Visualizing and Clustering Eye Tracking within 3D Virtual Environments

Brent Chamberlain[1], Scott Johnson[2], Charisse Spencer[2], David Evans[2], Phillip Fernberg[2], Emily Tighe[3], Morgan LaFavers[3], Sarah Creem-Regehr[3], Jeanine Stefanucci[3]

[1]Utah State University, Utah/USA · brent.chamberlain@usu.edu
[2]Utah State University, Utah/USA
[3]University of Utah, Utah/USA

Abstract: Visual perception is one of the most important sensory processes for most of the population. This process plays a key role in how we navigate and way find in urban environments. A wide range of literature offers insight into the relationship between the structure of urban spaces and navigability, as well as literature identifying how individual differences play a role in how well people can recall elements and navigate environments. Measurement techniques that reveal these differences are often captured as procedurally based evaluations after individuals have navigated through an environment. However, these valuations do not necessarily help us understand the process of how observations link to recall and navigation. In this paper, we show a new technique for conducting eye tracking in 3D virtual environments to assess the process of observation in urban environments. Further, we demonstrate how clustering techniques can be used to improve eye tracking data generated in these 3D environments. The techniques we provide can offer a new means to better understand how form, function, and design elements are observed.

Keywords: Landscape architecture, urban design, eye tracking, Machine Learning, Unity, navigation

1 Introduction

In navigation, the visual perception of an environment plays a significant role in decision-making, as well as informs knowledge about the properties of spaces and the relationships of objects that form the collective environment. One of the ways researchers have attempted to understand the decisions we make during navigation is to create highly controlled experiments using virtual environments (BRUNS & CHAMBERLAIN 2019). However, many of the previous experiments lack a comparable diversity of objects, routes, scales, and relationships between these elements in comparison to the real world. The benefit of virtual environments, particularly with modern gaming engines, is that the designer can control all elements within the environment. This offers a means to simplify a problem and employ a more deductive scientific process, but it may come at the cost of understanding how perception works holistically within complex spaces. Unfortunately, quantifying perception holistically would require new methods for analyzing the process of perception. Further, a method like this would need to be implemented in complex environments, which can be self-defeating if it requires an oversimplification of the environment itself to operate. Thus, finding a technique that enables both the employment of complex virtual environments and a seamless integration of analyzing perception holistically would be a major step in understanding the relationship between human and environmental interactions.

Designing spaces for intuitive navigation is an important process for urban designers, campus (e. g. business parks) planners, and outdoor recreation trail designers. There are many design problems to undertake in these instances, with navigation and wayfinding within the set of

Journal of Digital Landscape Architecture, 8-2023, pp. 318-325. © Wichmann Verlag, VDE VERLAG GMBH · Berlin · Offenbach. ISBN 978-3-87907-740-3, ISSN 2367-4253, e-ISSN 2511-624X, doi:10.14627/537740034.

issues. Both these processes require individuals to recognize spatial patterns, comprehend relationships of elements, make determinations of how to focus their attention, and remember important objects or spaces. There have been many approaches to assessing these processes, such as measuring response times and accuracy in remembering landmarks or locations (CHRASTIL & WARREN 2015, ERICSON & WARREN 2020, GAGNON et al. 2018, WEISBERG et al. 2014) and assessing map drawings of paths and spatial layout (BRUNS & CHAMBERLAIN 2019, GARDONY et al. 2016, WANG & SCHWERING 2015). However, these measures are usually done *post hoc* rather than in real time. Further, the measures are usually procedurally based (e. g., memory recall), rather than processes based (formation of memories). To improve our understanding of how process-based navigational activities unfold, we need to understand what drives perception and decision-making. A better understanding could help designers support meaningful relationships between objects to facilitate these perceptual processes. Fortunately, computational techniques can be created and then combined with cognitive science and urban and landscape design principles to better understand how individuals observe and make inferences about those spaces.

Eye tracking is one technique that has been used by researchers to better understand the process of observation. It has been used for decades to understand how and why an individual focuses on particular objects, areas, and elements of space. Implementations of eye tracking have been primarily conducted in 2D environments (e. g. looking at a screen or flat image). This includes architectural-related studies (XIANGMIN et al. 2021, ZHANG et al. 2019), with landscape studies emphasizing 2D static images (DUPONT et al. 2016). In landscape and architectural studies, eye tracking is used to identify fixations within scenes, and in psychology can help describe visual attention and arousal (KIM & LEE 2021). With many metrics that can be analyzed from these data, broadly, one major advantage is to provide an objective measure of perception (DUPONT et al. 2014). Yet, relative to 2D eye tracking studies, there is little literature showcasing implementation in 3D dynamic environments.

