Analysis of Mobile Eye-Tracking data: Scene Mapping

Jannik Hofmann
Chapter II-A
University of Tübingen

***Abstract*—Originally part of a larger collaborative literature review in the field of human-computer-interaction, this term paper gives an overview of the methods that can be applied to collected eye-tracking gaze data in order to map it toward a given 3-dimensional scene. Furthermore, it discussed how to reconstruct such a scene of the real world in the digital realm, depending on the eye-tracking hardware or based on the collected data that can be utilized for this task. Finally, a discussion of visualization techniques shows how to display the gaze data within this 3-dimensional scene in a meaningful manner.**

***Index Terms*—eye-tracking, gaze analysis, scene mapping, scene reconstruction**

I. Introduction to analysis of mobile eye-tracking data

Recording and understanding human behaviour is central part in the field of human-computer-interaction. Data collected during such experiments can guide to us integrate our digital environment more seamlessly and to gain a deeper understanding into the human psychology when living in and interacting with reality.

To quantify this behaviour and build such an understanding of our connection to the world around us, eye-tracking studies have proven to be a reliable source of data that can give us a glimpse into the human psyche. Not only is the gaze a very effective and quite accurate representation of the participant's attention at any given point in time, this data can even lead to models about people's emotion and intent of action.

When handling gaze data from an eye-tracking experiment, it always should be considered in the context of the world around it. Not only to understand and quantify what the participant was looking at, but also to map the gaze directions to coordinates in a virtual scene.

This section of the literature review will take a look at scene analysis and discuss how to generate a semantic understanding about the world around us from various data collected during eye-tracking studies. We will present how this gaze data can be mapped onto 3-dimensional virtual representations of the real scene and how to reconstruct such a scene from collected data. Furthermore, we will take a look at how to visualize the gaze information in that scene and what hardware can be used to achieve such results.

II. Scene analysis

We humans move around complex 3-dimensional environments in our everyday lives. To meaningfully understand our interactions with the real world, researchers are trying to quantify and model these scenes. With the advent of augmented reality in smartphones and a move towards seamless integration of the digital into the real world, it becomes increasingly important to gain a deeper understanding of the world around us from the data collected during mobile eye-tracking studies.

We describe how we can map the gaze information collected during an eye-tracking study towards 3-dimensional worlds. That means, how to translate the angular gaze directions into absolute scene coordinates. We consider how to track positional and angular head movement by the participants and how to treat object edge cases due to uncertainty owed to the precision of gaze data. This 3D mapping can be done by ray-tracing or by recording depth information. Afterwards, the gaze information can then be mapped either on a pre-existing virtual model of the scene, or one can be reconstructed from various types of data collected during the eye-tracking experiment (such as a 2D video, 3D depth video or additional video feeds). Furthermore, we present how to interact with the virtual environment and how to visualize the collected gaze data in it. This includes point clouds, voxel representations, polygon meshes, heatmaps and presenting ways to export these visualizations. And we take a look at hardware that can be used for 3D mapping and even for 3D reconstruction, such as the HoloLens 2 and the Vive Pro Eye.

A. Mapping towards 3D scenes (and reconstruction of such)

1) Introduction: Usually, eye-tracking studies are focused on analyzing gaze behavior of the participants in a 2-dimensional environment. That means, the gaze information only needs to be mapped on a flat plane, that resides in front of the participant's head.

However, since the world around us is made up of complex 3-dimensional geometries, it is important to consider how to map eye-tracking data in a 3-dimensional scene. This might for example be a participant looking at a statue from different directions, or participants moving around freely in a museum with many rooms, glancing at the various exhibits. Furthermore, these scenes do not always exist as a digitized virtualization in advance. Therefore it can also be a challenge to reconstruct that 3-dimensional scene in order to have enough data about the geometries of the space and the objects it contains, so that the researchers can generate a meaningful interpretation of the gaze data collected by the eye-tracker.

