Analysis of eye tracking data in 3D virtual environments

Author
DUAN ZHIHONG

Supervisor
Toinon Vigier,Matthieu Perreira Da Silva

UNIVERSITE DE NANTES

polytech nantes, ls2n

FRANCE NANTES

July 2, 2019

Contents

1 Introduction

In the eye tracking domain, we use eye tracker to collect eye movements data first, the next step is analysis. We can measure the difference of eye-tracking data between different people, then we can get a conclusion of how big is the gap between different people. This can help us understand how does human vision system works. Beside, we can also measure the gap between the ground truth (human eye-tracking data collected by eye tracker) and the artificial data (the region of interest predicted by vision attention model). This can tell us if the attention model works well, which aspect it is insufficient and how can we improve it.
The tools and techniques for comparing eye-tracking data in 2D such as the visualization of images or video on computer displays or TVs are numerous and well-known (detection of saccades and fixations, correlation, ROC analysis, etc.), but their equivalents for data acquired in a virtual 3D world are very limited for the moment. However, the virtual reality (VR) and augment reality (AR) becomes increasingly popular these years. The many recently released headsets (HTC Vive, Facebook Oculus, Sony Playstation VR, Google Daydream) are a clear sign of the democratization of this technology. One of the next technological innovations in the field of virtual reality is the embedding of eye-trackers in VR helmets. These devices can be of great use both to directly optimize the quality and the comfort of the rendering but also in the context of experimental research to understand the visual strategies of users.
Our research study on the concepts and methods for comparing eye-tracking data acquired while interacting with an environment in virtual reality. We acquire real eye-tracking data in an existing virtual environment (ex: ReViSTIM environment, built for acrophobia study) and then proposed an evaluate algorithms for comparing eye-tracking data in such an environment. The report is structured as follows. In the section 2 will given the background of the filed of eye tracking. It was divided into three parts, they are answered the question of what is eye tracking, how does eye tracker work, and what is virtual reality and augmented reality respectively. In the section 3, we briefly discussed the related works other researchers have done, but it is only about 2 dimensional related works. The related works in 3 dimensions will be given in section 4. In section 5 we proposed a new method to deal with the drawbacks present method existed. And in the section 6, we will give a conclusion for all the methods had mentioned, and in the last section we will analyze the drawbacks of the present method and give an outlook for the future works.

2 Background

Eye tracking is kind of technique that can record the track of human eye's movement through eye tracker device. Eye movements are the trajectories of eye's motion. It contains sorts of information, e.g. where do we focus on(at present), what do we ignore, and when do we blink our eye and etc. With these information, we can get a complete understanding of how does human's eyes motion. If researchers can get these eye tracking data and then they can analyze these data and built a model to predict the human eyes movement in a particular situation (in a given image or video in 2D or 3D environment). Our task here is to analyze eye tracking data in 3D virtual environment, judging how different two eye tracking data are. There are lots of measures can be used in 2D. However, since the difference between 3D and 2D, these measures can not be used directly in 3D. This section will be consisted of four parts, they respectively are what is eye tracking? How does eye tracker work? What is virtual reality (VR) and augmented reality (AR)? And why the measures in 2D can not be directly used in 3D? Through this section, you will get a better understanding of the eye tracking analysis and the problem we faced.

2.1 What is eye tracking?

"Eye tracking is the process of measuring either the point of gaze (where observer is looking) or the motion of an eye relative to the head"[20]. An eye tracker is a device for measuring eye positions and eye movement. Eye trackers are used in research on the visual system, in psychology, in psycho linguistics, marketing, as an input device for human-computer interaction, and in product design. There are a number of methods for measuring eye movement. "The most popular variant uses video images from which the eye position is extracted. Other methods use search coils or are based on the electrooculogram".[18] Below is an example of eye tracking (Figure 1.)

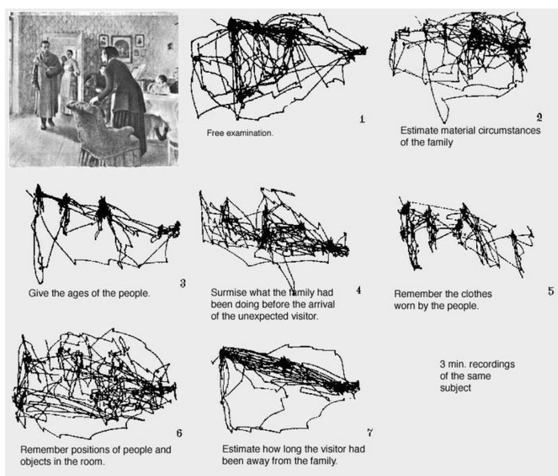

Figure 1: Eye tracking [23]

2.2 How does eye tracker work?

Eye tracker is a device used for eye tracking. Generally, there are two types of eye tracker, Screen-based(also called remote or desktop) and glasses(also called mobile). There is a infrared light from eye tracker and a camera will record the corneal reflection. The infrared light is directed towards the center of the eyes(pupil), causing detectable reflections in both the pupil and the cornea(the outer-most optical element of

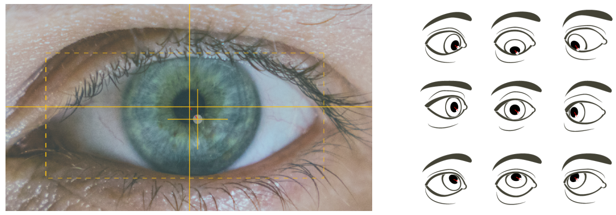

Figure 2: The operating principle of Eye tracker [8]

the eye). These reflections - the vector between the cornea and the pupil - are tracked by an infrared camera. This is the optical tracking of corneal reflections, known as pupil center corneal reflection(PCCR).
An infrared light source (and thus detection method) is necessary as the accuracy of gaze direction measurement is dependent on a clear demarcation (and detection) of the pupil as well as the detection of corneal reflection. Normal light sources (with ordinary cameras) aren't able to provide as much contrast, meaning that an appropriate amount of accuracy is much harder to achieve without infrared light. From Figure 2. The center of the eye is tracked in relation to the position of the corneal reflection. The relative distance between the two areas allows the calculation of the direction of the gaze. And the relative difference in location of the pupil center and corneal reflection allows for deduction of the gaze direction.
During the experiment, the eye tracking device record gaze points performed by a participant on a stimulus. The measurement precision is about 0.5 degree, low and hardware starts at around 1.0 degree, medium 0.5 degree, and high end at 0.1 degree or even less. The recording rates depend on the characteristics of the devices. The devices allow rates between 30-60 Hz. Special research equipment records at around 120-1000 Hz[8]. The recording rate specifies how many gaze points are recorded per second. Figure 3 and Figure 4 are two demonstrations of screen-based tracker and glasses eye tracker.