In this study, we combine the generation of virtual environments and eye tracking to visualize individual observational patterns in a virtual space. We extend previous work (FERNBERG et al. 2022) to showcase a new open-source software package we developed, as well as an analytical framework for representing these data. This paper deviates from the previous by showing specific visualizations and analyses of large datasets that have been analyzed, whereas the previous version was introducing the construct. The purpose of this study is to showcase how eye tracking data can be represented in a dynamic 3D environment and how those data can be analyzed using mathematical clustering mechanisms. We ask, to what extent can eye tracking be implemented in 3D gaming environments and analyzed post-hoc to determine fixations of objects within virtual urban spaces?

2 Data Collection from the Virtual Environment

For this work, we employ the Unity gaming engine and prefabricated 3D assets (from Kitbash) to create a procedurally generated urban environment which can be explored in VR. The design of the environment was not intended to be complex or realistic world because this is not necessary for the primary purpose of implementing eye tracking and testing different clustering techniques. The environment consisted of 40 x 100-meter-long blocks, where each block was one of five different architectural styles. The purpose of this setup was to observe

if the pattern of clustering was different closer to transitions between different architectural styles compared to areas where changes did not exist. However, in this study, we were merely attempting to identify how we might cluster data, the actual test of these transition zones is meant for a later study. For now, all elements are static, the user can make observations freely, but cannot change their location or speed of experience. Further, we have not included any other cues, such as sound, lights, or atmospheric changes. Each object was placed along a two-lane road in succession, with variable spacing between each building.

Fig. 1: Perspective views of the virtual environment in Unity as seen by study participants

The eye tracking software we developed, was created for implementation in the Unity gaming engine only. For this implementation, the Vive Eye Pro virtual reality headset was used. Within Unity, the headset was established as the user camera and the scripts were then associated with this camera. Movement through the environment was maintained at a consistent speed, but the viewer can fully move their head around. The eye tracking software stores the location and rotation of the user's eyes at every frame that gets rendered (about 60/sec). The data from each eye is averaged to create one point and one direction. This direction is, of course, where the user is looking. Using this information, we create an invisible virtual ray that extends from the eye outwards (see Figure 2). Once this ray hits an object, the collision point (location of the intersection of the eye tracking ray and the surface of the object) is recorded along with the object's name and position (recorded for accurate reproduction of the data before participation). Other metrics are recorded and computed such as eye angle (looking left/right/etc.), distance from eye to collision, and whether they are blinking or not. The figure below is a representation of rays produced along the route as a user looks at objects on the buildings' surfaces.

The data produced from a single experiment can result in a substantial number of data points collected. With such a vast amount of data being produced, it is important to identify the most relevant data that could provide researchers with meaningful interpretations of observational patterns. So, we needed to identify ways to reduce the amount of data by reducing noise and jitter. Then, we needed to cluster the remaining data into meaningful groups, referred to as fixations. From this data, we can determine the total dwell time and total fixation count for specific areas of interest. The areas of interest are regions in the environment that are important (HOLMQVIST et al. 2011) and could be identified as specific objects of general areas along an object.

Fig. 2: Example of perspective views of the virtual environment in Unity. Here, rays are shown while looking at specific targets to demonstrate both clustering of data and the ray from the observer to the surface of an object. Targets are shown here merely for demonstration and were not visible in this experiment.

To produce clusters from denoised data, we tested a clustering method called DBSCAN. DBSCAN stands for Density-Based Spatial Clustering of Applications with Noise. This technique uses an unsupervised machine-learning algorithm to identify clusters of observations. As the name implies, it uses the spatial density of the data points (in any number of dimensions) to create clusters and eliminate noise. One important function of this algorithm is that you can use more than the 3-dimensional distance to find spatial density. It can include factors such as eye rotation and time in its calculations. This can be useful because in a dynamic environment participants can first look at an object in the distance, then as they move forward through an environment, look back at the object again. This dynamic facet of 3D gamine environments makes it critically important to ascertain what is a fixation across distances, versus random noise that could have been part of a rapid eye movement across an area and along an object.

