EyeSee3D: A Low-Cost Approach for Analyzing Mobile 3D Eye Tracking Data Using Computer Vision and Augmented Reality Technology

Thies Pfeiffer*
CITEC - Cognitive Interaction Technology
Faculty of Technology, Bielefeld University, Germany

Patrick Renner†
Artificial Intelligence Group
Faculty of Technology, Bielefeld University, Germany

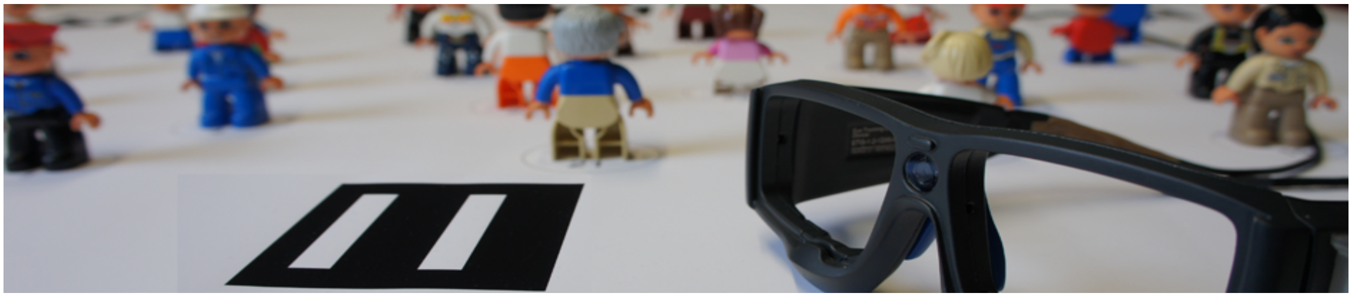


Figure 1: *The set-up: Lower right: The mobile eye tracking glasses, lower left: A fiducial marker, background: The figures are target stimuli of the experiment used to demonstrate the EyeSee3D approach.*

Abstract

For validly analyzing human visual attention, it is often necessary to proceed from computer-based desktop set-ups to more natural real-world settings. However, the resulting loss of control has to be counterbalanced by increasing participant and/or item count. Together with the effort required to manually annotate the gaze-cursor videos recorded with mobile eye trackers, this renders many studies unfeasible.

We tackle this issue by minimizing the need for manual annotation of mobile gaze data. Our approach combines geometric modelling with inexpensive 3D marker tracking to align virtual proxies with the real-world objects. This allows us to classify fixations on objects of interest automatically while supporting a completely free moving participant.

The paper presents the EyeSee3D method as well as a comparison of an expensive outside-in (external cameras) and a low-cost inside-out (scene camera) tracking of the eye-tracker's position. The EyeSee3D approach is evaluated comparing the results from automatic and manual classification of fixation targets, which raises old problems of annotation validity in a modern context.

CR Categories: I.4.8 [Image Processing and Computer Vision]: Scene Analysis—Tracking; H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems—Artificial, augmented, and virtual realities;

Keywords: 3D gaze analysis, eye tracking, motion tracking, marker tracking, geometric modelling

1 Introduction

Using computer-based desktop set-ups for analyzing human visual attention and attentive processes can often only be a technical crutch. Computer-based experiments can be highly controlled and desktop-based eye tracking comes with convenient tools for swift gaze analysis. However, we eventually approve to a loss of validity of our results for the human behaviour in the real-world situations we are originally interested in: In desktop-based experiments participants are rather stable, the content is presented (mostly) only in 2D and is often presented from a non-natural static perspective. Particularly if the situation would normally involve body movements, such as in sports, driving or shopping, this approach is too reductive and recent studies show that the transfer of findings to real-world settings is questionable (e.g. [Gullberg and Holmqvist 2002; Dicks et al. 2010]). Also, features used for the analysis of gaze behaviour, such as fixations or saccades, might differ substantially between desktop and mobile interactions (e.g. [Kinsman et al. 2012]).

With the advent of highly portable lightweight mobile eye tracking solutions [Babcock and Pelz 2004; Li et al. 2006; Kassner and Patera 2012], the transition from desktop to more natural real-world settings becomes technically feasible. However, to counterbalance the loss of control compared to desktop-based experiments, participant and/or item count have to be increased. In addition, the real environment is often more visually cluttered than we design our desktop-based experiments to be. Together with the increased effort required to manually annotate the gaze-cursor videos recorded during mobile eye-tracking sessions, this renders many studies unfeasible: It would simply take months to years to identify and annotate the data.

If the effort for the analysis of gaze data collected in a mobile setting could be similarly reduced as in desktop-based settings, this would in our opinion significantly boost research in visual attention and attentive processes.

In this paper we draw from our experience in augmented and virtual reality and present a low-cost approach to gaze

*e-mail: Thies.Pfeiffer@uni-bielefeld.de
†e-mail: prenner@techfak.uni-bielefeld.de

analysis in 3D, based on visual marker-tracking technology. Similarly to region analysis in desktop-based settings, we define models, i.e. geometric models, of the stimuli content relevant for the gaze analysis, the objects of interest. Instead of single-perspective areas of interest defining target stimuli in 2D screen space coordinates, these models are three-dimensional geometric approximations of the objects of interest and thus inherently support multiple perspectives and esp. perspective shifts of the participants. These geometric models are then synchronized with the real world setting: They can be thought of as a kind of virtual reality overlay on top of the real-world in which each geometric model coincides with the real-world object of interest in position, extension and orientation.