2) 3D mapping: Eye-tracking devices usually collect 2-dimensional data. The most common form of this are the pupils gaze coordinates, which specify the gaze direction of the participant over discretized time intervals. However, when measured with mobile eye-trackers, this data is relative to the head of the participant, so more information is needed to translate these coordinates into a meaningful representation. Depending on the hardware used, researchers can collect various types of additional input data during eye-tracking studies, which can then be utilized for localization, 3D gaze mapping and even for 3D reconstruction of the scenes.

a) Tracking of head movement: Furthermore, in many eye-tracking studies the participants have the ability to move around in space as they would normally do, so a head mounted eye-tracking device is employed. As the gaze data is relative to the head position and rotation and the participant can freely move their head, the movement of the head needs to be determined and considered when calculating the gaze positions in order to generate meaningful results from the experiment.

To locate the participant's head position and rotation in 3-dimensional space, usually a 2-dimensional RGB-video feed of the participants field of view is being recorded during the study. This RGB-video not only helps the researchers visualize the collected data and replay the participant's experience from their point of view, the video can then also be used to set markers and optically track the movement of these markers to then calculate the camera's transformation in 3-dimensional space, utilizing algorithms such as SLAM (Simultaneous localization and mapping) [4]. To support this process, a common strategy in eye-tracking studies is to print out distinct black-and-white markers with high-contrast corners and use them as static markers for the tracking algorithms, as is for example used in EyeSee3D [17].

Also, accelerometer data collected within the eye-tracker can augment these calculations, resulting in a higher accuracy.

Another way of tracking the participant's head movement is often found in Virtual Reality headsets like the HTC Vive, where active markers are mounted in a fixed position in the environment (for example in the corners of a room) [2]. Sensors on the headset itself then use signals from these markers to accurately determine the position and rotation of the headset in space.

b) Using 3-dimensional RGB-D video: To be able to map the gaze data onto an accurate 3-dimensional scene of the environment, the researchers need to have a pre-existing 3-dimensional model of that scene or reconstruct it themselves using data collected during the experiment. The model could be created beforehand, for example by measuring out the rooms and constructing a computer model of the scene by hand, or by scanning the environment with specialized hardware before the experiment takes place. When reconstructing the scene with data collected from the experiment itself, the 2-dimensional video feed can be used.

However, to aid in making this reconstruction process more robust, an RGB-D (containing depth information) feed can be recorded instead of the standard 2-dimensional RGB video feed; provided the eye-tracker has such a dedicated camera or two RGB-cameras to augment the depth data, or such hardware is mounted on the eye-tracking device. This data can then be utilized by the SLAM algorithm [4] to localize and track the participants head in 3-dimensional space [9], [21].

Using the depth information of the recorded video as measure of distance, it is trivial to calculate a coordinate in space, given the position and rotation of the head and the gaze direction at that time.

c) Calculating gaze coordinates in the scene: Given a 3-dimensional representation of the environment and the head position and rotation for each point in time, a normal ray can be traced in the gaze direction, until it meets an object in space, which would then be the coordinate of the users gaze target. The accuracy and precision of the recorded gaze data also applies here, in addition to the accuracy of the virtual scene and the calibration in that space. The farther away this coordinate is, the larger the area of uncertainty about the true user's gaze target.

Also, considerations need to be made when looking near the edge of objects. In a 2-dimensional visualization of the users gaze data (for example a heatmap over a frame of the users point of view), inaccuracies in gaze interpretation usually result in small deviations in the visualization. However, in 3-dimensional space, these inaccuracies can result in the gaze ray hitting an object close to the participant or missing it, and therefore traveling many more meters before hitting a completely different object in the distance. Assumptions can be made about the participant's preference to keep looking at a certain object. For example, if they have been watching the same object for a while and are now looking right at the edge of that object, it could be assumed that the participant is still focusing the object, rather than watching the sky behind it. This uncertainty can be analyzed and minimized with Bayesian inference methods to estimate the most probably result for gaze-to-object mapping [1].

3) 3D reconstruction: Reconstructing a real-world scene in a virtual model is a crucial problem when trying to understand human behavior in complex environments. There are different methods to achieve this task, depending on what data can be collected about the real-world scene.

a) Reconstruction from 2-dimensional video: First of all, reconstruction by a 2-dimensional video feed is rather difficult, but increasingly important with the advent of smartphones and the growing demand for augmented reality capabilities within these devices.

Although this problem can be viewed as a research topic separate from eye-tracking, the technique for this type of data is also especially relevant for eye-tracking studies, as most eye-trackers already have an integrated RGB-video camera (rather than a depth sensor).