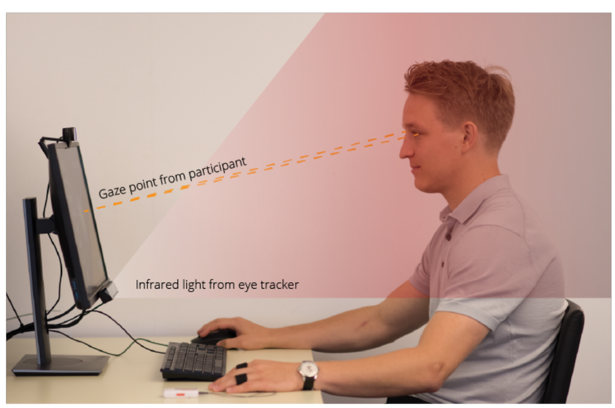

Figure 3: Screen-based eye tracker[8]

Figure 4: Glasses eye tracker[8]

2.3 What is Virtual Reality (VR) and Augmented Reality (AR)?

"Augmented Reality (AR) is an interactive experience of a real-world environment whereby the objects that reside in the real-world are "augmented" by computer-generated perceptual information" [17], sometimes across multiple sensory modalities, including visual, auditory, haptic (the ability "to grasp something"), somatosensory, and olfactory. In other words, AR adds digital elements to a live view often by using the camera on a smartphone in some applications like Pokémon Go, or to a head mounted display (e.g.: Microsoft Hololens). Virtual Reality implies a complete immersion experience that shuts out the physical world [21]. Eg : HTC Vive Oculus Rift. The experience is taking place within a simulated environment, that incorporates mainly auditory and visual, but also other types of sensory feedback like haptic. This immersive environment can be similar to the real world or it can be fantastical, creating an experience that is not possible in ordinary physical reality.

2.4 Why the measures in 2D can not be directly used in 3D?

Why the measures in 2D can not be directly used in 3D? We will now enumerate the specificity of eye traces in 3-dimensional environments in order to make an inventory of the constraints.

2.4.1 Combination movements head, body and eyes

The first specificity that comes to mind when we talk about virtual reality visualization is the possibility even the need to move the head and the body to look at the entire environment. Indeed, unlike the visualization of a 2D image, the individual moving in a space in virtual reality will turn his head, and potentially move in space. There are therefore 3 dimensions influencing the capture of eye paths:

1. The position of the eye, and more precisely of the pupil (as for 2D)
2. The position of the head in the frame of reference of the body

Figure 5: Eye tracker [12][6]

3. The position of the body in the repository of the physical space

The data collected by eye tracker between 2D and 3D is different. The position of the observer is not considered in 2D since the user visualizes something fixed, there are only two dimensions. On the other hand, this data enters into account in the 3D. In addition, in 3D, we have the information of the point looked at by the user in three coordinates (against 2 in coordinates in 2D). It is necessary to take into account the height in the space.

2.4.2 Difference of point of view

Unlike the simple visualization of 2-dimensional images where the individual does not move, virtual reality environments are designed for the user to move, or at least to turn his head. This gives rise to several factors that influence and complicate the comparison of data from 3D stimuli.
The first difference is perception of an object according to its different sides. In virtual reality, users have the opportunity to turn around an object, or move to see it from different points of view. Because of this, the vector can vary greatly between two users even though they have looked at the same object: they have looked at it from a different point of view. Figure 6 illustrates this case. This has a real influence on the comparison possible between the coordinates of the looks. Indeed, if two individuals look at the same object but from points of opposite view, they will not necessarily see the same thing of this object, and yet the Euclidean distance between two positions looked at will be weak, which may leave believe that both individuals have seen the object of the same point of view.

The second question in 3 dimensions is the Occlusion of objects. Also, we must take into account the possible occlusions of objects: one object can hide another. It is possible to have a virtual environment very provided in terms of objects. As a result, a user A can potentially see an object that another user B cannot see, because of their

Figure 6: Image representing two individuals viewing an object from two different points of view [13]

different positions (and thus points of view) in the space: the object is hidden by another object for user B. So, depending on the position in the space and the angle of view, objects will be visible or not. To mitigate these eventualities, some environments in virtual reality propose to apply a transparency to the object looked at by the user, so that it can see the potential objects behind.

The last factor we should consider is the perception of an object according to the viewing distance. As previously stated, it is possible to move in a virtual environment. This implies that different individuals will potentially look at the same things but at more or less reduced distances. The further the user is from the object he is looking at, the larger his visual field will be, as we can see in Figure 7. A user who looks at an object from a distance will not see the same things that a user who looks at its front, since his visual field (the circle projected on the figure) will be bigger. This has an influence on the relevance to make the comparison of these locations of the gaze.

2.4.3 Distinction between foreground and backplane

The interaction of an individual in a virtual reality setting involves viewing a 3-dimensional environment. This means that the individual evolves in an environment in which he can see objects on different planes, from the foreground to the background. This poses difficulties in the location of the user's gaze: the distinction between a look at the edge of an object in the foreground and a look at an object in the background is very fine. Figure 8 illustrates this case: the green and red arrows symbolize the direction of gaze of the Actor individual. We notice that it is difficult to know if the individual is looking at the corner of the chair or the vacuum cleaner.

3 Related works in 2D

Even though lots of difference between 2 dimensional and 3 dimensional environment, it is necessary to understand how does the comparison methods work in 2D. This

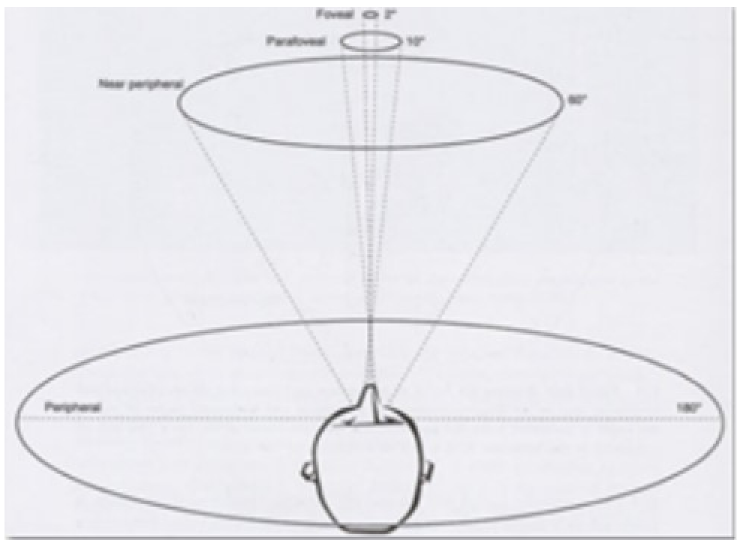

Figure 7: Diagram of the projection of the visual field of an individual

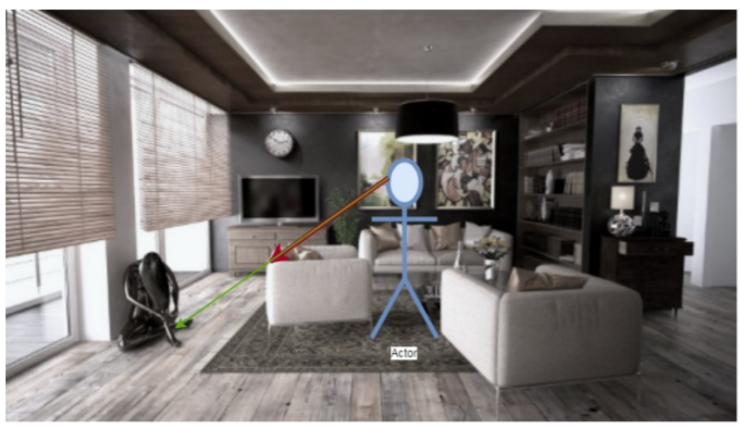

Figure 8: Image illustrating the difficulty of distinguishing a look on an object in the foreground from a look at an object in the background