This virtual overlay is anchored in the real world with the help of tracking technology. Distinct markers are defined in the real world and tracked, e.g. with a camera system, that have pendants in the virtual world in exactly the same places. By aligning the virtual markers with the real markers, an isomorphic coordinate system is overlaid on top of the real world in which every real target stimulus can be represented by a virtual proxy object and vice versa.

When the pose of the head is estimated using tracking technology, gaze directions can be represented in the model. Then, the targets of fixations can be automatically classified and the results should be identical to a manual annotation of gaze-cursor videos.

Besides our general approach, we also present an evaluation in which we compare the classification results based on manual annotation of gaze-cursor videos with the automatic classification based on our approach. In addition to that, we compare two tracking solutions, an expensive outside-in tracking and a low-cost inside-out tracking which solely relies on the video recorded by the mobile eye tracking system [Pfeiffer 2012].

2 Related Work

Recently, research in technologies that improve the analysis of mobile eye tracking data has gained momentum. In the following, we will thus present related work organized based on the approach used for identifying the objects of interest in the environment.

2.1 Approaches based on 2D Computer-Vision

For environments for which a geometric model cannot be easily obtained or for analyses targeting highly dynamic scenes, pure image-based approaches have been proposed. These approaches focus on the analysis of local features, such as the area of interest around the fixated area. These approaches, however, then require a learning-phase in which the instances are labelled by a human annotator.

With their SemantiCode approach, Pontillo et al. [2010] follow a semi-automatic approach based on 2D image classification (colour histogram). The gaze data is analysed by SemantiCode and samples of the original scene-camera video centred around the fixation are presented to the human annotator. The annotator labels the samples and SemantiCode step-by-step constructs a database of examples that are used to suggest labels in subsequent presentations.

Toyama et al. [2012] presented Museum Guide 2.0 which uses an adapted SIFT feature extractor and an approximate nearest neighbour approach to classify images of fixated objects extracted from the scene camera, based on a pre-defined database of potential target objects. The database has to be filled with image instances of target objects taken from every perspective. The speed of the classification depends on the number of features and the size of the database. The system was able to operate in real-time when image processing was reduced to less than 25 frames per second on a small set of objects. Harmening and Pfeiffer [2013] extended this approach with a spatially organized database to speed up processing significantly. Brone et al. [2011] proposed a similar approach to interactively create training examples with their pre-study training step *training-by-looking-at.*

The described 2D approaches can deal with stationary as well as with moving objects of interest. However, they require an intensive data collection phase and classifications are not guaranteed to operate in real-time, as they depend on database size. The detection quality also heavily depends on the features of the target objects and it is not possible to differentiate between multiple occurrences.

2.2 Geometry-based Approaches

Geometry-based approaches use a 2D or 3D model of the environment to calculate the intersection of the line of sight and the objects of interest. They require, however, the position and orientation of the eye tracking system at least once for each fixation (tracking). They also require that the geometric model is precisely aligned to the real target objects. Technically, the alignment (often also called registration) and the tracking of the eye tracking system might be handled by the same system.

Areas of human-computer interaction where precise 3D geometric models of the visible environment are ubiquitous are virtual reality and augmented reality. Tanriverdi and Jacob [2000] used gaze as an interaction method in an immersive virtual reality environment presented via head-mounted display to select objects arranged freely in 3D space. Others used gaze for collaboration in virtual reality [Duchowski et al. 2004], for inspection task training [Duchowski et al. 2002] or for system interactions [Gepner et al. 2007]. All these approaches have in common that they have the capability to classify the objects of interest in real-time, but they require an artificial presentation of the target objects to achieve this.

A recent approach that uses similar techniques was presented by Pfeiffer [2012]. They were able to reconstruct so-called 3D attention volumes over real objects, but they lack a geometric model and are thus not able to classify fixations on target objects of interest. A more promising approach was presented by Paletta et al. [2013]: They use a rig consisting of a camera and a Microsoft Kinect to create a 3D scan of the target environment. They are then able to estimate position and orientation of mobile eye tracking data by using the created model as one large 3D marker. Regions in this 3D model can then be annotated and used as regions of interest. As an additional advantage, the 3D model can also be used to visualize the collected data. Their approach is highly suitable for such areas as supermarkets, were the objects of interest are of reasonable size and not densely packed in space. It requires, however, an intensive preparation phase, both manually and computationally.

Somewhere in between is the work of Nilsson et al. [2007] who use gaze for interacting with an augmented reality system also in the context of technical maintenance. They link real and virtual models, but focus on a gaze-based interaction with virtual constructs.

Another interesting approach was presented by Pirri et al. [2011] who create the 3D model of the environment on-the-fly using structure and motion techniques from the area of self-localization and mapping. The created models are rather rough, as participants typically do not move as continuous as required for smooth model generation. But the coarse structure of the environment can still be identified and each point of the 3D model can be linked to instances of scene-camera images that were taken while fixating on the selected area (see also [Kassner and Patera 2012]).

The work presented here is more in line with the work of Munn and Pelz [2009]. In their FixTag approach, they used a computer-vision based feature tracker to estimate the position and orientation of the eye tracker in space. Their system requires a manual calibration of eight reference points in selected keyframes to establish a reference for the tracking. Based on the tracked features, position and orientation of the eye tracker are then interpolated on all neighbouring video frames. Regions of interest that have to be defined beforehand in 3D relative to some reference points are then mapped onto the image space of the scene-camera video of the eye tracking system based on the computed eye tracker transformation. The detected fixations (2D in image space) are then classified according to the mapped regions of interest. The FixTag system operated off-line in less than real-time (2 s per frame on a 2.5GHz MacBook Pro). Only one frame per fixation is classified.

