EyeSee3D 2.0: Model-based Real-time Analysis of Mobile Eye-Tracking in Static and Dynamic Three-Dimensional Scenes

Thies Pfeiffer*, Patrick Renner†, Nadine Pfeiffer-Leßmann‡
Center of Excellence Cognitive Interaction Technology, Bielefeld University, Bielefeld, Germany

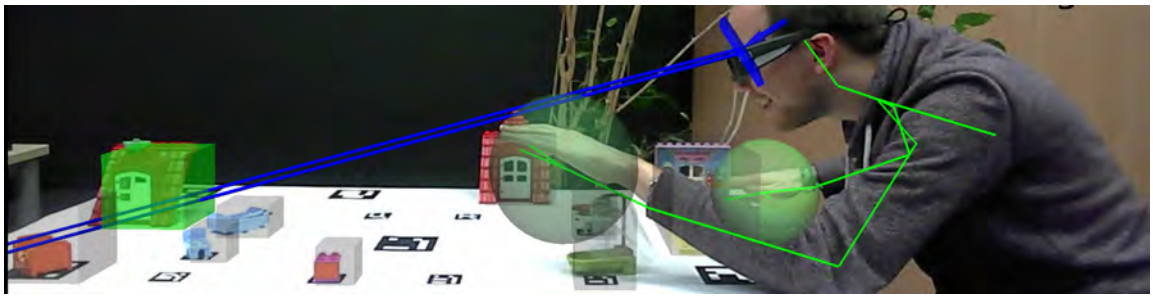

Figure 1: *EyeSee3D analyses eye gaze on dynamic areas of interest in 3D environments. Objects and body parts can be tracked using a variety of tracking systems. Data is fused in a common 3D situation model. The example shows tracked head, hands, and gaze, as well as target stimuli of a LEGO toy kit. The user's gaze (blue) currently fixates a part of the roof (highlighted in green on the left).*

Abstract

With the launch of ultra-portable systems, mobile eye tracking finally has the potential to become mainstream. While eye movements on their own can already be used to identify human activities, such as reading or walking, linking eye movements to objects in the environment provides even deeper insights into human cognitive processing.

We present a model-based approach for the identification of fixated objects in three-dimensional environments. For evaluation, we compare the automatic labelling of fixations with those performed by human annotators. In addition to that, we show how the approach can be extended to support moving targets, such as individual limbs or faces of human interaction partners. The approach also scales to studies using multiple mobile eye-tracking systems in parallel.

The developed system supports real-time attentive systems that make use of eye tracking as means for indirect or direct human-computer interaction as well as off-line analysis for basic research purposes and usability studies.

Keywords: eye tracking, mobile eye tracking, gaze analysis, 3D, augmented reality, social interaction, joint attention

Concepts: •**Human-centered computing** → **Mixed / augmented reality; Interaction techniques;** *User interface toolkits; Empirical studies in HCI;*

*e-mail: thies.pfeiffer@uni-bielefeld.de
†e-mail: prenner@techfak.uni-bielefeld.de
‡e-mail: nlessman@techfak.uni-bielefeld.de

ETRA 2016, March 14 - 17, 2016, Charleston, SC, USA

1 Introduction

Recent developments in mobile eye tracking, both in research [Babcock and Pelz 2004; Li et al. 2006; Kassner and Patera 2012] and industry (Tobii Glasses [Tobii AB 2015], SMI ETG [SensoMotoric Instruments GmbH 2015], ASL Mobile Eye [Applied Science Laboratories 2015], etc.) have enabled researchers to study human behavior in everyday environments. The analysis of eye-movement data gathered under mobile conditions, however, is difficult and requires large resource investments.

Currently, the analysis of eye movements can be differentiated into three different levels: On the first, the **context-free level**, eye movements are interpreted without visual contextualization, just by their movement patterns [Bulling et al. 2011]. This approach enables researchers to identify different types of actions with particular eye-movement patterns, such as reading [Ishimaru et al. 2014]. While the development of the classification algorithm will require contextualized data, the derived algorithms will later-on identify user activities just based on the eye movements alone. Second, on the **immediate-context level**, eye movements are directly linked to the visual content the eyes are fixating on [Pelz 2011]. Such approaches make use of computer-vision techniques to identify the object of interest fixated by the user [Toyama et al. 2012; Brône et al. 2011]. They will typically extract the region of interest that is fixated by the user from the image of the scene camera and try to identify the object based on this information. Third, on the **situation-model level**, an abstract representation of the current interaction context, called **situation model**, is used. Such model-based approaches try to reconstruct the position and orientation of the eye-tracking device and use a 3D model representation of the environment to determine the object of interest [Hammer et al. 2013; Pfeiffer and Renner 2014].