can inspire our thinking about the 3D environment. In this part we will give a brief introduction of some works other researchers have done in 2D, the main reference is "The comparison of scanpath comparison methods", which is written by Nicola etc [14] . And the methods in 2D will still be categorized into four types based on their respective characteristics, they are *Grid-based*, *Sample-based*, *Direct measure*, and *Recurrence-based* measures. After reading this chapter, you will have a global view of the various methods in 2D.

3.1 Grid-based

This kind of methods will divided a whole image into tones of grids, and each grid is assigned a unique character. The *Edit Distance*and *ScanMatch* are belong to this kind of method.

3.1.1 Edit Distance

Definition: edit distance is a way of quantifying how dissimilar two strings are to one another by counting the minimum number of operations required to transform one string into the other [4]. Example: the *Levenshtein distance* between “defeat” and “default” is 3.

1. defeat $\rightarrow$ defat (deletion of ”e”)
2. defat $\rightarrow$ defaut (insertion of ”u” behind ”a”)
3. defaut $\rightarrow$ default (insertion of ”l” in frond of t).

The question is how can we translate scanpath into string? To achieve this, a grid is overlaid on an image, and each cell in the grid is assigned a unique character.

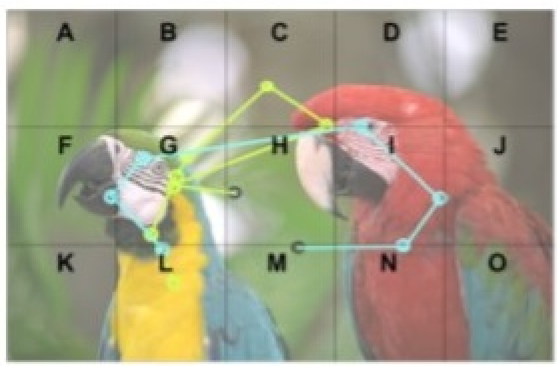

Figure 9: Principle of wearable eye tracker

3.1.2 ScanMatch

This approach was modified from a bioinformatics algorithm (Needlwman-Wunsch algorithm[2]) which used to compare DAN sequences, it is a new way to compare saccadic eye movement sequences [3]. And this method is composed with five steps below.

1. The data needs to be filtered down to saccades and fixations.
2. Image have to be divided into RoIs (Region of interests), a letter is then assigned to each region and every eye Fixation within that region is tagged with its name.
3. Introduce temporal binning, in a way that is proportional to the fixation duration.

4. Creating a matrix with all scoring possibilities based on a substitution matrix and a gap penalty.

5. Seeking the optimal alignment, tracing back the matrix from the top left corner to the outmost column or row by selecting the optimal route.

The main advantage of the string-edit measure lies in the fact that it captures the intuitive notion of scanpath distance in a simple way. However, there is not a guideline telling us how to define the grids. And the definition is time-consuming and is independently of image content and may be too coarse in regions of interest while being too fine in other regions [14]. Besides, two fixations may be considered different even when they are close together, namely if they fall on either side of a grid line.

3.2 Sample-based

Sample-based is kind of method that the scanpath are first re-sampled uniformly in time (at a fixed frequency, e.g 60Hz), and truncated to the shorter length. *Fixation overlap*, *Temporal Correlation* and *Gaze shift* are three representative methods of them.

3.2.1 Fixation overlap

Defining the proportion of overlapping samples. Two samples (at time t) overlap if the Euclidean distance between two samples is less than a predefined threshold. Because of this reason, the fixation is extremely sensitive to differences in timing but is slightly less sensitive to position. However, the pre-defined radius threshold is arbitrary, the finally performance is very much depend on the pre-defined radius [15].

3.2.2 Temporal Correlation

The temporal correlation is defined as the average of the correlation between their x-coordinates and y-coordinates, respectively. It is very sensitive to temporal and spatial differences between the two scanpath and lead less robust to noisy data than other grid or radius measures [15]. Below is the formulation about how to calculate Temporal Correlation T_c.

$$T_c = \frac{corr(f_x, g_x) + corr(f_y, f_y)}{2} \tag{1}$$

3.2.3 Gaze shift

It is computed in the same manner as the temporal correlation, but using the first derivative instead of the position [15]. The first derivative is computed by convolving each scanpath with the derivative of a Gaussian filter.

$$G = \frac{corr(|f_x^{'}|, |g_x^{'}|) + corr(|f_y^{'}|, |f_y^{'}|)}{2} \tag{2}$$

3.3 Direct Measures

These approaches does not need to be quantified as in the string-edit method, it simply compares each fixation in one scanpath with the fixations in another in terms of their spatial similarity. There are two measures here, they are *Linear Distance* and *Multi-Match* respectively [11].

3.3.1 Linear Distance

The linear distances between the fixation in the first scanpath and the nearest neighbor in the second scanpath, as well as the linear distances between the fixation in the second scanpath and the nearest neighbor in the first scanpath. The distance is averaged and normalized against randomly generated scanpath. It does not need to be quantified as in the string-edit method. However, by comparing only nearest neighbor fixation in terms of distance, this method ignores sequential information. Given two scanpath f and g with fixation n_1 and n_2, the similarity of the two scanpath is defined as below [11]:

$$S = (1 - \frac{D}{D_r}) \tag{3}$$

with

$$D^2 = \frac{n_1 \sum_{j=1}^{n_2} d_{2j}^2 + n_2 \sum_{i=1}^{n_1} d_{1i}^2}{2n_1 n_2 (w^2 + h^2)} \tag{4}$$

and w and h are the height and width of the image, D_r is the same as D but with randomly generated scanpaths.

3.3.2 Multi-Match

The Multi-Match methods consists of five separate measures that capture the similarity between different characteristics of scanpath. They are vector, direction, length, position, and duration respectively. And there are three steps in this kind of method, the three steps are Simplification, alignment and comparison. Next we will introduce those five measures and three steps separately [9].

Vector: the vector difference between aligned saccades pairs, normalized by the screen diagonal and averaged over scanpaths. It is sensitive to spatial differences in fixation positions without relying on pre-defined quantization.