2.3 Marker Tracking

Some of the aforementioned approaches as well as the approach we propose use markers, either visible or invisible (infra-red) to instrument the environment. Kohler et al. [2011] provide an overview about different optical markers.

Instrumenting the environment with markers always raises the discussion, whether the presence of such markers does or does not alter the gaze behaviour of the participants. At least for the infra-red markers used by Tobii no significant effects on the presence of markers have been found [Ouzts et al. 2012].

3 EyeSee3D Approach

With our current research, we are striving for a low-cost solution for automatic 3D gaze analysis that could be applied on-line on any mobile eye tracking system with a scene camera. As not many systems provide a software development kit for on-line access to eye tracking and scene-camera data, our approach also supports an off-line analysis.

The starting point of our approach is a mobile eye tracking system with a scene camera (see Figure 2). It detects human pupils using attached eye cameras. From this, the gaze direction can be calculated and 2D fixations can be detected. These are represented as coordinates in the 2D scene-camera image, i.e. the human gaze is mapped to pixels on the video image. This is what is typically visualized in gaze-cursor videos. Additionally, a 3D direction vector can be computed in a coordinate system which is relative to the position and orientation, furthermore referred to as *pose*, of the scene camera. Our current implementation of EyeSee3D is based on the SMI Eye Tracking Glasses (see Figure 1), the general approach, however, is applicable to all other scene-camera-based eye-tracking systems.

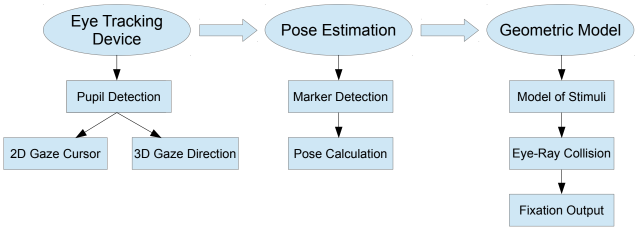


Figure 2: *The different steps of the EyeSee3D approach: All gaze-related computations are typically done by the eye tracking device. The scene-camera image is used for estimating the pose of the device using marker tracking. Both information are combined to identify objects of interest by means of ray-casting in the 3D geometric model.*

Instead of looking at the content of the scene-camera video to identify the objects under fixation, the idea followed by EyeSee3D is to make use of the 3D direction vector representing the line of sight and combine this with a pose estimation. If the scene camera is calibrated, i.e. its intrinsic parameters like focal length, image format and principal point are determined, it is possible to transform known geometries which are detected in the 2D image to their actual 3D pose. In other words, the pose of the camera in the world, relative to the detected geometries, can be calculated. There are several of such geometries which can be easily detected in camera images. We chose fiducial augmented reality markers (one is shown in Figure 1) which have a high contrast and a unique id number encoded by the white areas. Moreover, this kind of markers with its simple structure can be detected in real-time.

Summing up, the link of the scene camera to its 3D environment can be established by detecting the fiducial markers and calculating the pose transformation with respect to the calibrated camera parameters. Combining this with the gaze direction results in a 3D gaze ray that can be casted into the 3D environment. What is missing for annotating fixations on 3D objects of interest for gaze analysis is their position and extension in the world. For this, 3D proxy geometries approximating the objects of interest have to be created and anchored relatively to the tracking markers in the geometric model. The advantage of such a 3D model is that it can be used from any perspective. Thus gaze analysis is not bound to a fixed location. In EyeSee3D we use the W3C standard X3D [Web3D 2001] for modelling, which can be edited manually similar to HTML. It is also supported by many graphical modelling tools.

The virtual model can be aligned to the world because the location of the markers are known in both. It can then be presented as an overlay of the scene-camera image: Each object in the virtual world coincides in position and orientation with the corresponding real object. If the gaze direction is now modelled as a ray which starts from the eye, objects which are fixated can be identified by detecting a collision between the ray and their virtual proxy geometry. The system can then output the occurrence of a fixated object fully automatically (see 6.1).

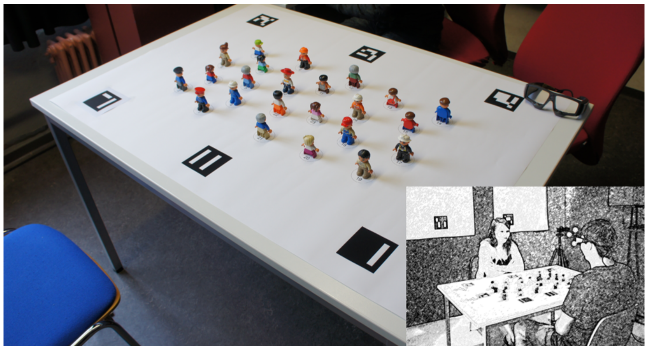


Figure 3: *The target domain of the study: A table with 23 figures of the LEGO Duplo toy set. Lower right: To evaluate the tracking system, parts of the study were conducted within the set-up of a stationary optical outside-in tracking system.*

The described approach supports static scenes. If objects are not located relative to a fiducial marker, their position cannot be tracked with the inbuilt inside-out tracking algorithm. Individual markers, however, can be dynamically moved and every geometry linked to such a marker will be updated accordingly. EyeSee3D can be extended to support additional tracking algorithms for specific scenarios. To give one example: in a subsequent set-up, we added face-tracking to update a proxy geometry for the interlocutor's head [Renner and Pfeiffer 2013]. As an alternative, movements of models can also be modelled explicitly by hand and then using the off-line-analysis.