This challenge is further complicated if the video feed contains arbitrary movement (i.e. chosen by the participant) instead of strategic exploration of the 3-dimensional scene, as one would do to retrieve a complete scan with few occlusions [10].

Solutions for this challenge include calculating the camera movement by traditional tracking methods such as SLAM [4] and then calculating a point-cloud model from the colored pixel values, utilizing camera movement in consecutive frames to determine the distance of these colored voxels from the camera.

Newer approaches use machine learning to feed the video into a network and receive a voxel or point cloud representation of the virtual world. Although requiring a large amount of training data, these methods tend to be more robust in their understanding of real-world geometries due to the high generalization capabilities of neural networks [14].

b) Reconstruction from 3-dimensional video: An arguably easier problem is the reconstruction by a RGB-D video feed, meaning a video feed that includes depth data. This can be achieved by mounting a time-of-flight camera to the eye-tracker or by utilizing two cameras and using a stereoscopic algorithm to approximately calculate the depth map.

While still needing to accurately estimate the camera movement from this 3-dimensional video feed, the depth data can aid in these calculations and no guesses need to be made about how far the objects in the video are in reality. This simplifies the problem, as one could already make a point-cloud model of the environment with just one single frame of that video feed, with the resulting accuracy depending on the accuracy of the collected data. The data of multiple frames then needs to be matched into a single virtual scene, having to decide how to handle overlapping regions, conflicting data points and outliers [6]. For example, Bayesian point cloud reconstruction could be used to merge the various observed data points into a single robust model [7].

Fig. 1. 3D scene reconstruction as point cloud from 2D eye-tracker video [9]

c) Reconstruction from additional video feed: If researchers want to avoid the problem of arbitrary movement by the participants (for example nobody walking behind an object), which can leading to areas of missing data, they can record an additional video for 3-dimensional reconstruction, independent from the eye-tracker's field-of-view perspective.

This could for example be achieved by exploring the space with drones that fly above or around the space, in which the participants will move around [13]. This way, the researchers can actively track which areas are covered well by the reconstruction and which perspectives need more input data.

4) Visualization:

a) 3-dimensional scene representations: The virtual scene could then be displayed as a point cloud, matched into a colored voxel representation of the scene or converted into a polygon mesh, depending on the properties of the provided virtual scene or the technique used to reconstruct that scene.

Furthermore, this model could be used to estimate simple geometric shapes in the environment, covered with textures from the captured color data, based on assumptions about the shapes in that scene (for example flat house facades with preferably vertical and horizontal edges resulting in simpler polygon meshes) [5], [15].

b) Rendering gaze data: We can then render the collected gaze data in various ways onto that 3-dimensional model, in a similar fashion as one would do for 2-dimensional eye-tracking experiments. Saccades and fixations can be drawn as points and lines, respectively. If they occur on the convex surface of the object instead of jumping through empty space, they should preferably be rendered directly onto the object as part of its texture. This way, the visualizations are directly bound to the surface of that object and can be easily exported with the object itself.

Heatmaps can also be rendered into the texture in the same manner [16], or they can be drawn onto a transparent cylinder that wraps directly around the object, in order to avoid high-cost texture calculations on high-poly models. When drawing a heatmap onto the scene, it could be advantageous to not keep the affected area restricted to the object itself, but rather to consider the viewing direction and perspective of the observer. This way, heat can spill on the objects or wall behind the observed object, while creating a shadow of unaffected area right behind the object, resulting in a direct representation of all areas that were close to the participant's fovea and therefore in the observer's center of attention [12].

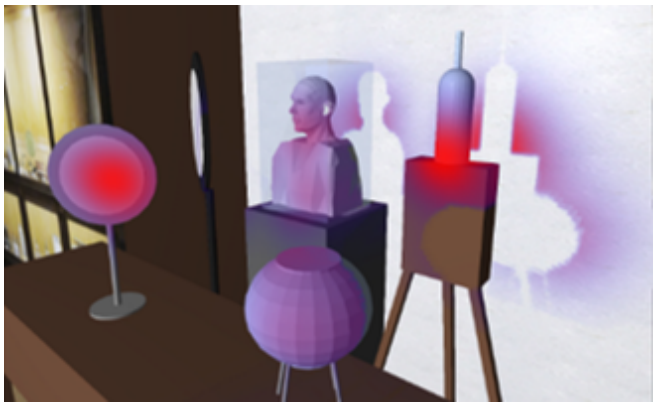

Fig. 2. Heatmap of gaze data spilling on the wall behind the object [12]