Length: It is the absolute difference in the amplitude of aligned saccade vectors, normalized by the screen diagonal and averaged over scanpaths. it is sensitive to saccade amplitude only, not to the direction, location, or the duration of the fixations.

Direction: Direction similarity is computed as the angular difference between aligned saccades. normalized by π and averaged over scanpaths. This measure is sensitive to saccade direction only, but not to amplitude or absolute fixation location.

Position: It is computed as the Euclidean distances between aligned fixation. normalized by the screen diagonal, and averaged over scanpaths. This measure is sensitive to both saccade amplitudes and directions.

Duration: it is computed as the absolute difference in fixation duration of aligned fixations. normalized by the maximum duration and averaged over scanpaths. This measure is insensitive to fixation position or saccade amplitude.

3.4 Recurrence-based

This form of cross-recurrence analysis can provide an overall measure of similarity across two eye movement sequences. It need fixation sequences f and g have the same lengths. For sequences of unequal length, the longer sequence will be truncated [1].

3.4.1 Cross-recurrence

Consider two fixation sequences $f_i, i = 1, ...N, with f_i =< x_i, y_i >$ and $g_i, i = 1, ...N, with g_i =< x_i, y_i >$. For fixation sequences of unequal length, the long sequence is truncated. Two fixation f_i and g_j are cross-recurrent if they are close together, i.e., we define the cross-recurrence of two fixations c_{ij} as

$$c_{ij} = \begin{cases} 1, d(f_i, j_i) \leq p \\ 0, otherwise \end{cases} \tag{5}$$

where d is the Euclidian distance, and p is a given threshold.
Several measures can be used for characterizing cross-recurrence patterns. The measures are extensions of the recurrence measures introduced by Anderson et al.[]. Let C be the sum of recurrences, i.e., $C = \sum_{i=1}^{N} \sum_{j=1}^{N} c_{ij}$. Further, let D_L be the set of diagonal, H_L the set of horizontal, and V_L the set of vertica lines in the cross-recurrence matrix, all with a length of at least L, and let $|\cdot|$ denote cardinality
The cross-recurrence measure of two fixation sequence is defined as

$$REC = 100 \cdot \frac{C}{N^2} \tag{6}$$

It represents the percentage of cross-recurrent fixations, i.e., the percentage of fixations that match (are close) between the two fixation sequences.

3.4.2 Determinism

It measures the percentage of cross-recurrent points that form diagonal lines and represents the percentage of fixation trajectories common to both fixation sequences. That is, it quantifies the overlap of a specific sequence of fixations, preserving the sequential information. The minimum line length of diagonal line elements was set to L = 2.

$$DET = 100 \cdot \frac{|D_L|}{C} \tag{7}$$

3.4.3 Laminarity

Laminarity represents locations that were fixated in detail in one of the fixation sequences, but only fixated briefly in the other fixation sequence. Again, we set the minimum line lengths of vertical and horizontal lines to L = 2.

$$LAM = 100 \cdot \frac{|H_{L|} + |V_L|}{2C} \tag{8}$$

3.4.4 Corm

CROM is defined a as the distance of the center of gravity from the main diagonal, normalized such that the maximum possible value is 100.

$$CORM = 100\frac{\sum_{i=1}^{N}\sum_{j=1}^{N}(j-i)c_{ij}}{(N-1)C} \tag{9}$$

The CORM measure indicates the dominant lag of cross-recurrences. Small CORM values indicate that the same fixations in both fixation sequences tend to occur close in time, whereas large CORM values indicate that cross-recurrences tend to occur with either a large positive or negative lag.

3.5 Summary

In this part we make a table listing all the methods has mentioned above, and it's convenient for us to compare the advantages and disadvantages of different measures. For the Grid-based measures, it compares the scanpath is a simple way. However, how to define the region of interest is a trade-off problem. The Sample-based methods is sensitive to temporal. But it just like Grid-based methods, the pre-defined threshold is arbitrary. The concept of Grid-based and Sample-based method could be used in 3D environment but we should take into account the Z axis information, and think about how to define region of interest in 3D environment. The Direct measure does not need to be quantified as Grid-based measure or Simple-based measure, it simply compares the fixations between two scanpath. The last one is Recurrence-based measures. This kind of methods can provide an overall measure of similarity across two scanpath, but for two sequences of different lengths, the longer need be truncated. The Direct measure could be used in the case of 3D, because it only considers the fixations of scanpath. If we can get the scanpath in 3D environment we can directly use this measure. For the Recurrence-based measures, I think we cannot use it in the case of 3D directly. Because the vertical and horizontal lines is difficult to determine. Please check table1 and table2 for more details.

4 related works in 3D

In this section I will introduce two methods. The first is called "A Vector-based, Multi-dimensional Scanpath Similarity Measure“. Written by Gerd Marmitt, Andrew,etc. It's a very early paper published at 2002 and makes contributions on the field of comparing human and artificial scanpaths recorded in VR [10]. The second method is "Gazer VR2.0: Comparison of eye-tracking data from viewing a 3D scene in virtual reality", which is proposed by Matthieu Le Flohic, Solange Daguise, are two engineering students. This method use three types of data to represent a point in 3D environment, they are timestamp, the position of headset, the position of gaze point. And then, there are three parameters α, β, and γ to give a coefficient to the three types of data.

Comparison List			
Group	Measure	Advantage	Disadvantage
Grid-based	Edit distance	**It** directly compare results to these earlier studies and captures the intuitive notion of scanpath distance in a simple way	**The** definition of regions of interest was time-consuming. And fixations may be considered different even when they are close together
	ScanMatch	**It** can take into account spatial, temporal, and sequential similarity between scanpaths.	**It** suffers from the quantization issues inherent in any measure using regions of interest or grids.
Simple-based	Fixation overlap	**this** method preserves temporal ordering,extremely sensitive to differences in absolute timing between two scanpaths	**It** uses an arbitrary, predefined radius threshold, with similar disadvantages to the grid-based quantization of string-edit and ScanMatch.
	Temprol Correlation	**It** is very sensitive to temporal and spatial differences between the two scanpaths and is more sensitive to similarities in position than the fixation overlap method	**Be** less robust to noisy data than other measures that employ a grid or radius.
	Gaze shift	**It** reflects the similarity between two scanpaths in sequence of large and small saccades and gives a global viewing strategy.	**It's** hard to prediction because it simultaneously quantifies and temporal similarity

Table 1: Table about comparison methods

4.1 Modeling Visual Attention in VR

This method was proposed by Gerd Marmitt and Andrew T. Duchowki [5] in 2002. It contributed a new method that comparing eye tracking data in 3D environment (i.e VR). First, it will get the ground truth of human eye movement by eye tracker. And then, a computational model of visual attention predict human regions of interest. The comparison method was based on string editing. More details will be discussed at the next section.