Our method is also compatible with other tracking mechanisms, such as outside-in optical target-based tracking or inertial tracking. These approaches can be added for enhancing the pose estimation of the eye tracking device and thus increasing the accuracy of the subsequent steps.

4 Practical Example

The EyeSee3D approach introduced above was developed and evaluated in the context of an experiment analyzing gazing behaviour in a face-to-face interaction of two human interlocutors [Pfeiffer-Lessmann et al. 2013].

In the experiment, the interlocutors face each other sitting at the ends of a table (see Figure 3). There are 23 LEGO Duplo™figures standing in five rows on the table, each facing either one of the participants. The figures have specific features only visible to one of the participants. One of the participants wears mobile eye tracking glasses equipped with a scene camera recording the field of view. Each trial starts with a verbal description of the figure sought-after. The task for both participants is to find the figure and the finder has to negotiate with the interlocutor which figure it is. The study was conducted with 12 pairs of participants and each pair had to identify 20 figures.

The positions of the figures are static which allows us to geometrically model the scenario without considering dynamically moving objects. The figures in the set-up are 3D objects which are stacked in depth and may partially occlude each other depending on the perspective of the viewer. Because of this occlusions, approaches which are restricted to 2D planes are not capable of reliably annotating fixations automatically.

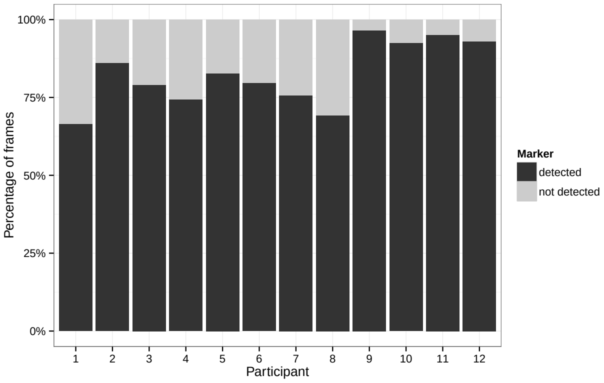


Figure 4: *The figure shows the percentage of frames in which the pose of the eye tracking system could be successfully determined by inside-out marker tracking.*

5 Evaluation of the Tracking

There are two crucial aspects relevant for evaluating the marker-based inside-out tracking: The markers have to be detected in the images provided by the scene camera and the pose of the eye tracking system has to be determined based on the found marker with an adequate accuracy.

In the following section we start by an analysis of how many of the frames recorded by the scene camera we can cover by detecting the markers. If this is not possible, the current pose of the eye tracking system cannot be determined and thus no ray can be cast into the geometric model and consequently no fixation can be classified. This section is then followed by a section on the analysis of the accuracy of the pose estimation.

5.1 Coverage

The orientation and position of the eye tracking system can be estimated for every frame in which a marker can be identified by the computer vision algorithm. In the 228150 frames of our test recordings, this was true for 186537 frames or 81.76 percent. This means that 129.54 minutes of the recorded 158.44 minutes of gaze data can be classified automatically. Figure 4 shows the details for individual participants. After the first eight sessions, we optimized the placement and size of the markers behind the interlocutor and thus were able to optimize frame coverage to over 92 percent. That marker detection failed in the remaining number of frames has several reasons. First, sometimes the participant looked sideways and there simply was no marker within view. More often, however, swift head movements or extreme position changes were causing these issues. The results presented here are based on a per-frame detection and do not include a tracking of the pose from frame to frame, which could further increase coverage.

Another effect that can be observed is that the longer the fixation, the more likely markers have been detected during the ongoing fixation (see Figure 5). During stable fixations, a single marker detected would lead to a successful classification of the object of interest. Figure 6 shows a histogram of observed durations during which no marker was detected. About 40 percent are below 100 ms and thus are not relevant for many analyses of meaningful fixations.

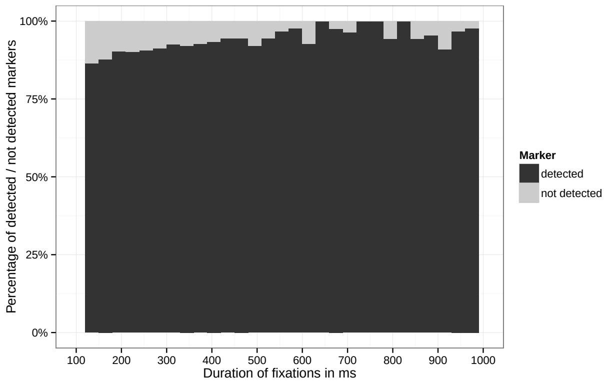


Figure 5: *The figure shows the percentage of fixations (> 100 ms) for which a marker has been detected, i.e. for which the position of the eye tracking system could be detected, depending on the duration of the fixations.*