3 Implementation and Outcomes

In this section we highlight the results from the clustering technique to show: 1) the volume of data produced by a single participant, 2) the observational patterns of the participant, and 3) the effects of implementing the clustering algorithm on the previous two. In this section, we provide context to the implementation (for each of these three), takeaways from our experience, and general statistics to highlight an overview of the outcomes.

In our experiment, a single individual produced 39404 observations over the entire 9 minutes and 35 seconds of the experience. This averaged about sixty-eight observations per second.

The rate of eye tracking data collection depends upon Unity's internal update function, which is the same as the framerate. Framerate is affected by how many objects are in view and need to be processed by the GPU. Therefore, the framerate can vary throughout the experiment. While it can be helpful to maintain a high-frequency rate to minimize motion sickness and improve realism, it is unknown the extent to which the rate of data collection would impact the results, for whatever results are being sought.

Using these data, we developed a simple metric to highlight how often an individual may look at objects versus other elements in the environment. In our implementation, the objects were buildings, and the other elements included the ground (terrain), the street, and the sky. In our implementation, the ground and street are objects because they have a surface with a collider that enables the collection of eye tracking data points when the vector of the observation intersects with that surface. Unlike buildings, these two surfaces are continuous throughout the entire environment, whereas buildings are separate objects. Our eye tracking also indirectly collected observations of the sky (or distant void), in which there is a frame with no distinct object with a collision surface. Figure 3 shows global statistics of the proportion of observations made between the sky, terrain, road, and buildings. The figure also compares those data after removing saccades. Saccades and their removal are explained below.

Eye tracking data can be difficult to interpret. This is particularly true in 3D, where there are very few studies that have attempted to validate how observational patterns in 3D are associated with meaningful outcomes of navigation (UGWITZ et al. 2022). One crucial step in generating interpretations of data is to remove irrelevant data (e. g., noise), such as saccades. A saccade is essentially a rapid view of an object, then a focus away from that object with another quick return to the original object. In terms of perception, little to no information is gleaned during a saccade. Therefore, in addition to general noise (single random observation in space), eliminating saccades can also help streamline the data analysis.

However, identifying these saccadic movements requires playing with the clustering parameters. This is because eye tracking data is generated based on a collision point for each frame, but how the algorithm determines if a single data point belongs to a cluster or not is a little tricky. To reduce noise and eliminate saccades, we implemented DBSCAN. DBSCAN takes the {X, Y, Z} vector position where it collided with the object, but also the time dimension of when it collided. Clusters are determined by locational and temporal similarities of vectors by turning two parameters. First, *epsilon* is the distance threshold from one observation to another (in 4D). Second, *minimum points* is the number of minimum points that constitute a cluster. Our parameters were an *epsilon* of two meters and a *minimum points* of seven, which represents roughly a tenth of a second or the approximate minimum amount of time for a fixation. Again, Figure 3 highlights the global statistics before and after the removal of saccades, or points that did not belong to a cluster.

Figure 4 further depicts an example of different clusters using different colored dots. In this figure, similar colored dots represent a single observational point. Find the set of green dots right below the purple dots. The large cluster of green dots shares a similar proximity with a single black dot right on the roof ridge. This black dot seems to be part of that green dot cluster. However, DBSCAN identified that the observation point represented as the black dot should not belong to a cluster. This was because the observational point was created several seconds prior as part of an earlier saccade (singular rapid observation), whereas the other observations were made in sequence (suggesting a focal point or area).

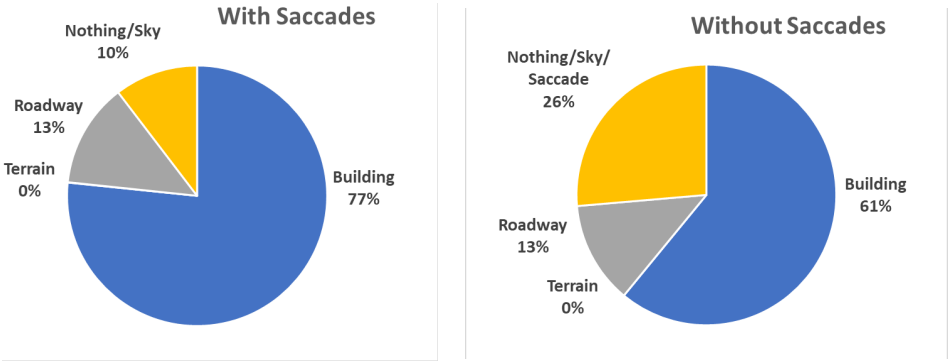

Fig. 3: Chart showing the distribution of observations by category (sky, terrain, road, and buildings). The chart shows both the original data (With Saccades) and data after removing Saccades.