4.1.1 Methodology for Model Evaluation in VR

The VR environment is not like the traditional still 2D images, testers can move their head and look around in a virtual environment. To compare scanpaths in VR, this methodology collected a sets of still image in VR first. And then, two novel methods

Comparison List			
Group	Measure	Advantage	Disadvantage
	Linear distance	**Don't** need to be quantized and was east to compare fixations between two scanpaths in terms of their spatial similarity.	**This** method ignores sequential information.
Direct measure	MultiMatch	**It** provides several measures for assessing scanpath similarity, and each measure on its own captures a unique component of scanpath similarity	**It's** difficult to assess which measure is most applicable in a given scenario. And it is also not clear how robust each measure is to scanpath variations.
Recurrence-based	Cross-recurrence, Determinsim, Laminaity, Corm	**This** form of cross-recurrence analysis can provide an overall measure of similarity across two eye movement sequences.	**For** sequences of unequal length, the longer sequence is truncated.

Table 2: The remaining part of table 1

of temporal analysis are presented: head-based and time-based. Head-based analysis is used to locate sequences of still images where the head is stable (but the eyes may not be). Based on estimates of head stability, this analysis technique presents a still image to the attentional model for variable periods of viewing time. To more closely resemble the real-time use of the attentional model in VR, the time-based analysis approach assumes a constant frame rate (10 fps), it is constrained to locating aROIs within a constant time period (100 ms). The head-based approach favors the attentional model (since the model is given more time to analyze an image) while the time-based technique better mimics real-time constraints placed on the attentional model in a VR application.

4.1.2 Head-Based Analysis

head movements are considered separately within 3 degrees of freedom in terms of their transnational and rotational movement components. The three degree are Euler angles roll (α), elevation (β), and azimuth (γ) respectively. To obtain angular velocity (ω, in deg/s), the difference between two successive orientations is calculated as a 3-vector and divided by the time between samples. That is, given the instantaneous orientation vector $\theta_i = (\alpha_i, \beta_i, \gamma_i)$

$$\omega = \frac{\|\theta_{i+1} - \theta_i\|}{\Delta_t} deg/s \tag{10}$$

where ω is the angular velocity, Δ_t is the time between samples. And given two successive head positions in three-space, $p_i = (x_i, y_i, z_i)$ and $p_{i+1} = (x_{i+1}, y_{i+1}, z_{i+1})$, velocity is calculated as

$$v = \frac{\|p_{i+1} - p_i\|}{\Delta_t} ft/s \tag{11}$$

where Δ_t is the time between samples. This paper had done experiment to establish a appropriate thresholds for each parameter. To summarize, the following four conditions:

1. $\omega \leq T_\omega$
2. $v \leq T_v$
3. $|p + \gamma| < A$
4. $|\theta| < B$

where the parameters $T_\omega, T_v, A, andB$ well be given by empirically based on the experiment.

4.1.3 Time-Based Analysis

A time-based analysis approach was developed to evaluate the accuracy of Itti et al.'s model []in a real-time setting where images of constant duration (based on frame rate) are presented to the viewer. To match the real-time frame rate of 10-12 fps reported by Haber et al., the performance of Itti et al.'s algorithm was examined. Following Haber et al.'s recommendations, tests showed that Itti et al.'s algorithm can be made in most cases to extract the most salient region within 100 ms. This time, instead of identifying sequences of images displayed during stable head movements, hROIs(human eye movements Region of Interest) and aROIs (attentional Region of Interest) are compared over frames collected every 100 ms.

4.2 Our team's previous work

In this section, we will introduce a method they have had proposed, "Gazer VR2.0 etc"

4.2.1 Collecting data

The data set collected is formed as a vector list which describes the position in the space of the helmet, the position in the space of the point watched as well as the temporal data. The head and gaze position are represented by 3D coordinates, it means the position of the head and gaze in the space. And time stamp is a scalar. Fig 10 shows the format of our data.

4.2.2 Measurement

They compare two sets of data and derive several measures based on various criteria. The data used here are the result of interaction, of an individual with a virtual reality 3D environment in which it is possible to move. Here is the description of how is the data they compared. Each data is a list of multidimensional vectors. Each vector contains the following data:

```json
"dataset": {
  "0": {
    "timestamp": 1548248188437,
    "pos": {
      "x": 1.3445571000095538,
      "y": 2.0344827504014082,
      "z": 2.043493299448808
    },
    "gaze": {
      "x": 0.9132832374728457,
      "y": 1.0384518128496083,
      "z": 0.6921216235538485
    }
  },
  "1": {
    "timestamp": 1548248188737,
    "pos": {
      "x": 1.3445571000095538,
      "y": 2.0344827504014082,
      "z": 2.043493299448808
    },
    "gaze": {
      "x": 2.2781107724865226,
      "y": 0.538542749287847,
      "z": 1.406209740687915
    }
  },
```

Figure 10: The data we collected

- Time data: timestamp in milliseconds
- The position of headset: coordinate (x, y, z)
- The position of gaze point: coordinate (x, y, z)

Definition 1. The time distance of the two timestamps of two vectors v_{A1} and v_{B1} of data sets A and B is calculated as follows:

$$d_t(t_{A1}, t_{B1}) = \frac{abs(t_{A1} - t_{B1})}{max(T_A, T_B)} \tag{12}$$

We choose to divide the difference between the two timestamps by the higher of the two duration of the data sets, as this allows us to have a time distance between 0 and 1.

Definition 2. The angular distance between the two positions of the helmets p_{A1} and p_{B1} and the attachment g_{A1} is as follows ($p_{A1}\widehat{g_{A1}}p_{B1}$ is the angle $\angle g_{A1}p_{B1}g_{B1}$):

$$d_p(p_{A1}, p_{B1}) = \frac{p_{A1}\widehat{g_{A1}}p_{B1}}{180} \tag{13}$$

Definition 3. The angular distance between the position of the helmet p_{A1} and the two fasteners g_{A1} and g_{B1} is as follows:

$$d_g(g_{A1}, g_{B1}) = \frac{g_{A1}\widehat{p_{A1}}g_{B1}}{180} \tag{14}$$

Definition 4. Let 3 coefficients α, β and γ be chosen as a function of the importance given to time dimensions, angular positional distances, and angular distances of fixations. The overall distance between two vectors v_{A1} and v_{B1} computed from a multidimensional distance vector is calculated as follows:

$$d_v(v_{A1}, v_{B1}) = v_{d_v(v_{A1}, v_{B1})} \times \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} \tag{15}$$

that is to say,

$$d_v(v_{A1}, v_{B1}) = \alpha d_t(t_{A1}, t_{B1}) + \beta d_p(p_{A1}, p_{B1}) + \gamma d_g(g_{A1}, g_{B1}) \tag{16}$$

4.2.3 Comparison

There are two data set and each of them has consist of numbers of sets of vectors. Fig 11 shows how are the two data set compared. The vectors in the first data set will be compared with each vectors in the second data set. We will first compare V_{A1} with V_{B1} and V_{B2} and etc. Until the last vector of the second data set. Then we will compare V_{A2} with V_{B1} and V_{B2} and etc.