With EyeSee3D we follow the situation-model based approach (see Figure 2) allowing us to analyze mobile eye-tracking data in typical interaction scenarios, such as joint tasks of several humans or interactions of humans with household appliances. With the optimized workflow we have developed, eye-tracking experiments can be set-up and analyzed as quickly as those for desktop-based sce-

ISBN: 978-1-4503-4125-7/16/03
DOI: http://dx.doi.org/10.1145/2857491.2857532

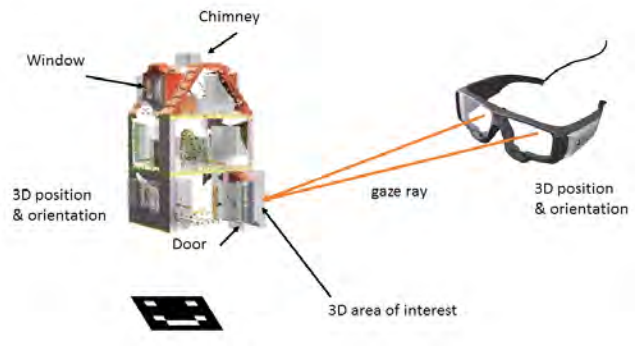

Figure 2: *A 3D situation model is used to determine the objects of interest the user is currently fixating on. In this example, printed fiducial markers are used to create a common coordinate frame in which the eye-tracking system and the target objects are represented.*

narios. We thus broaden the range of interactions that are easily accessible for eye-tracking studies while maintaining the comfort of an automatic identification of fixated objects of interest.

After a short discussion of related work, we will present our approach and describe in particular the workflow we have designed to allow researchers to easily use the approach in their own research. We will provide two examples: for a study on human-human cooperation using two mobile eye-tracking systems we will present an in-depth analysis of the gained advantage in terms of annotation speed compared to manual annotations. In a second example, we demonstrate how even dynamic body movements of dialogue partners can be taken into account and, how e.g., attention targeted at the face or at gesturing hands during communication can be analyzed.

2 Related Work

2.1 Context-free Approaches

Basic eye movements can be measured using electrooculography, which requires less efforts in terms of computational resources than computer-vision based approaches. Several approaches for signal detection and classification have been applied to eye movements being detected using electrooculography, e.g. to classify context-free activities, such as reading [Bulling et al. 2008b] or, later, to differentiate between multiple context-free activities such as typing, reading, eating and talking [Ishimaru et al. 2014]. Advantages of electrooculography consist of not being affected by lighting conditions, a problem that persists in outdoor vision-based eye tracking, requiring less computational resources and as it does not try to link eye movements to external coordinate frames, not suffering so much from drifts. The detection of perceptual activities based on eye movements can be used to provide data for context-aware applications [Bulling et al. 2008a].

2.2 Immediate-context-based Approaches

Context-based approaches are typically tied to computer-vision based eye tracking and require a scene camera for real-world settings. Toyama et al. use an image region around a fixation mapped to a scene-camera video to identify target objects from an object database in their Museum Guide 2.0 [Toyama et al. 2012]. Harmening and Pfeiffer [Harmening and Pfeiffer 2013] extended this approach by employing a hierarchically structured database exploiting geographic relationships to improve processing speed and classification accuary. While these two approaches require a carefully pre-designed database of target objects, Brône et al. [Brône et al. 2011] presented an approach in which the database was created by the user in a training step.

2.3 Situation-Model-based Approaches

In the frame of the ARtSENSE Projekt, Hammer et al. [Hammer et al. 2013] used a Dikablis Wireless monocular eye-tracking system as an interaction device for a museum assistant. They reconstructed depth information of the point of regard based on the 2D marker-tracking capabilities of the Diablis software and cast a ray into their static 3D situation model. The authors state that they use this technique for offline visual analysis and for online implicit interest detection, but do not provide more details on that, in particular not on performance data, such as latency or accuracy. In later work, they showed follow up work on visualizing heatmaps on their situation model [Maurus et al. 2014].

We introduced our marker-based EyeSee3D approach at ETRA 2014 [Pfeiffer and Renner 2014]. At that time, we were already able to analyze the data for one interaction partner inspecting a set of static objects and achieved a coverage of the scene camera videos of about 92 percent. However, in comparison to a human rater this initial version only achieved about 65 percent of agreement. The remaining 8 percent of video non-coverage was due to marker losses (median of continuous marker loss at about 160 ms), in particular when the participants made quick head movements. The system was also not usable for interaction purposes, as it had a rather high latency of about 380 ms. The updated approach presented in this paper provides significant improvements regarding all these issues and adds support for multiple interaction partners, all equipped with eye-tracking systems, and body tracking.

The aforementioned approaches rely on manually created 3D situation models. A slightly different approach is taken by Pirri et al. [Pirri et al. 2011], who create a 3D model of the environment on-the-fly using an approach known as visual slam (simultaneous localization and mapping [Smith et al. 1990]) from the fields of computer-vision and robotics. This model, however, would have to be semantically annotated before it would be useful for further interpretation of the eye movements. In particular, relevant objects would have to be identified. To this regards, their approach is similar to that of Paletta et al. [Paletta et al. 2013], who create an even more realistic static model of the interaction environment based on similar techniques, but with high-quality equipment, and then use this 3D model as means to localize the eye-tracking device relative to this model.

