-head co-ordination in natural exploration* Network: Computation in Neural Systems, 2007, in press.
4) Marc Halbrügge, B. Deml, B. A. Färber, and S. Bardins. Di*e Erweiterung von ACT-R um Bildverarbeitungsalgorithmen erlaubt die schnelle Erzeugung mächtiger Benutzermodelle.* In 49. Fachausschusssitzung DGLR T5.4 Anthropotechnik - Stand und Perspektiven der simulationsgestuetzen Systemgestaltung. DGLR, 2007.
5) S. Kohlbecher, T. Poitschke, M. Ablassmeier, G. Rigoll, S. Bardins, and E. Schneider. *Gaze-vector detection by stereo reconstruction of the pupil contours.* Abstract in Journal of Eye Movement Research, 1, 2007.
6) T. Poitschke, M. Ablassmeier, S. Reifinger, and G. Rigoll. *A multifunctional VR-simulator platform for the evaluation of automotive user interfaces.* In Proc. of 12th International Conference on Human-Computer Interaction HCI Interantional 2007, 22.-27.07.2007, Beijing, P.R. China. CD-ROM., 2007.
7) S. Reifinger, F. Wallhoff, M. Ablassmeier, T. Poitschke, and G. Rigoll. *Static and dynamic hand-gesture recognition for augmented reality applications.* In Proc. of 12th International Conference on Human-Computer Interaction HCI Interantional 2007, 22.-27.07.2007, Beijing, P.R. China. CD-ROM., 2007.
8) J. Vockeroth, S. Bardins, K. Bartl, T. Dera, and E. Schneider, The combination of a mobile gaze-driven and a head-mounted camera in a hybrid perspective setup. In Proceedings of the International IEEE Conference on Systems, Man and Cybernetics (SMC2007), Montreal, 2007, in press.

References

[1] Cotesys project #106: Action recognition from look-ahead fixations of objects in space with self-calibrating gaze control of head-mounted stereo cameras.

[2] Camera calibration toolbox for matlab. `www.vision.caltech.edu/bouguet/calib_doc`, 09 2007.

[3] S. De Ma. Conics-based stereo, motion estimation, and pose determination. *International Journal of Computer Vision*, 10(1):7–25, 1993.

[4] T. Dera, G. Boening, S. Bardins, E. Schneider, and T. Brandt. Low-latency video tracking of horizontal, vertical, and torsional eye movements as a basis for 3dof realtime motion control of a head-mounted camera. In *Proceedings of the IEEE Conference on Systems, Man and Cybernetics (SMC2006), Taipei, Taiwan*, 2006.

[5] Thomas Eggert. Eye movement recordings: methods. *Dev Ophthalmol*, pages 15–34, 2007.

[6] Marc Halbrügge, Barbara Deml, B. A. Färber, and S. Bardins. ACT-CV – die Erweiterung von ACT-R um Bildverarbeitungsalgorithmen erlaubt die schnelleErzeugung mächtigerBenutzermodelle. In *49. Fachausschusssitzung DGLR T5.4 Anthropotechnik – Stand und Perspektiven der simulationsgestützen Systemgestaltung*. DGLR, 2007.

[7] J. Heikkila and O. Silven. A four-step camera calibration procedure with implicit imagecorrection. *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 1106–1112, 1997.

[8] C. Hennessey, B. Noureddin, and P. Lawrence. A single camera eye-gaze tracking system with free head motion. In *Eye Tracking Research & Applications 2006, pp. 87-94.*, 2006.

[9] S. Kohlbecher, T. Poitschke, M. Ablaßmeier, G. Rigoll, S. Bardins, and E. Schneider. Gaze-vector detection by stereo reconstruction of the pupil contours. *Abstract in Journal of Eye Movement Research*, 1, 2007.

[10] Stefan Reifinger, Frank Wallhoff, Markus Ablassmeier, Tony Poitschke, and Gerhard Rigoll. Static and dynamic hand-gesture recognition for augmented reality applications. In *Proc. of 12th International Conference on Human-Computer Interaction HCI Interantional 2007, 22.-27.07.2007, Beijing, P.R. China. CD-ROM.*, 2007.

[11] E. Schneider, K. Bartl, T. Dera, G. Boening, and T. Brandt. Gaze-aligned head-mounted camera with pan, tilt and roll motion control for medical documentation and teaching applications. In *Proceedings of the IEEE Conference on Systems, Man and Cybernetics (SMC2006), Taipei, Taiwan*, 2006.

[12] Z. Zhang. A flexible new technique for camera calibration. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(11):1330–1334, 2000.