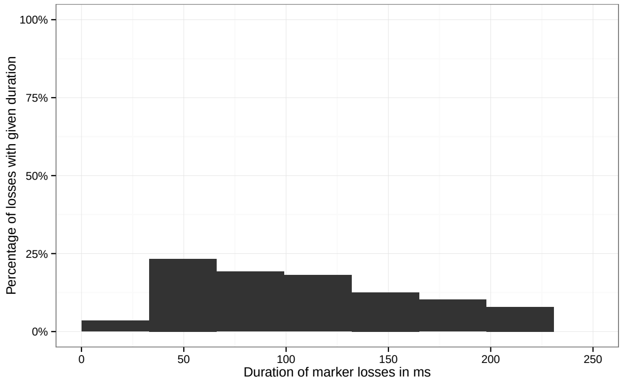


Figure 6: *A histogram of the durations of marker losses found in the experiment. The median duration during which the markers are lost is 158.18 ms.*

5.2 Accuracy

Outside-in tracking systems provide accurate results with little delay. On the other hand, they are often expensive, only limitedly portable and thus one cannot cover large areas. Inside-out tracking, as used in our approach, is cheap and able to cover large areas. The question is, if the accuracy that can be achieved is high enough for analyzing gaze.

5.2.1 Outside-in vs. Inside-Out Tracking

To evaluate the inside-out tracking approach in which the pose of the eye tracking system is estimated by detecting and tracking fiducial markers, we ran an experiment using in parallel an optical outside-in tracking system DTrack2 from AR-Tracking GmbH with a sample rate of 60 Hz and a millimetre accuracy. The data of the DTrack2 system was thus taken as ground truth.

When matching the pose estimations output by the outside-in tracking system with the inside-out tracking system, we found that the latter had a mean delay of 379 ms (SD 90 ms), which is stable over the whole recording. This delay can be attributed to the time required by the computationally more expensive processing chain when using the scene-camera video for pose estimation: video acquisition, computation of gaze direction, marker detection and finally the actual pose estimation. The higher delay is thereby not a problem for on-line or off-line analysis, as the system always synchronizes gaze and position estimates correctly. It would however be a problem if used for interactive applications, as a reaction time of above 400 ms is far too high for many applications.

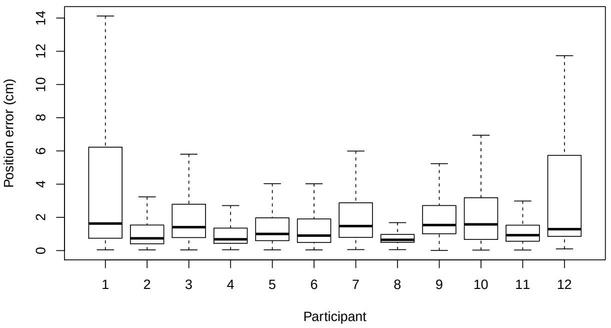


Figure 7: *The position error for the different participants between outside-in and inside-out tracking.*

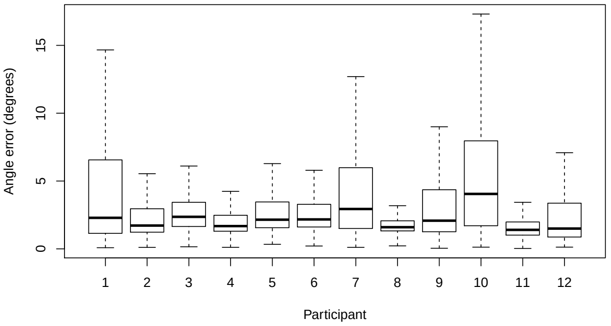


Figure 8: *The angle error for the different participants between outside-in and inside-out tracking.*

Measuring gaze estimation accuracy would require measuring the accuracy of the eye tracker itself. In our study, both pose-tracking systems were used in parallel, so any errors in estimating the gaze direction apply to both approaches. We therefore only consider the accuracy in pose estimation here.

The accuracy of the pose estimation can be determined in terms of position (see Figure 7) and angle errors (see Figure 8). The adjusted overall mean position error is 1.23 cm for the three dimensions (SD 0.90 cm). The adjusted mean angle error of 2.25 degrees (SD 1.46 degrees) is small enough to ensure an accurate calculation of fixations on objects which are not in direct proximity.

The errors were adjusted by removing outliers that result from a technical experiment with two additional markers whose positions and orientations were not pre-defined in the geometric model but learned on-line during their first encounter during the experiments. They were positioned at the

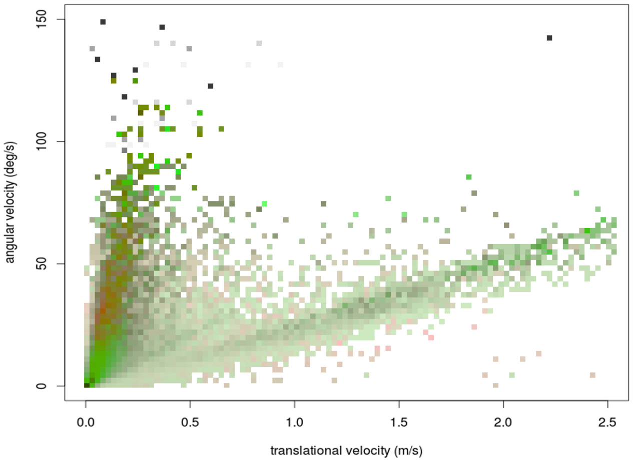


Figure 9: *Ratio of detected and not detected markers relative to angular and translational velocity. The more detected markers prevail, the greener. The more markers are lost, the redder. Intensity is modulated by density of occurrences.*

wall behind the table to increase tracking probability when the participants were looking upwards at their interlocutor. We were testing the possibility to start with one pre-defined marker and extend the knowledge about the geometry on-the-fly. However, our current approach is not stable enough and results in a higher standard deviation when these markers are the only one visible, because errors during the initial learning phase are not yet optimized over time.