With this strategy, each vector of the first set of data will be compared with all the vectors of the second set of data. And each of these comparisons will return a vector containing three distances. And the three coefficients will be allocated to the three distances like the formulation 16 to give a weight to the three distances. After that, for each comparison there will be a value like $d_v(v_{A1}, v_{B1})$ to measure the difference between the two vectors. And only the smallest value will be kept as a match. That means V_{A1} in the first data set only match one of the vectors in the second data set. And for V_{A2} and so on and so forth. Finally, for each vectors in the first data set, this algorithm will find the best match of the second data set. After that , three new coefficients will be allocated to the three different distances of each vectors, and all the differences between each vectors will be accumulated and then averaged. This algorithm only return one value to represent the difference of two data set.

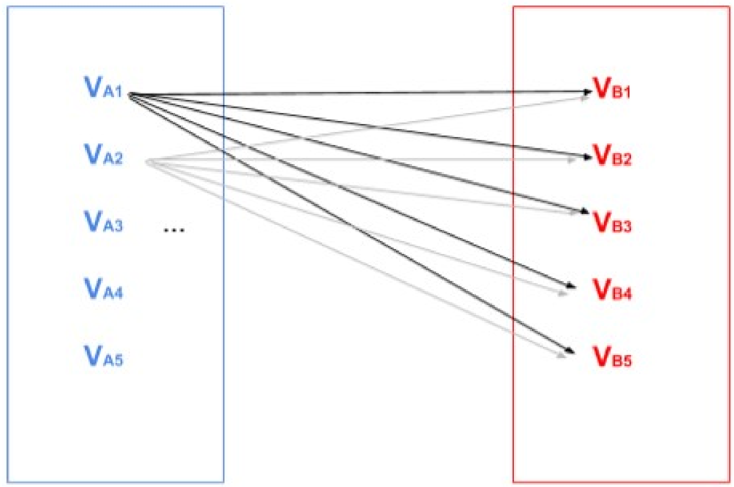

Figure 11: Comparison method and left is first data set right is the second data set.

4.2.4 Drawbacks of the existing method

1. They didn't consider the distance information of headset and gaze. The only information they have is the angle between the two vectors. This means if two observers stand at different distance but has the same angle with the gaze object, there will be the same similarity (difference) between the reference target (based on their equation 13).But we hope the closer observer should has high similarity than the far one. As figure 12 shows.

2. All calculation is linear. we can't think it is linear default. Based on the human visual system, with the angle between the observer and gaze object increased to $180°$, the influence will be increased sharply. More details look at figure 13.

5 Our proposed methods

As mentioned above, there still exist some shorts in the previous methods. The new methods we proposed hope to deal with the two shorts. To make the problem simple we focused on a very easy scenario —— only two observers were looking at the same gaze point (Figure 14). To determine which area the observer were looking at, we need take into the fovea–the small area on the human eye's retina which has a sharp vision. In the Fig14 we assume the orange dotted ellipse (called s_{p1}) is the area the observer $P1$ is looking at, and the blue one (called s_{p2}) is the observer $P2$ looking at. The intersect area (called s_{sec}) is the common area the two observer looking at. So, for the observer $P1$, the similarity is: $\frac{s_{sec}}{s_{p1}}$. For the observer $P2$, the similarity is $\frac{s_{sec}}{s_{p2}}$. The similarity between $P1$ and $P1$ is:

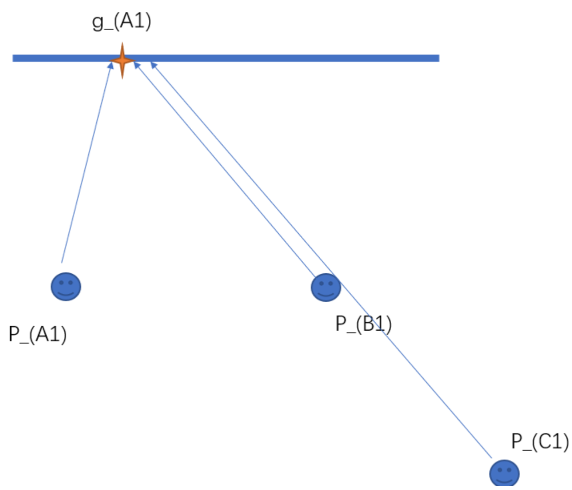

Figure 12: there are three observers P_{A1},P_{B1},P_{C1} and they all look at the same object g_{A1}. P_{B1} and P_{C1} have the same angle with P_{A1}, and P_{C1} stands at a further place. In this situation, $d(P_{A1}, P_{B1})$ (distance between P_{A1} and P_{B1}) should less than $d(P_{A1}, P_{C1})$. But $d(P_{A1}, P_{B1}) = d(P_{A1}, P_{C1})$ if we based on equation 13 .

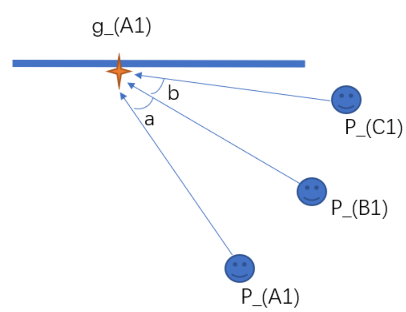

Figure 13: there are three observers P_{A1},P_{B1},P_{C1} and they all look at the same object g_{A1}. a represents the angle between observer P_{A1} and P_{B1}, b is the angle between observer P_{B1} and P_{C1}, and $a = b$. Based on the equation 13, $d(P_{A1}, P_{B1}) = d(P_{A1}, P_{C1})$, but we hope $d(P_{A1}, P_{B1}) < d(P_{A1}, P_{C1})$. Because observer P_{C1} has a very sharp angle with the plane, and this means he can see barely about the object g_{A1}.

$$similarity = ave(\frac{s_{sec}}{s_{p1}} + \frac{s_{sec}}{s_{p2}}) \tag{17}$$

We can simplify this question as how to calculate the area of the ellipse which formed by a plane intersected with a conic, and the area of the intersection of two

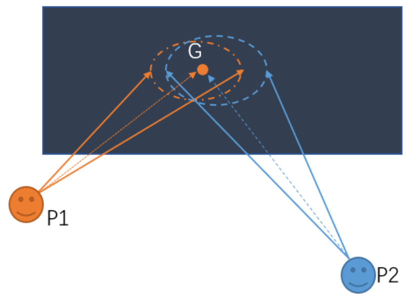

Figure 14: Two observers were looking at the same gaze point G.

ellipses. The eyesight is a conic which project by human's eyes and it will be intersected by a plane. The gaze points are on this plane. The eye has roughly 200 degrees field, but to form a high resolution images, only 15 degrees data is clearly. In low light, the degrees will be more decreased [7]. As show in Fig 15, $2\alpha = 15°$, and we already know the coordinates of position point P and gaze point G, O is the center of the formed ellipse. L is the length of the vector $\vec{PG}$, β is the angle between $\vec{PG}$ and the norm vector of the plane. So, how can we get the area of the two ellipses and the intersection part respectively?