3 Own Approach and Extensions to the Previous Version

The aim of our approach is to enable automatic analyses of experiments in real-world scenarios where mobile eye-tracking is involved. However, it is also applicable for pure virtual reality scenarios, when the position and orientation towards the visualized content is known. The central idea is modeling the environment as an abstract 3D situation model where the relevant stimuli are represented (see Figure 3, right). Several alternative sensors can be used during experiment recording to update the situation model accordingly (sensor fusion, see Figure 3, middle). For a mobile eye-tracking system this includes the head position and orientation as well as the orientation of the eye(s). The head position and orientation can be determined either by an external tracking system or

Figure 3: *EyeSee3D process:* left: *real world figure identification task;* middle: *interaction is tracked using different sensors, such as eye tracker or Microsoft Kinect;* right: *EyeSee3D fuses tracking data in a 3D situation model including areas of interest for all figures and the heads of the interaction partners (boxes). The picture shows a screenshot from the EyeSee3D preview window for the situation at the left.*

by our integrated fiducial marker tracking based approach (see Figure 3, left). For the integrated approach, markers have to be placed in the environment in a way that at least one marker is visible in the scene camera image of the eye-tracker when a stimulus is fixated. Fiducial markers that are detected in the scene camera image can be used to calculate the position and orientation of the camera and thus the eye-tracking device. As this approach requires an instrumentation of the environment with markers, it cannot be applied to every research scenario. However, this approach is easy to set-up and cost-efficient, as it only adds the cost of printed markers. In our experiments, we have observed that participants notice the marker during the preparation, but as soon as they are occupied with the experiment task, we did not observe noticeable numbers of fixations on the markers.

Based on this updated situation model, we can cast a 3D gaze ray into the situation model for each participant wearing an eye-tracking system. In a second step, we can identify the objects of interest being gazed at by intersecting these gaze rays with the 3D models (see Figure 2).

Besides the information for the eye-tracking glasses, other sensor information can be integrated to update the locations of relevant objects during the experiment. One example, which is described later in the paper, is the integration of person tracking using a Microsoft Kinect v2 [Microsoft 2015] to enable the detection of fixations on body parts (hands, face, etc).

3.1 Architecture

The main issue in automatizing the analysis of mobile eye-tracking using our approach is to integrate data recorded by multiple sensors during an experiment session into a 3D situation model of the environment (see Figure 3). For this EyeSee3D is not restricted to eye-tracking data, but also supports other tracking systems, e.g. for full-body tracking.

Currently, EyeSee3D supports SMI eye-tracking glasses, which provide a software API to access the eye-tracking data in real-time. However, in general any mobile eye tracker could be supported. As not all eye trackers support their data in real-time, EyeSee3D also supports offline analysis. Then, the data of the different sensors is recorded during the experiment and the fusion into the situation model is done after the experiment. This approach can also be used, if the available hardware is not powerful enough to support the real-time fusion.

For tracking body movements, we have tested the system with ART optical tracking systems [Advanced Realtime Tracking GmbH 2015] and OptiTrack [NaturalPoint, Inc. 2015], as well as the Microsoft Kinect v2, but other solutions, such as VICON [Vicon Motion Systems Ltd. 2015], can be integrated similarly.

In one EyeSee3D session, several sensor connections can be used in parallel (see Figure 4). For situations in which the different sensors cannot be connected to the same machine, EyeSee3D supports distributed sensor networks. The sensor data is fused in the 3D situation model. This includes head positions and orientations as well as gaze directions for each connected eye-tracking system as well as body data for each motion captured person.

The pre-created 3D model of the environment, containing the user-defined areas of interest, is then augmented by the sensor data and the result is shown live in EyeSee3D's preview window. Based on the fused information, EyeSee3D computes the currently fixated areas of interest for each participant, including a 3D point of regard, the ID of the current area of interest, angular distances to other areas of interest and typical indices such as gaze durations.

A logging API provides support for a flexible output of the original sensor data and the results of EyeSee3D's analysis to a suitable file format, e.g. CSV. A second gaze event API can be used to provide gaze information for real-time interactive systems.

An example set-up for tracking the gaze of one participant on static areas of interest as well as on body parts (see e.g. Figure 1) consists of EyeSee3D running on a high-end consumer notebook (Dell M4800, Core i7-4810MQ, 2.8 GHz, 16 GB RAM). Connected are SMI eye-tracking glasses to track the gaze and one Microsoft Kinect v2 to track bodies. Marker tracking is used to fuse the coordinate frames of the eye tracker (30Hz) and the Kinect (30Hz). The Kinect provides skeleton tracking of all interaction partners. Based on these data, EyeSee3D will then provide gaze events whenever the participant looks on relevant body parts of interaction partners, such as faces, hands, legs, arms, or the torso.

4 Workflow for Designing Experiments using our Approach

Using the presented approach in an experiment requires some preparation steps, which are detailed in the following.

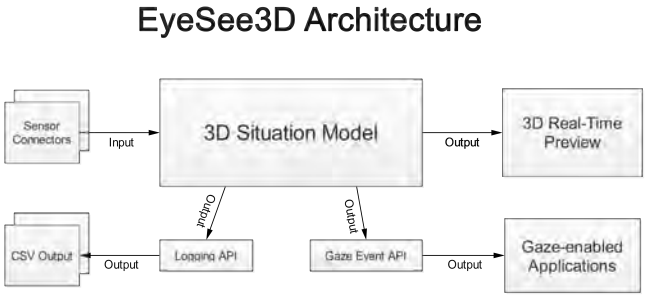

Figure 4: *Data from multiple sensors is fused in a 3D situation model and previewed in 3D. EyeSee3D's APIs provide logging data for experiments and gaze events for gaze-enabled applications.*