As an alternative to drawing heatmaps as a texture, whole objects could be colored depending on their gaze duration [20].

c) Exporting visualizations: This visualization can then be explored in 3-dimensional space by the researchers or users, or static images of a certain perspective within that scene can

(a) Projected (top view) (b) Object-based (c) Surface-based

Fig. 3. Different visualization approaches for object-based gaze data [20]

be rendered. With static images, an informative perspective needs to be selected, that allows the viewer to grasp the relevant data in that scene, without hiding important aspects about the data either by occlusion or by positioning it outside of the viewed area.

In some cases, it might make sense to render a 360° view of the virtual visualization scene as a flattened panorama. And if the 3-dimensional scene contains mostly flat objects of interest (for example paintings in a gallery with many rooms), it could be more useful to isolate these objects and simply render the gaze data within these objects like classical 2-dimensional eye-tracking results.

5) Hardware: Various types of hardware exist that are specifically designed for eye-tracking studies. Mentioning all existing eye-tracking devices would be out of scope for this area, so we will focus here on showing devices that support 3-dimensional tracking and reconstruction. It has been explained above how traditional eye-tracking hardware can be augmented, in order to capture more data relevant for 3-dimensional mapping or reconstruction. This includes multiple cameras, RGB-D cameras or depth sensors to produce a 3-dimensional video feed, accelerometers and fixed trackers to augment positional and angular tracking of the device.

a) VR / AR: However, some consumer goods can also be used to achieve the same results, mainly virtual and augmented reality headsets, which increasingly include eye-tracking capabilities out of the box. Utilizing these devices can simplify the workflow of researchers, as these types of headsets rely on accurate positional and angular tracking in 3-dimensional space and even already provide some 3d-reconstruction capabilities implemented by the manufacturer. On the other hand, implementing eye-tracking methods in consumer electronics and in the entertainment industry opens the door to new frontiers of human-computer-interaction, where users only need to move their eyes to interact with a machine, eliminating the need for hand movements and breaking down conceived barriers in the direct interaction with a virtual environment.

b) HoloLens 2: The Microsoft Hololens 2 is an example for such a consumer-oriented device that supports eye-tracking, in order to better understand the users intent and allow for implicit actions, like selecting objects and interacting with the world only by gaze [19]. This furthermore enables for attention tracking and gaze based scrolling. The accuracy of gaze data collected by the Hololens 2 is within 1.5° in visual angle after calibration [19]. A feature called spatial mapping creates and caches a mesh of the room around the user, which can be used for 3-dimensional reconstruction in eye-tracking experiments [11]. The mesh created by this process was measured to be precise up to 2.25cm [8].

Fig. 4. 3D mesh automatically created by Hololens with spatial mapping [8]

c) Vive Pro Eye: The HTC Vive Pro Eye also has eye-tracking features, with a gaze data accuracy of around 4.16° [18]. HTC uses this information for a feature called foveated rendering, in which the display within the headset only renders the part around the fovea, at the center of the users gaze direction, with high detail, while leaving the areas that the user is not looking at in lower resolution [3]. This increases the performance and of the device in the rendering pipeline, as less detailed information needs to be calculated each frame, resulting in a higher framerate and less battery consumption for wireless usage.

6) Conclusion: In this section we saw an overview and some techniques to process eye-tracking data and map collected gaze coordinates towards a 3-dimensional scene and how to visualize the collected data in such an environment. This includes various ways of tracking the headsets positional and angular movement through space, using raytracing to determine 3-dimensional coordinates from gaze directions and rendering gaze paths, point clouds or heatmaps into the scene. Furthermore, we explored different methods and hardware that can be utilized to reconstruct such a 3-dimensional scene from data collected in the real world, like reconstructing from a 2-dimensional video feed, a video with depth information, utilizing an additional video feed or using a pre-existing virtual model of the scene.

Many advances have been made in the recent years towards collecting more accurate data from the real world, using cheaper yet reliable hardware. This not only makes eye-tracking more accessible, but also facilitates research in 3-dimensional scene reconstruction and better real-time rendering of visualizations. Advances in machine learning assist in easy, increasingly robust and accurate 3-dimensional reconstruction from video data. And capabilities of newer hardware, especially in the virtual and augmented reality sector, make this type of studies possible with consumer devices.