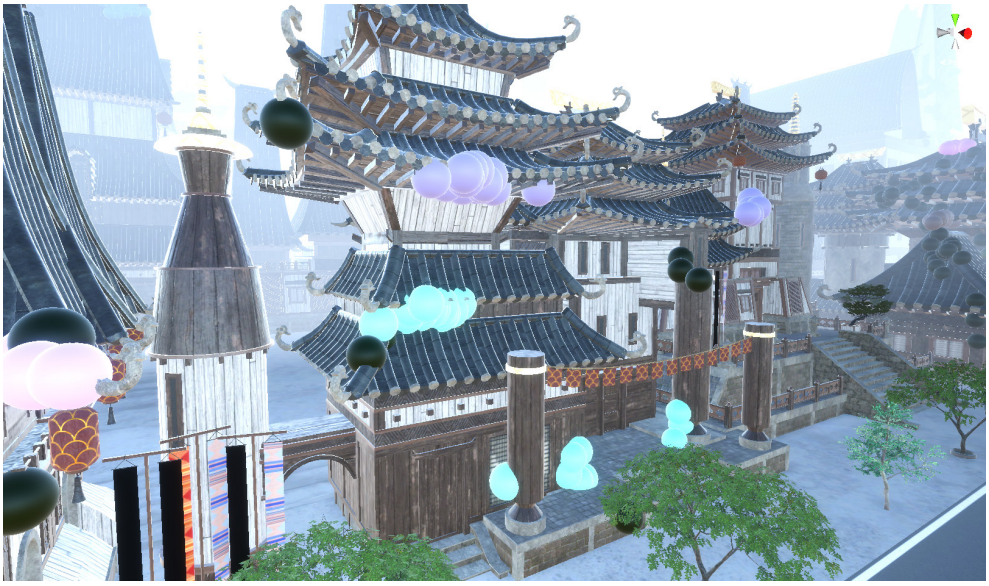

Fig. 4: Depicts clusters (colors) and saccades (black). Image brightness has been increased and the rendering is done from an elevated perspective, both to improve representation for the demonstration of these data.

4 Discussion and Conclusion

3D gaming platforms offer the ability to produce vast amounts of user-centric data. Having tools to analyze these data can help designers identify environmental cues or triggers that could influence the perception of a design or plan. Eye-tracking offers a non-intrusive, process-based technique for collecting very precise observations within a space. However, finding a robust algorithm to cluster thousands of eye data points is essential to making mean-

ingful interpretations of these perceptions. Using the software we produced, these patterns can be visualized and reduced for making assessments about areas or objects favored by users. DBSCAN is one of several techniques available but has been shown to produce good results (ESTER et al. 1996). This paper was not intended to conduct a systematic comparison across these techniques and variations, but instead to demonstrate the potential for eye tracking data in combination with a clustering technique to produce useful data.

The next major step in this research is to understand how these data can be related to meaningful observations to help form decision-making and recall. Understanding this link can help designers better associate the placement and patterns of objects, such as landmarks (BRUNS & CHAMBERLAIN 2019), within the environment to improve wayfinding and navigation. As Thus, some next research questions are: to what extent do eye tracking observations in a dynamic 3D virtual environment correlate with memory recall, navigational decisions, and pattern recognition about the overall design of the environment? More broadly, to what extent can eye tracking help us understand how individuals form mental maps? In our experience, we noticed several situations where individuals were following unique building features, peering through passageways, and scanning the topography of buildings. While these observations are anecdotal and with limited data, they do suggest these data could help validate the importance of or focus on different architectural forms, textures, and aesthetics.

Implementing eye tracking in 3D-controlled virtual environments shows promise for aiding the examination of observational processes. This will have relevance in multiple fields. Certainly, eye-tracking has been used in 3D gaming, but studies in psychology, architecture, urban design, interior design, and landscape architecture could benefit from having access to individual patterns of observation data. Eye tracking is a well-established technique but employing it within 3D environments and determining how to associate these data with meaningful interpretations will provide new opportunities and insights for landscape studies.

Acknowledgments

This work was funded by the US Army Research Institute for Social and Behavioural Sciences, award W911NF2010291. The views, opinions, and/or findings contained in this paper are those of the authors and shall not be construed as an official Department of the Army position, policy, or decision unless so designated by other documents.