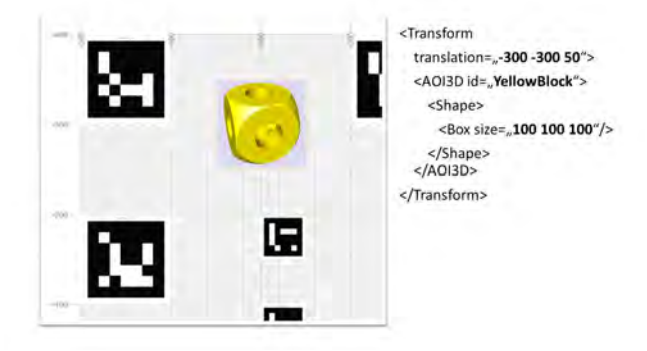

Figure 5: *Several coordinate-system templates of different sizes have been designed to help the researcher to link coordinate frames of different devices. The figure shows a larger DIN-A0 paper with a millimeter grid depicting the exact coordinate system that is defined by the augmented reality markers (dark-black patterns) also printed on the paper. A yellow wooden block is used as target stimuli and placed on the template. The coordinates of the AOI can be read from the millimeter coordinate system and transfered to a file describing the 3D situation model for the experiment (right side).*

4.1 Preparing the Tracking Area

Our approach requires precise knowledge about the head postures of the eyetracked user. The first step is thus to make sure that the target areas are covered by appropriate head-tracking systems. One can use either external tracking solutions, we have worked with VICON, OptiTrack or ART, or one can use the build-in internal tracking solution based on marker-tracking. If the latter is used, the relevant area has to be covered by enough markers to ensure that for each relevant head posture at least one marker can be seen in the scene camera video. The coverage of the areas can easily be checked using the real-time preview of our software.

4.2 Modeling Areas of Interest

Three-dimensional areas of interest have to be modeled for all target stimuli. These models can be rather abstract, e.g. just boxes of the maximum extensions of the target stimulus, or precisely modeled digital replicas of the original physical object. Thus, for the three-dimensional case basically the same choices for the design of areas of interest have to be made as for desktop-based scenarios. Everything that is relevant for the later analysis has to be modeled. This also includes obstacles that could potentially occlude certain target stimuli, e.g. pillars in the room.

Figure 5 shows how this modeling step could be done using the marker-based approach: one of our pre-defined coordinate frames in DIN A0 has been used for this example. It comes with a good coverage of markers for table-top experiments (see Section 5) and provides a millimeter grid for orientation. The yellow cube is one of the target stimuli. It has been placed at the target location on the printed coordinate frame so that its center location and its extensions can be read. In this case, we used a box model to represent the corresponding area of interest. The right side of the figure shows the textual representation that is used to define this target stimuli in our software. An important step is the naming of the areas of interest, here "YellowBlock". This ID is later used in the output file generated by the software, whenever the area of interest has been looked at.

Advanced users may also model complex areas of interest, such as the house depicted in Figure 2 using a three-dimensional modeling software. We use the opensource software Blender3D [The Blender Foundation 2015] for this. This complex model may contain any geometries, also such that are only used for getting potential occlusions right. Multiple areas of interest can be marked in such models by a naming convention. Our software will search for any geometries having names with the prefix AOI_ and will include them in the evaluation report. The house in Figure 2, for example, contains areas of interest such as *AOI_Door*, *AOI_Chimney*, *AOI_Bed*, etc.

4.3 Validating the Tracking

Once the environment has been set-up like this, the software can be started. It will show the live video image from the scene camera and whenever it is in a tracked range (within range of the tracking system or while markers are being visible), then the three-dimensional areas of interest will be overlayed over the scene camera image. This way the experimenter can check whether all areas of interest have been correctly placed.

4.4 Running the Experiment

After the set-up has been validated, the experiment is ready to be started. Besides starting our software and some initial settings, such as declaring the dominant eye or the target name for the logfiles, only the standard-procedure for mobile eye-tracking studies needs to be followed, most importantly the calibration of the eye-tracking device for the particular user. For this calibration procedure, we rely on the implementations provided by the vendor of the eye-tracking systems.

During the experiment, the experimenter can observe the current state of the system using the live preview. Here the gaze-rays are shown being projected into the video of the scene camera view. The view also shows the virtual areas of interest. Whenever the user fixates on such an area of interest, the corresponding geometry is highlighted in the scene and the name of the area of interest is presented in a status line (see Figure 6). This way the experimenter can monitor the data quality achieved with the current tracking setup and the gaze calibration.

4.5 Collecting the Results

The results of an experiment run is a logfile containing all relevant information being provided by the eye-tracking system. In addition to that, our software stores the following information for every frame of the scene camera video:

AOI One column contains the name of the first AOI that has been directly hit by the gaze-ray, or NA if no AOI was hit.

Stacked_AOI This column may contain a list of AOIs if multiple AOIs being stacked in depth are hit by the gaze-ray.

Stacked_AOI_Distances If there are stacked AOIs, then this column contains a list in the same order containing the distances of the AOIs to the observer.

Angular_AOI_Deviations This column holds the angular distances of the gaze ray to all AOIs defined in the scene.