While an outside-in tracking system is relatively robust against quick head movements, they result in smearing in the camera of the inside-out system. Therefore, especially for high angular head velocities, the rate of marker detections decreases. Figure 9 reveals that from angular velocities of more than 18 deg/s the detection rate quickly decreases (red area). Pure translational head velocities have a minor effect.

6 Evaluation of the Classification

6.1 Model-based Approach

By the use of a tracking approach (inside-out/outside-in) the pose of the eye tracker and the positions of the markers are known. For automatically annotating fixations on stimuli, marker locations can be made explicit by modelling them in the geometric model of the stimuli as active elements.

For the human-human experiment set-up described before the required target stimuli are the figures and the interlocutor's face. The figures can be part of a static geometric model of the scenario. Thus, 23 geometric 3D representations of them were created. As we are only interested in whether or not a figure was being gazed at, we could omit small details. Thus, in this case small proxy boxes could be used for representing the figures, as they suffice for approximating their actual characteristics. The position of the interlocutor's face is approximated by a rectangle located at the position where the participants are seated (see Figure 10).

Having modelled the scenario, it has to be aligned with the real-world scene, i.e. with the video of the eye tracker's scene camera. Therefore, the markers used for tracking have to be modelled as well. When knowing the position and orientation of them in the real world, their virtual correspondents can be anchored as an overlay over the scene. This way, an isomorphic coordinate system is spanned in the real world. Since the stimuli and the markers are at fixed positions, the positions of the stimuli are relative to the markers. Thus, when the virtual markers are aligned with their real-world correspondents, also the proxy geometries of the stimuli are located in the right place (see Figure 10).

By the use of tracking, the positions of the markers in the real world and the position and orientation of the eye tracking glasses is known. When combining this with the gaze direction detected by the eye tracking system, the former can be made explicit by modelling it as a gaze ray (see rays in Figure 10). This enables testing for collisions between the proxy geometries of the stimuli and the ray. An object of interest is fixated when the gaze ray intersects with the proxy geometry and consequently the line of gaze intersects with the stimulus in the real-world.

When a collision is detected, the virtual proxies actively generate an event which is logged by the system. The generated events, as well as all the relevant data from the system (raw eye tracking output, tracking results, additional information like timestamps) are written into an extensive log-file to be used for the post-hoc analysis.

6.2 Manual vs. Automatic Classification

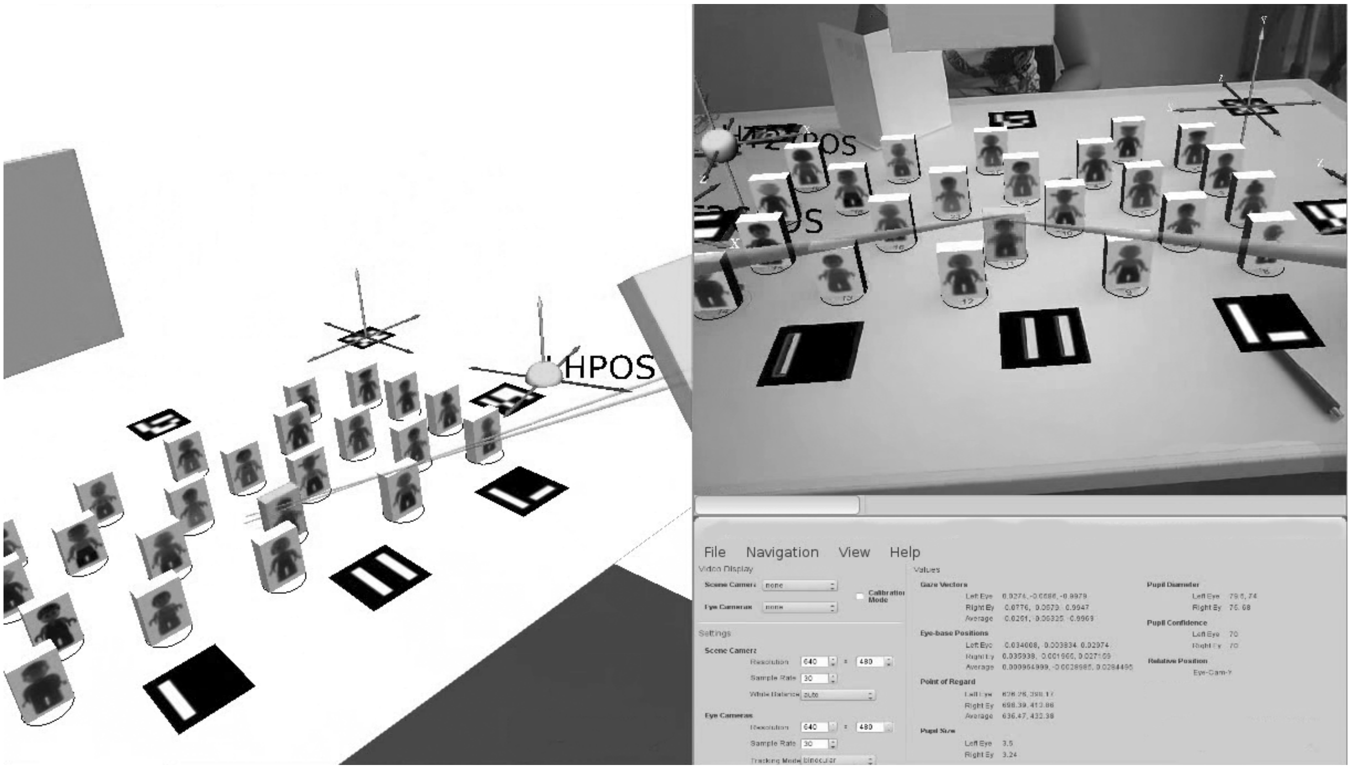
We have first results on comparing manual annotation with automatic classification. For this we annotated the videos of one participant and compared the data with the results provided by EyeSee3D. For 784 relevant fixations the automatic classification agreed with the human annotator on 64.29 percent. This is good, but not yet satisfying.