References

[1] Matthias Bernhard, Efstathios Stavrakis, Michael Hecher, and Michael Wimmer. Gaze-to-object mapping during visual search in 3d virtual environments. *ACM Transactions on Applied Perception (TAP)*, 11(3):1–17, 2014.

[2] Sean Buckley. This is how valve's amazing lighthouse tracking technology works, May 2015.

[3] HTC Corporation. Vive pro eye features - vive european union, 2021.

[4] Hugh Durrant-Whyte and Tim Bailey. Simultaneous localization and mapping: part i. *IEEE robotics & automation magazine*, 13(2):99–110, 2006.

[5] Bing Han, Christopher Paulson, and Dapeng Wu. 3d dense reconstruction from 2d video sequence via 3d geometric segmentation. *Journal of Visual Communication and Image Representation*, 22(5):421–431, 2011.

[6] Hui Huang, Dan Li, Hao Zhang, Uri Ascher, and Daniel Cohen-Or. Consolidation of unorganized point clouds for surface reconstruction. *ACM transactions on graphics (TOG)*, 28(5):1–7, 2009.

[7] Philipp Jenke, Michael Wand, Martin Bokeloh, Andreas Schilling, and Wolfgang Straßer. Bayesian point cloud reconstruction. In *Computer Graphics Forum*, volume 25, pages 379–388. Wiley Online Library, 2006.

[8] K Khoshelham, H Tran, and D Acharya. Indoor mapping eyewear: geometric evaluation of spatial mapping capability of hololens. 2019.

[9] Mickey Li, Noyan Songur, Pavel Orlov, Stefan Leutenegger, and A Aldo Faisal. Towards an embodied semantic fovea: Semantic 3d scene reconstruction from ego-centric eye-tracker videos. *arXiv preprint arXiv:1807.10561*, 2018.

[10] Ting-Hao Li, Hiromasa Suzuki, and Yutaka Ohtake. Visualization of user's attention on objects in 3d environment using only eye tracking glasses. *Journal of Computational Design and Engineering*, 7(2):228–237, 2020.

[11] Mattzmsft. Spatial mapping, Mar 2018.

[12] Michael Maurus, Jan Hendrik Hammer, and Jürgen Beyerer. Realistic heatmap visualization for interactive analysis of 3d gaze data. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, pages 295–298, 2014.

[13] Annette Mossel and Manuel Kroeter. Streaming and exploration of dynamically changing dense 3d reconstructions in immersive virtual reality. In *2016 IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct)*, pages 43–48. IEEE, 2016.

[14] Zak Murez, Tarrence van As, James Bartolozzi, Ayan Sinha, Vijay Badrinarayanan, and Andrew Rabinovich. Atlas: End-to-end 3d scene reconstruction from posed images. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*, pages 414–431. Springer, 2020.

[15] Irina Nurutdinova and Andrew Fitzgibbon. Towards pointless structure from motion: 3d reconstruction and camera parameters from general 3d curves. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2363–2371, 2015.

[16] Thies Pfeiffer and Cem Memili. Gpu-accelerated attention map generation for dynamic 3d scenes. In *2015 IEEE Virtual Reality (VR)*, pages 257–258. IEEE, 2015.

[17] Thies Pfeiffer and Patrick Renner. Eyesee3d: a low-cost approach for analyzing mobile 3d eye tracking data using computer vision and augmented reality technology. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, pages 195–202, 2014.

[18] Alexandra Sipatchin, Siegfried Wahl, and Katharina Rifai. Accuracy and precision of the htc vive pro eye tracking in head-restrained and head-free conditions. *Investigative Ophthalmology & Visual Science*, 61(7):5071–5071, 2020.

[19] Sostel. Eye tracking on hololens 2, Oct 2019.

[20] Sophie Stellmach, Lennart Nacke, and Raimund Dachselt. Advanced gaze visualizations for three-dimensional virtual environments. In *Proceedings of the 2010 symposium on eye-tracking research & Applications*, pages 109–112, 2010.

[21] Haofei Wang, Jimin Pi, Tong Qin, Shaojie Shen, and Bertram E Shi. Slam-based localization of 3d gaze using a mobile eye tracker. In *Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications*, pages 1–5, 2018.