Most commonly one would just use the AOI column to identify the target stimulus that has been fixated. In more complex scenarios, the other columns will provide valuable information for a more in-depth analysis. If the study, for example, is about stimulus non-attendance, then the column of Angular_AOI_Deviations can be checked using a threshold angle for the non-attendance of particular target stimuli. If, for example, the angular distance to a particular area of interest is always greater than 5° of visual angle, the target can be considered as not being attended. The choice of the particular threshold, however, depends on the overal setup of the environment. Similar information are currently not provided by any other software we are aware of.

Besides the logfile, which can be used to analyze the experiment results in a statistical package such as R, SPSS, or even Microsoft Excel, we also provide a small tool that will create annotation tiers for the annotation software ELAN [Wittenburg et al. 2006] representing the relevant information (blinks, fixations, areas of interest).

4.6 Optional Step: Re-Running the Analysis

In our own research life, we every now and then encounter a situation where we want to re-analyze previously recorded studies, for example with a refinement of the definition of some areas of interest or with additional areas of interest. E.g. in one study, we started by analyzing fixations on different product packages and later on decided that we want to differentiate between different parts of the packages (brand, type description/depiction, incredients, etc.). In a standard video-based analysis, we would have had to manually annotate all the recorded videos again.

This is much more convenient with the model-based approach: one just has to update the area of interest definitions (see Section 4.2) and use the offline-mode of our software to re-analyze the gaze data in all the recorded video files once again. This will take some time for larger studies, but can run unattended and in parallel on multiple machines.

5 Example Study: Measuring Joint Attention of Two Interaction Partners

We applied our approach in a study on joint attention between two interaction partners in a cooperative search task. The participants were facing each other sitting at a table where 26 different LEGO Duplo figures were arranged in five rows. Each figure was facing one of the participants, thus revealing necessary disambiguation information only to one of the participants. In each trial, the participants' were given a verbal specification of a figure to be found. Speech and gestures were forbidden, only interaction by gaze was allowed. The participants thus had to negotiate by gaze which figure was the correct one.

For analysing the interactions, a situation model was created including the figures as stimuli. They were modeled using small proxy boxes being sufficient for our analysis. Both participants were equipped with mobile eye-tracking glasses from SMI. The positions and orientations of both devices were determined using the internal marker-tracking approach of our software. The calculated head positions of the participants were then also integrated into the situation model as proxy boxes in order to be able to analyze gaze on the interaction partner as well. So alltogether the study featured 28 areas of interest: 26 figures and the heads of two participants. Data from both mobile eye-tracking systems was integrated in the same 3D situation model, which allowed us to asses mutual gaze on areas of interest, which is at the essence of joint attention.

Figure 6: *Study scenario from the view of one interaction partner. The participant wearing the eye tracker is currently fixating on a figure. On the left, the fixation is depicted in a still from a gaze video generated by the SMI software: the interpretation of the fixated figure is up to the human viewer. On the right, the 3D situation model is aligned to the perspective of the eye tracker and the target figure is determined and highlighted by our software.*

Figure 6 shows the scenario: On the left-hand side, the view from the scene camera of one participant's eye-tracking glasses is shown. The participant's fixation is depicted by the 2D gaze cursor as it is normally generated by mobile eye-tracking software (here SMI's BeGaze). On the right, the same picture is overlaid with the 3D situation model matched to the perspective of the scene camera by our software. The gaze cursor is now replaced by the 3D gaze rays. To visualize that a fixation on a stimulus is ongoing, the fixated figure is highlighted in green. The large green box in the upper part of the image represents the area of interest for the interaction partner. Its position and orientation is updated according to the tracking information gathered from the corresponding participant's scene camera.

The data recorded during our study on joint attention was utilized for evaluating the automatic annotation approach. The evaluation consists of two parts: first, we focus on the coverage of the tracking in terms of frames, i.e. what percentage of the video footage can be annotated automatically (quantity). Second, we compare the automatic approach to human annotators to assess the quality of the automatic annotations.

5.1 Tracking Coverage

During the study, in 13 experiments we recorded 383.5 minutes of interactions, and thus 767 minutes (or 12.8 hours) of gaze videos for the two interaction partners. In total, 37134 relevant fixations (which occured during the actual trials) were recorded.

For generating annotations automatically, using the integrated marker-tracking approach it is crucial that markers are detected in relevant frames. Our system detected markers in 477102 of 529575 relevant frames that occurred during the trials, which corresponds to 90%. The missing frames also include those, where no marker was visible at all in the scene camera image of the eye-tracker, e.g. when participants did not look at the table, but sideways. Table 1 shows the percentage of detections for all experiment runs and interaction partners. The median percentage of frames with marker detections for one experiment and participant is 91.1% (sd=4.9).