This cannot be explained by lost markers alone, as we have a detection rate that is above 90 percent in the selected case. And even so, as Figure 6 shows, markers are rarely lost longer than needed for a fixation of interest (> 100 ms), so the real upper boundary should be even higher.

In contrast in the evaluation of the FixTag system [Munn and Pelz 2009], two sessions of about 2 minutes of interaction each (184 and 172 fixations respectively) were automatically analysed with a 99.5%/98.3% agreement of FixTag with manual annotators.

How can the difference in performance be explained? First of all, the context is completely different. The target scene used for FixTag consisted of 2D planes in a perpendicular arrangement, so no occlusions between different regions of interest occurred. Also, the target areas in our study are very small, only 3-4 cm wide and 8 cm tall, and they are densely packed. This results in many cases of ambiguities when annotating gaze data. As will be detailed in the following, this raises more general questions on the classification of gaze data in 3D scenarios.

Investigating on the cases in which automatic tracking and human annotation disagreed, we found the following main theme: As Figure 11 demonstrates, the target objects are densely packed. Thus given a single frame with a gaze-cursor it is often difficult to decide on which figure the participant actually focussed on. This is the situation the automatic classifier is currently faced with, as it computes the intersection of the line-of-sight on a per-frame basis. The human



Video Display
Scene Camera: none
Eye Cameras: none
□ Calibration Mode

Settings
Scene Camera
Resolution 640 x 480
Sample Rate 30
White Balance auto
Eye Cameras
Resolution 640 x 480
Sample Rate 30
Tracking Mode binocular

Values
Gaze Vectors
Left Eye 0.0274, -0.0686, -0.9979
Right Ey 0.0776, 0.0579, 0.9947
Average 0.0261, 0.06325, 0.9969
Eye-base Positions
Left Eye 0.034008, 0.003834, 0.02974
Right Ey 0.035938, 0.001966, 0.027109
Average 0.000964999, -0.0028985, 0.0284495
Point of Regard
Left Eye 626.26, 398.17
Right Ey 698.39, 413.86
Average 636.47, 432.38
Pupil Size
Left Eye 3.5
Right Ey 3.24
Pupil Diameter
Left Eye 79.6, 74
Right Ey 75. 68
Pupil Confidence
Left Eye 70
Right Ey 70
Relative Position
Eye-Cam-Y

Figure 10: *A screenshot from the off-line analysis of the tracking data. The left side shows the virtual model of the scenario. The upper right shows the fit of the virtual overlay on top of the real video.*

annotator, however, apparently used some heuristics to arbitrate between candidates in ambiguous situations. In the example, the annotator could, knowing which of the figures had been fixated on in the previous frames, be either conservative in maintaining his decision or progressive and attribute the fixation to the new figure. In a few cases, the decision even seemed to be backtracked, so when the following fixation would again be on the initial figure, the fixation in between was also attributed to the initial figure although it was more than half-way on the other figure. In some examples it also mattered whether the head or the foot of a figure was being touched by the gaze-cursor.

Thus, before comparing manual and automatic annotations, we would need to specify a coding manual for gaze annotations that covers all these cases. We would then need to implement the rules of the manual in the automatic analysis system if we are interested in creating an automatic classification working the human-way. It could, however, be also worthwhile to consider taking the automatic classification as the more objective approach, without any, maybe misleading, interpretations of the gaze behaviour of the participants.

7 Conclusion

We presented EyeSee3D, an approach for an automatic analysis of mobile eye tracking data that operates in real-time, but currently with a delay of about 380 ms. The presented approach uses low-cost printable markers to instrument the environment, but it is compatible with other tracking technologies as well, such as the outside-in optical tracking system by AR-Tracking. In our example we used SMI Eye Tracking Glasses for real-time analysis. EyeSee3D is also able to analyse previously recorded sessions in off-line mode and thereby supports off-line only systems such as Tobii glasses.