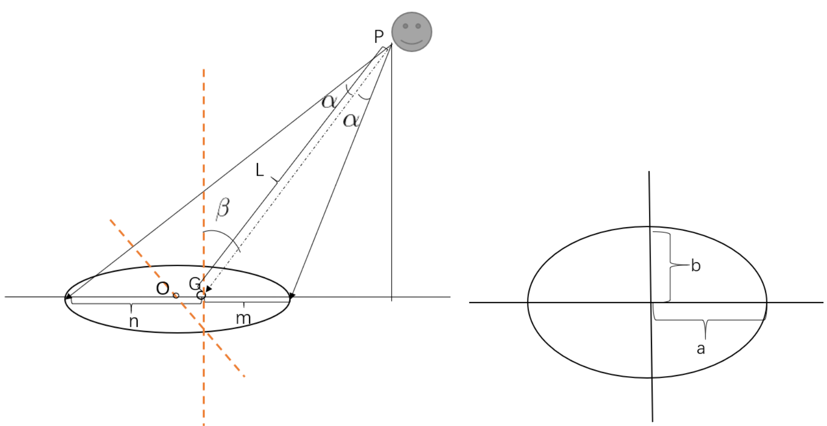

Figure 15: The formed ellipse

5.1 how to get the area of each ellipse

We already have the information of position point P and gaze point G, and the angle of the fovea is $15°$. These information are not enough for us to get the area of formed ellipse, we still need the norm vector of the projection plane (which ellipse projected on), we assume it as $no\vec{r}m$. From the Fig15 left, we can get L, γ, and α. Through trigonometric function, we can get m and n, the formula is as follows:

$$m = L * \sin\beta - L * \tan(\beta - \alpha) * \cos\beta \tag{18}$$

$$n = \frac{L * \cos\beta}{\tan(\pi/2 - \beta - \alpha)} - L * \sin\beta \tag{19}$$

$$a = \frac{m+n}{2} \tag{20}$$

And from the Fig15 right, we can know that $2a = m + n$, so we can get a (formula 20). And then look at Fig16, we through gaze point G make a line l_2 which parallel to l_1 and intersect the ellipse with point Q. The Fig16 is below. $\angle QPG$ is α, $\angle PGQ$ is $90°$. if we assume the origin is the center point O, we can deduce the coordinates of point Q:

$$\begin{cases} X_Q = a - m, \\ Y_Q = L * \tan\alpha \end{cases} \tag{21}$$

Base on the formula of ellipse and the cooridinates of point Q, we can deduce b as follows:

$$\begin{cases} \frac{x^2}{a^2} + \frac{y^2}{b^2} = 1, \\ (X_Q, Y_Q), \\ b = ay\sqrt{\frac{1}{a^2 - b^2}} \end{cases} \tag{22}$$

Eventually, we can calculate the area of the formed ellipse, $s = \pi * a * b$

5.2 How to calculate the area of the intersecting part

There are two different situations for intersecting. The first one is only two intersecting points, and the second one is four intersecting points. As follows is the demonstration (Fig17). We need to get the coordinates of these intersecting points and then we can calculate the intersecting area of it. I will show how to deal with it respectively.

5.2.1 How to calculate the coordinates of the intersecting points

We have two ellipses here and we already know the a, b of it. The idea is if we can put these two ellipses in the same coordinates system, then we can get the equation of each of then, we can combine these two ellipses equation and solving the two equation to get the solution.
If there is no solution for the two equations, it means the two ellipses have no intersecting point. And if only one solution only one intersecting point, two solutions

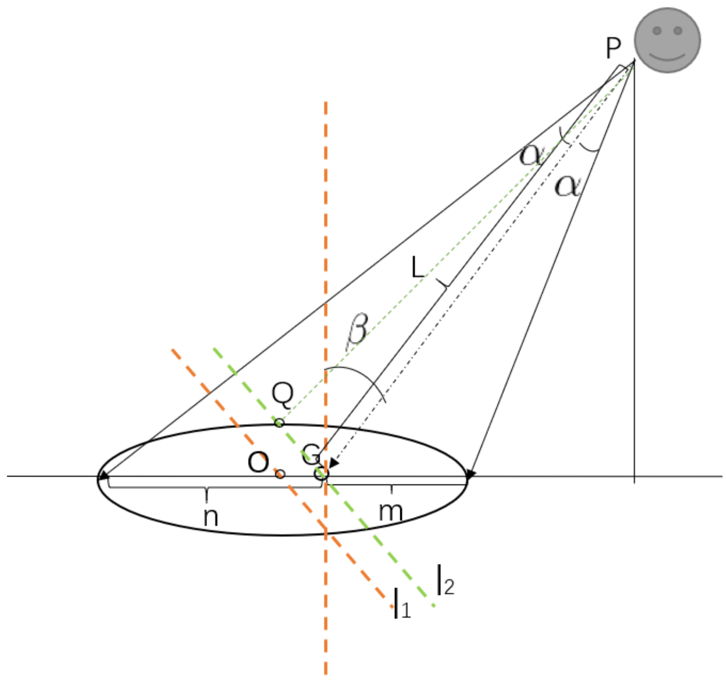

Figure 16: The formed ellipse.

two intersecting points, three solutions three intersecting points. And up to four solutions, it means four intersecting points for the two ellipses. We view the first ellipse as the main ellipse, it's equation as

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1 \tag{23}$$

And for the second ellipse we view it as inferior one, we assume its equation as (This is the general equation of the ellipse, (x_0, y_0) is the coordinates of the center point of the ellipse)

$$A(x - x_0)^2 - B(x - x_0)(y - y_0) + C(y - y_0))^2 + f = 0 \tag{24}$$

And we also know the formula below (θ is the angle from the positive horizontal axis to the ellipse's major axis [19])

$$\begin{cases} A = a^2(\sin\theta)^2 + b^2(\cos\theta), \\ B = 2(a^2 - b^2)\sin\theta\cos\theta, \\ C = a^2(\cos\theta)^2 + b^2(\sin\theta), \\ f = -a^2b^2 \end{cases} \tag{25}$$