Table 1: *Coverage of the video frames with marker detections and coverage of fixations with at least one marker detection, such that an annotation can be done automatically.*

Experiment Number		1	2	3	4	5	6	7	8	9	10	11	12	13	Median	Mean	SD
Frame Coverage in %	Participant 1	92	88	91	90	91	91	90	91	89	93	91	86	90	91	90	1.9
	Participant 2	88	93	93	93	91	91	93	92	92	93	91	71	77	92	89	6.7
Fixation Coverage in %	Participant 1	98	94	98	100	99	100	99	100	99	100	100	98	97	99	99	1.7
	Participant 2	97	100	100	100	100	100	97	99	100	100	100	88	95	100	98	3.4

As the aim in our experiments (and most likely also in most mobile eye-tracking experiments) was analyzing fixations on areas of interest, it is crucial to cover the frames in which fixations occur. 98.5 percent of all marker losses during the study only lasted shorter than 70 ms, which corresponds to two frames in case of our 30Hz scene camera video provided by the eye tracker. The average interval between two marker losses was 687 ms. Relevant fixations start from a duration of about 100 ms, so there should be only few fixations which cannot be automatically annotated at all. Indeed, 36557 of 37134 relevant fixations (or 98.4%) could be annotated, as there was at least one frame during each fixation where markers were detected. In table 1, the percentages of automatically annotated fixations for each experiment run and trial are depicted. Obviously in the last two experiment runs, quality dropped for one participant (maybe because of lighting conditions). In median, 99.4% (sd=2.7) of all fixations of an experiment could be annotated.

5.2 Interrater Agreement Check

We determined the interrater reliability for two human annotators and our automatic annotation system. Their single task was to identify the most likely area of interest for each fixation. The start and end time of the fixation was annotated automatically based on the logfile of the eye-tracking system. Some note about the gaze video: the size of the gaze cursor was left at the defaults of the BeGaze software by SMI, which lead to a width of about 13 pixels. Distances between the centers of two areas of interest were as low as 10 pixels depending on the perspective of the eye-tracked participant. The discrimination task was thus very difficult for the annotators, as well as for the automatic annotation.

Annotator reliability of the human annotators was tested by manually annotating the same 5 minutes of gaze videos of the presented study, which took them almost perfectly equal 1 h and 40 min each. Manual annotation of our data thus had an annotation ratio of 1:10; annotators needed ten time the recorded time to complete the classification of fixations on areas of interest. Projecting this on the total length of 767 minutes of collected videos, human annotators would at least have needed 7670 minutes to annotate the study material, or 128 hours, or 16 days 8 hours a day. Not taking into account a decreasing annotation performance after longer coding sessions or effects of fatigue. The annotation results for the automatic approach were available right after the recording of each study session, without any extra efforts.

The interrater reliability for these two human annotators was found to be Kappa = 0.8314, which is considered to be almost perfect [Landis and Koch 1977]. The strongest disagreement between the raters was when considering fixations that had only a small overlap with a certain area of interest. In about one third of the fixations classified as being 'out' the raters disagreed.

Annotation reliability of our software was tested against both human annotators. For annotator 1 Kappa was 0.7691, which is considered to be a substantial agreement and it is quite close to 0.8, from which on almost perfect agreement starts. It is also quite close to the interrater agreement measured for the human annotators of 0.8314. For annotator 2 Kappa was 0.7343, which is still in the range of substantial agreement.

The software's perceptual advantage. Looking closer at those annotations were the human annotator and the software disagree, it turned out that in 39 of 1103 annotations of annotator 1 the software was actually able to identify the correct target stimulus while the gaze cursor was not visible in the scene camera video. These occurances are emphasized by the design of our study, as it contained two interesting major areas, the table with the target objects and the face of the interaction partner. As the participants had to switch their gaze back and forth between the two areas, some fixated at the figures, then, while holding their heads still, made a short glimpse towards the eyes of the interlocutor. These were the examples in which the gaze ray we could reconstruct using our method was beyond the borders of the scene camera view, the latter only showing the lower half of the head of the interlocutor or even less. As the identification by the software is relying on the three-dimensional model, it is not necessary that the fixation is visible in the restricted view of the scene camera. In these cases, the scene camera is only used for locating the eye-tracking device in space, while the gaze direction determined by the eye-tracking cameras is sufficient to reconstruct the three-dimensional gaze-ray to test for hits with the areas of interest. Thus one source of disagreement between the automatic annotation and the human annotator was that the automatic annotation had a knowledge advantage for fixations beyond the borders of the scene camera video.

Problematic cases were observed which we called "running fixations": in some cases the two human annotators and the algorithm disagreed about the area of interest, with at least two different suggestions. Investigating the movements of the gaze cursor closer for these fixations brought up that the gaze cursor was not stable on one figure during these instances, but ran from one figure to the other figure. This only happened when the two figures were close to each other, but then led to different choices depending on whether the annotator or algorithm weighted the initial or the final position stronger. However, we asked ourselves, how could it happen in the first place, that the fixation classification provided by the BeGaze software could have classified such running eye movements as a fixation? At the time being, we have no explanation for that, but we will investigate this issue further in a future study, because it seems highly relevant to be thoroughly addressed.

Sometimes the contrary happened, participants gazed at a figure, then started turning their heads upwards, as in preparation of a gaze towards the interlocutor, and only after some milliseconds let their gaze follow to jump to the eyes of the interlocutor. These were situations of "smooth pursuit" (inverse, so to say) in which on the 2D surface of the scene camera video the gaze cursor was running, while in fact the focus stayed on the same stationary object.

There are several other types of disagreement, for example when the gaze cursor only touches an area of interest, or touches two at the same time. We will investigate this issue in a dedicated analysis of the study to learn more about these issues. Some of these prob-

Figure 7: *Agreement of one annotator with the automatic annotations for the different stimulus figures. The arrows depict stimuli that were prevalently confused.*

lematic cases are due to the dense experimental setup used in the study. This can be seen in figure 7: Stimuli in close neighbourhood were likely to be confused. The same goes for stimuli that are more distant to the interaction partner. When areas of interest are more clear-cut and have larger spaces in-between, based on the described results we expect that the agreement between automatic annotation and human annotators should be almost perfect.