References

BRUNS, C. R. & CHAMBERLAIN, B. C. (2019), The influence of landmarks and urban form on cognitive maps using virtual reality. Landscape and Urban Planning, 189, 296-306. https://doi.org/10.1016/j.landurbplan.2019.05.006.

CHRASTIL, E. R. & WARREN, W. H. (2015), Active and passive spatial learning in human navigation: Acquisition of graph knowledge. Journal of Experimental Psychology: Learning, Memory, and Cognition, 41 (4) 1162-1178. http://dx.doi.org.er.lib.k-state.edu/10.1037/xlm0000082.

DUPONT, L., ANTROP, M. & VAN EETVELDE, V. (2014), Eye-tracking Analysis in Landscape Perception Research: Influence of Photograph Properties and Landscape Characteristics. Landscape Research, 39 (4), 417-432. https://doi.org/10.1080/01426397.2013.773966.

DUPONT, L., OOMS, K., ANTROP, M. & VAN EETVELDE, V. (2016), Comparing saliency maps and eye-tracking focus maps: The potential use in visual impact assessment based on landscape photographs. Landscape and Urban Planning, 148, 17-26. https://doi.org/10.1016/j.landurbplan.2015.12.007.

ERICSON, J. D. & WARREN, W. H. (2020), Probing the invariant structure of spatial knowledge: Support for the cognitive graph hypothesis. Cognition, 200, 104276.

ESTER, M., KRIEGEL, H.-P. & XU, X. (1996), A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. Kdd, 96 (34), 226-231.

FERNBERG, P., TIGHE, E., LAFAVERS, M., SPENCER, C., STEFANUCCI, J., CREEM-REGEHR, S. & CHAMBERLAIN, B. (2022), Measuring Perception of Urban Design Elements in Virtual Environments Using Eye Tracking: Benefits and Challenges. Journal of Digital Landscape Architecture, 7-2022, 463-470. https://doi.org/10.14627/537724045.

GAGNON, K. T., THOMAS, B. J., MUNION, A., CREEM-REGEHR, S. H., CASHDAN, E. A. & STEFANUCCI, J. K. (2018), Not all those who wander are lost: Spatial exploration patterns and their relationship to gender and spatial memory. Cognition, 180, 108-117. https://doi.org/10.1016/j.cognition.2018.06.020.

GARDONY, A. L., TAYLOR, H. A. & BRUNYÉ, T. T. (2016), Gardony Map Drawing Analyzer: Software for quantitative analysis of sketch maps. Behavior Research Methods, 48 (1), 151-177. https://doi.org/10.3758/s13428-014-0556-x.

HOLMQVIST, K., NYSTRÖM, M., ANDERSSON, R., DEWHURST, R., JARODZKA, H. & WEIJER, J. van de (2011), Eye Tracking: A comprehensive guide to methods and measures. OUP Oxford.

KIM, N. & LEE, H. (2021), Assessing Consumer Attention and Arousal Using Eye-Tracking Technology in Virtual Retail Environment. Frontiers in Psychology, 12, 2861. https://doi.org/10.3389/fpsyg.2021.665658.

UGWITZ, P., KVARDA, O., JUŘÍKOVÁ, Z., ŠAŠINKA, Č. & TAMM, S. (2022), Eye-Tracking in Interactive Virtual Environments: Implementation and Evaluation. Applied Sciences, 12 (3), 1027.

WANG, J. & SCHWERING, A. (2015), Invariant spatial information in sketch maps – a study of survey sketch maps of urban areas. Journal of Spatial Information Science, 11, 31-52.

WEISBERG, S. M., SCHINAZI, V. R., NEWCOMBE, N. S., SHIPLEY, T. F. & EPSTEIN, R. A. (2014), Variations in cognitive maps: understanding individual differences in navigation. Journal of Experimental Psychology: Learning, Memory, and Cognition, 40 (3), 669.

XIANGMIN, G., WEIQIANG, C., TIANTIAN, L. & SHUMENG, H. (2021), Research on dynamic visual attraction evaluation method of commercial street based on eye movement perception. Journal of Asian Architecture and Building Engineering, 21 (5), 1779-1791. https://doi.org/10.1080/13467581.2021.1944872.

ZHANG, L.-M., ZHANG, R.-X., JENG, T.-S. & ZENG, Z.-Y. (2019), Cityscape protection using VR and eye tracking technology. Journal of Visual Communication and Image Representation, 64, 102639. https://doi.org/10.1016/j.jvcir.2019.102639.