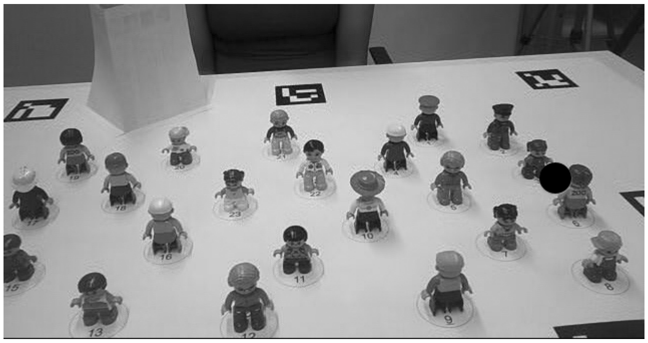


Figure 11: *Annotating the videos of the test scenario is difficult even for human annotators. In the example, the gaze cursor (black circle) is in between two figures on the right during a fixation. Which figure is finally annotated depends on the deliberations of the annotator.*

The current implementation of EyeSee3D supports static scenes of any complexity (several objects stacked in 3D). EyeSee3D also supports moving objects, but only if they either bear a tracking marker (then the objects can be moved freely) or if their movements can be represented in the geometric model as animations of the proxy models. It can, however, be easily extended to support additional tracking algorithms (see Harmening and Pfeiffer [2013] or Renner and Pfeiffer [2013] for examples).

One issue is the required instrumentation, which might be difficult in some domains, e.g. shopping, where the environment is already densely packed. We are therefore planning to add other tracking methods, such as poster trackers, which can use arbitrary areas of high visual saliency that might be part of the natural environment to make the instrumentation of the environment less obtrusive.

Acknowledgements

This work has been supported by the Deutsche Forschungsgemeinschaft in the SFB 673 Alignment in Communication and the Center of Excellence Cognitive Interaction Technology.

References

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.

Dicks, M., Button, C., and Davids, K. 2010. Examination of gaze behaviors under in situ and video simulation task constraints reveals differences in information pickup for perception and action. *Attention, Perception, & Psychophysics 72*, 3, 706–720.

Duchowski, A. T., Medlin, E., Cournia, N., Gramopadhye, A., Melloy, B., and Nair, S. 2002. 3D Eye Movement Analysis for VR visual inspection training. In *ACM ETRA 2002*, ACM, 103 – 110.

Duchowski, A. T., Cournia, N., Cumming, B., McCallum, D., Gramopadhye, A., Greenstein, J., Sadasivan, S., and Tyrrell, R. A. 2004. Visual Deictic Reference in a Collaborative Virtual Environment. In *ACM ETRA 2004*, ACM Press, 35–40.

Gepner, D., Simonin, J., and Carbonell, N. 2007. Gaze as a supplementary modality for interacting with ambient intelligence environments. *Lecture Notes in Computer Science 4555*, 848.

Gullberg, M., and Holmqvist, K. 2002. Visual attention towards gestures in face-to-face interaction vs. on screen. In *Gesture and Sign Language in Human-Computer Interaction*. Springer, 206–214.

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.

Kassner, M. P., and Patera, W. R. 2012. *PUPIL: constructing the space of visual attention.* PhD thesis, Massachusetts Institute of Technology.

Kinsman, T., Evans, K., Sweeney, G., Keane, T., and Pelz, J. 2012. Ego-motion compensation improves fixation detection in wearable eye tracking. In *ACM ETRA 2012*, ACM, 221–224.

Kohler, J., Pagani, A., and Stricker, D. 2011. Detection and identification techniques for markers used in computer vision. In *Visualization of Large and Unstructured Data Sets-Applications in Geospatial Planning, Modeling and Engineering*, vol. 19, 36–44.

Li, D., Babcock, J., and Parkhurst, D. 2006. openEyes: a low-cost head-mounted eye-tracking solution. In *ACM ETRA 2006*, ACM, 95–100.

Munn, S. M., and Pelz, J. B. 2009. Fixtag: An algorithm for identifying and tagging fixations to simplify the analysis of data collected by portable eye trackers. *ACM Transactions on Applied Perception (TAP) 6*, 3, 16.

Nilsson, S., Gustafsson, T., and Carleberg, P. 2007. Hands free interaction with virtual information in a real environment. *Proceedings of COGAIN*, 53–57.

Ouzts, A. D., Duchowski, A. T., Gomes, T., and Hurley, R. A. 2012. On the conspicuity of 3-d fiducial markers in 2-d projected environments. In *ACM ETRA 2012*, ACM, 325–328.

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.

Pfeiffer-Lessmann, N., Pfeiffer, T., and Wachsmuth, I. 2013. A model of joint attention for humans and machines. In *Book of Abstracts of the 17th European Conference on Eye Movements*, vol. 6. Journal of Eye Movement Research, 152.

Pfeiffer, T. 2012. Measuring and visualizing attention in space with 3d attention volumes. In *ACM ETRA 2012*, ACM, 29–36.

Pirri, F., Pizzoli., M., Rigato, D., and Shabani, R. 2011. 3d saliency maps. In *Computer Vision and Pattern Recognition Workshops (CVPRW)*, IEEE, 9–14.

Pontillo, D. F., Kinsman, T. B., and Pelz, J. B. 2010. Semanticode: using content similarity and database-driven matching to code wearable eyetracker gaze data. In *ACM ETRA 2010*, ACM, 267–270.

Renner, P., and Pfeiffer, T. 2013. Studying joint attention and hand-eye coordination in human-human interaction: A model-based approach to an automatic mapping of fixations to target objects. In *Proceedings of the First International Workshop on Solutions for Automatic Gaze-Data Analysis 2013*, Center of Excellence Cognitive Interaction Technology, Bielefeld, Germany, 28–31.

Tanriverdi, V., and Jacob, R. J. K. 2000. Interacting with eye movements in virtual environments. In *CHI 2000*, ACM Press, New York, 265–272.

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.

Web3D, 2001. X3D. Online. http://www.web3d.org/x3d/specifications/x3d/, last checked November 2013.