6 Annotating Gaze on Moving Human Bodies

In many research questions, e.g. regarding human-human interaction, not only fixations on areas of interest in the environment, but also fixations on the interaction partner are of relevance. Gaze on the partner's face can be determined by marker tracking if each participant is wearing mobile eye-tracking glasses, as has been shown in the study described above. For cases where other body parts are relevant, we integrated body tracking of the Microsoft Kinect v2 into our system.

Figure 1 shows an assisstive scenario where a person is wearing an eye tracker. He is supported in building a toy house out of LEGO Duplo parts. The Kinect sensor's depth camera is used to detect the user's body, providing a stick-figure skeleton. In order to establish a link between the coordinate systems of the Kinect and the areas of interest in the environment, markers are tracked on the RGB camera images of the Kinect. This way, the skeleton representing the user can be integrated in the 3D situation model for the gaze analysis. In the figure, the tracked hands of the person are marked as relevant 3D areas of interest and thus self-fixations can be detected automatically. Using augmented reality technology, all data can also be overlaid on top of the scene camera image of the eye tracker.

The integration of moving bodies works in real-time, as does the overall system. In comparison to tracking the body using an optical tracking system (ART, OptiTrack, VICON), using Microsoft Kinect v2 renders this approach low-cost and portable. The Kinect can also be used during the preparation of the study to create 3D scans from the stimuli to have the geometries of the areas of interest in the situation model match the real-world objects more closely. This approach, e.g., has been followed to create the 3D house structure shown in Figure 2.

7 Conclusion

Starting with a distinction of three different ways to analyze eye movements, we have presented EyeSee3D, a model-based approach, which allows us to identify fixated areas of interest automatically. We have shown that the approach can be extended to support more than one camera system by reporting on a study with two eye-tracked participants and the description of a set-up in which one participant's gaze and body movements were tracked using a Microsoft Kinect v2. Both examples also demonstrated that the presented approach is capable of handling moving areas of interest (head, hands). With regard to this aspect, EyeSee3D also surpasses previous methods (see Section 2). The presented approach can be applied using low-cost equipment: tracking can be achieved using a marker-based approach which makes use of the already available scene camera video and consumer devices like the Microsoft Kinect v2 can be used to provide a tracking of interaction partners.

The presented approach requires that the position and orientation of the camera, for mobile eye-tracking systems the scene camera, is tracked. This can be done with a classic outside-in motion tracking system, which is supported by our software, but which is not discussed in this paper. Alternatively, we have presented a system based on an inside-out tracking of salient markers, which significantly outperforms our previous version (see Section 2) in terms of coverage and comparability to annotations of human raters.

While the automatic annotations are already very similar to those of the human annotators, there are some important differences. First of all, the automatic annotations can also cover areas that are not visible in the scene camera video. This is due to the fact that the eye movements detectable by the eye-tracking cameras cover a larger area than what the scene camera covers. This can be in particular interesting for the analysis of actions close to the participants body, an area which is typically not covered by the scene camera. Second, there are ambiguous cases in which human raters use heuristics that require a more systematic analysis. However, the target scenario chosen in the experiment used as an example was very demanding because of small areas of interest in close vicinity. Depending on the perspective, there were cases in which the distance between the centers of two neighboring areas of interest was smaller than the size of the gaze cursor hovering over them. A more sparse distribution of the stimuli would have led to less ambiguities and we expect an even better performance for those cases.

We have also described certain tasks that have to be added to the workflow of designing and conducting an eye-tracking experiment using the presented approach. This workflow and the developed templates for marker-covered coordinate frames will be further refined. All information regarding EyeSee3D can be found on the accompanying website [Pfeiffer et al. 2016].

The setup of the markers and the definition of the 26 areas of interest for the study on joint attention took about 2 hours. A time well spend given that this not only saved us from over 128 hours of manual annotations, but also ensured a repeatable systematic annotation over all recorded sessions.

Overall, with EyeSee3D, studies on eye movements in life-sized real-world and virtual scenarios can be conducted with similar convenience than in desktop-based set-ups. If the marker-based approach can be applied, then there are no additional costs except for printing some paper. If markers cannot be applied, external tracking technologies can be used instead, which, however, come at some cost.

Acknowledgements

This research was supported by the Cluster of Excellence Cognitive Interaction Technology 'CITEC' (EXC 277) at Bielefeld University, which is funded by the German Research Foundation (DFG). This research was partly funded by the German Federal Ministry of Education and Research (BMBF) in the project Adaptive and Mobile Action Assistance in Daily Living Activities (ADAMAAS).

References

ADVANCED REALTIME TRACKING GMBH, 2015. ART Advanced Realtime Tracking Company Website. WWW: http://www.ar-tracking.com/home/, last checked October 2015.

APPLIED SCIENCE LABORATORIES, 2015. ASL Company Website. WWW: http://www.asleyetracking.com/Site/, last checked October 2015.

BABCOCK, J. S., AND PELZ, J. B. 2004. Building a lightweight eyetracking headgear. In *ACM ETRA 2004*, ACM, 109–114.

BRÔNE, G., OBEN, B., AND GOEDEMÉ, T. 2011. Towards a more effective method for analyzing mobile eye-tracking data: integrating gaze data with object recognition algorithms. In *Proceedings of the 1st international workshop on pervasive eye tracking & mobile eye-based interaction*, ACM, New York, NY, USA, PETMEI '11, 53–56.

BULLING, A., ROGGEN, D., AND TRSTER, G. 2008. It's in your eyes - towards context-awareness and mobile hci using wearable eog goggles. In *Proc. of the 10th International Conference on Ubiquitous Computing (UbiComp 2008)*, 84–93.

BULLING, A., WARD, J. A., GELLERSEN, H., AND TRSTER, G. 2008. Robust recognition of reading activity in transit using wearable electrooculography. In *Pervasive Computing*, J. Indulska, D. Patterson, T. Rodden, and M. Ott, Eds., vol. 5013 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 19–37.

BULLING, A., WARD, J., GELLERSEN, H., AND TROSTER, G. 2011. Eye movement analysis for activity recognition using electrooculography. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 33*, 4 (April), 741–753.

HAMMER, J. H., MAURUS, M., AND BEYERER, J. 2013. Real-time 3d gaze analysis in mobile applications. In *Proceedings of the 2013 Conference on Eye Tracking South Africa*, ACM, New York, NY, USA, ETSA '13, 75–78. 00001.

HARMENING, K., AND PFEIFFER, T. 2013. Location-based online identification of objects in the centre of visual attention using eye tracking. In *Proceedings of the First International Workshop on Solutions for Automatic Gaze-Data Analysis 2013*, Center of Excellence Cognitive Interaction Technology, Bielefeld, Germany, 38–40.

ISHIMARU, S., UEMA, Y., KUNZE, K., KISE, K., TANAKA, K., AND INAMI, M. 2014. Smarter eyewear: using commercial eog glasses for activity recognition. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*, ACM, 239–242.

KASSNER, M. P., AND PATERA, W. R. 2012. *PUPIL: constructing the space of visual attention*. PhD thesis, Massachusetts Institute of Technology.

LANDIS, J. R., AND KOCH, G. G. 1977. The Measurement of Observer Agreement for Categorical Data. *Biometrics 33*, 1 (Mar.), 159. 30945.

LI, D., BABCOCK, J., AND PARKHURST, D. 2006. openEyes: a low-cost head-mounted eye-tracking solution. In *ACM ETRA 2006*, ACM, 95–100.

MAURUS, M., HAMMER, J. H., AND BEYERER, J. 2014. Realistic heatmap visualization for interactive analysis of 3d gaze data. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, ACM, New York, NY, USA, ETRA '14, 295–298. 00000.

MICROSOFT, 2015. Kinect for Windows Website. WWW: https://dev.windows.com/en-us/kinect, last checked October 2015.

NATURALPOINT, INC., 2015. OptiTrack Motion Capture Systems Company Website. WWW: http://www.optitrack.com/, last checked October 2015.

PALETTA, L., SANTNER, K., FRITZ, G., MAYER, H., AND SCHRAMMEL, J. 2013. 3d attention: measurement of visual saliency using eye tracking glasses. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems*, ACM, 199–204.

PELZ, J. B. 2011. Semantic analysis of mobile eyetracking data. In *Proceedings of the 1st international workshop on pervasive eye tracking & mobile eye-based interaction*, ACM, New York, NY, USA, PETMEI '11, 1–2.

PFEIFFER, T., AND RENNER, P. 2014. Eyesee3d: a low-cost approach for analysing mobile 3d eye tracking data using augmented reality technology. ACM, Proceedings of the Symposium on Eye Tracking Research and Applications, 195–202.

PFEIFFER, T., RENNER, P., AND PFEIFFER-LESSMANN, N., 2016. Companion website to this paper: http://etra2016eyesee3d.eyemovementresearch.com/.

PIRRI, F., PIZZOLI., M., RIGATO, D., AND SHABANI, R. 2011. 3d saliency maps. In *Computer Vision and Pattern Recognition Workshops (CVPRW)*, IEEE, 9–14.

SENSOMOTORIC INSTRUMENTS GMBH, 2015. SensoMotoric Instruments GmbH Company Website. WWW: http://www.smivision.com/en.html, last checked October 2015.

SMITH, R., SELF, M., AND CHEESEMAN, P. 1990. Estimating uncertain spatial relationships in robotics. In *Autonomous robot vehicles*. Springer, 167–193.

THE BLENDER FOUNDATION, 2015. Blender. WWW: http://www.blender.org/, last checked October 2015, October.

TOBII AB, 2015. Tobii. WWW: http://www.tobii.com/, last checked October 2015.

TOYAMA, T., KIENINGER, T., SHAFAIT, F., AND DENGEL, A. 2012. Gaze guided object recognition using a head-mounted eye tracker. In *ACM ETRA 2012*, ACM, 91–98.

VICON MOTION SYSTEMS LTD., 2015. VICON Company Website. WWW: http://www.vicon.com/, last checked October 2015.

WITTENBURG, P., BRUGMAN, H., RUSSEL, A., KLASSMANN, A., AND SLOETJES, H. 2006. Elan: a professional framework for multimodality research. In *Proceedings of LREC*, vol. 2006, 5th.