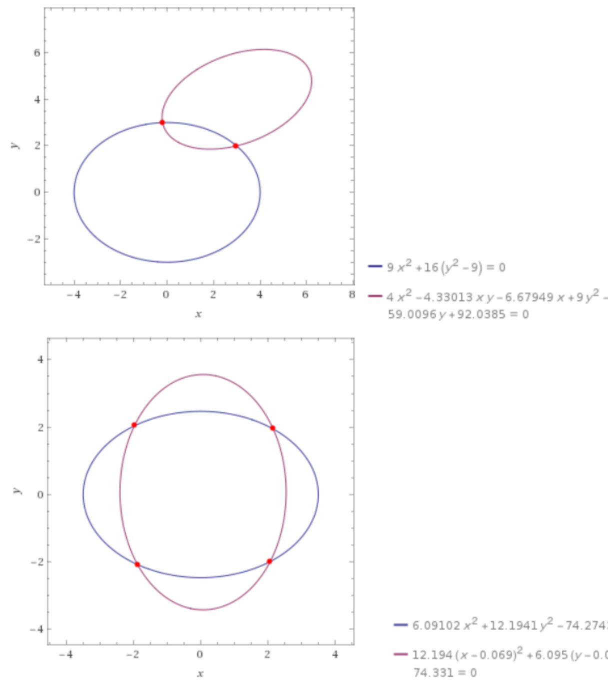

Figure 17: The formed ellipse

For more information about Ellipse please visiting https://en.wikipedia.org/wiki/Ellipse. We already know a, b of the two ellipse, the question is how can we get θ and the center point coordinates (x_0, y_0) of the inferior ellipse. In the discussion following, we just represent θ as γ. γ is the angle between plane P1 and plane P2 (show as Fig 18 and Fig19). Then I will show how to get the coordinates of center point (x_0, y_0) and γ we already know the coordinates of P1, Q, P2, and norm vector of the plane $\vec{Normvector}$. The vector $\vec{P_1Q}$ and vector $\vec{P_2Q}$ is easy to get, and the angle $\angle P_2QF$, $\angle P_1QE$ is easy to get, the formula is following.

$$Angel = \arccos \frac{\vec{A}\vec{B}}{|A||B|} \tag{26}$$

After we get angle of the observing vector and the norm vector, it is not hard to get the angel γ between plane P1 and plane P2.

$$\begin{cases} \vec{P_1E} = \vec{P_1Q} + \vec{QE}, \\ \vec{QE} = (0, 0, |P_1| \cos Angle_{P_1QE}) \ (and\ the\ same\ for\ \vec{P_2F}), \\ \gamma = \arccos \frac{\vec{P_2F}\vec{P_1E}}{|P_2F||P_1E|}, \end{cases} \tag{27}$$

For Fig19, O_1 and O_2 are the center point of main ellipse and inferior ellipse respectively, we assume the coordinates of O_1 is (0,0), and we already know a_1, b_1, m_1, n_1

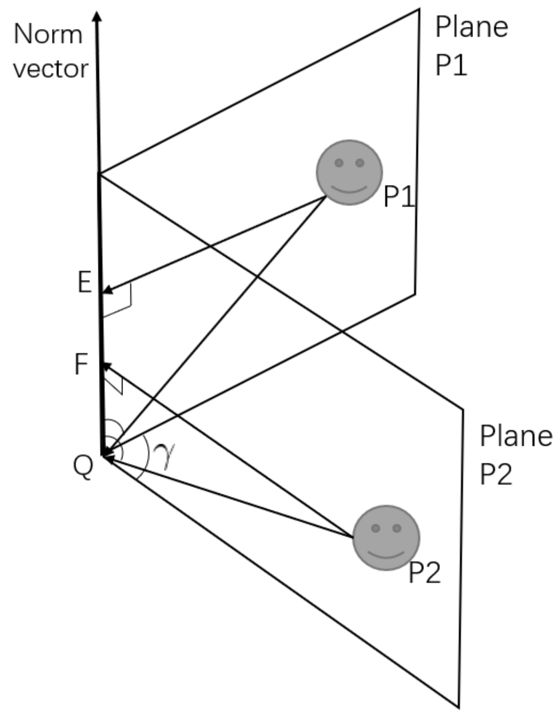

Figure 18: How to get the angle between the two planes

(the formula 18, 19, 20, 22). So, we can get the coordinates of O_2

$$\begin{cases} X_{O_2} = a_1 - m_1 + (a_2 - m_2)\cos\gamma, \\ Y_{O_2} = (a_2 - m_2)\sin\gamma,\ (m_2\ is\ O_2Q, and\ it\ isn't\ marked\ on\ Fig19) \end{cases} \tag{28}$$

Look back on the formula 24 and formula 25, we already get x_0, y_o (they are X_{O_2}, Y_{O_2}) and θ (it is γ), so we can combine the formula 23 and formula 24 to get the solutions of these two equations. The equation may have 0 to 4 solutions. For 0 and 1 solution case, it means there is no intersecting region. For 2 and 3 solution case, it is show like Fig 20. For the 4 solution case it is show like Fig 23. Our goal is to calculate the area of the intersecting region (the colored region) and we will discuss it in the next chapter.

5.2.2 For the two intersecting points case

Below is the demonstration of the two intersecting case (Fig20), our goal is to get the area of the black region. We divide this region into two parts, like Fig 21. If we can get the area of each parts we can get the area of the whole black region. Here we only show how to get the area of the left one (Fig21 left). For the area of this region, we

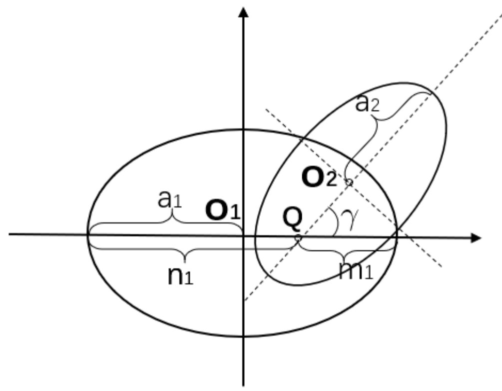

Figure 19: get the coordinates of the center point of the inferior ellipse

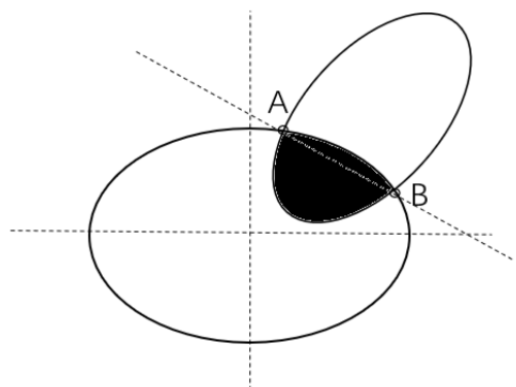

Figure 20: Two points case 1

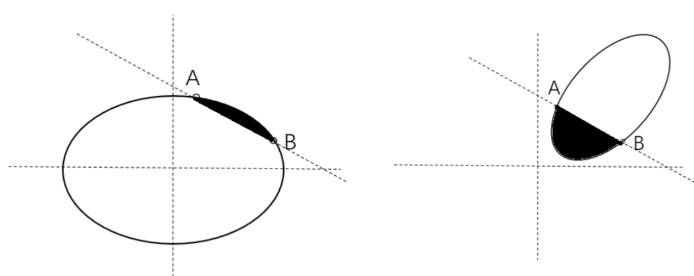

Figure 21: Two points case 2

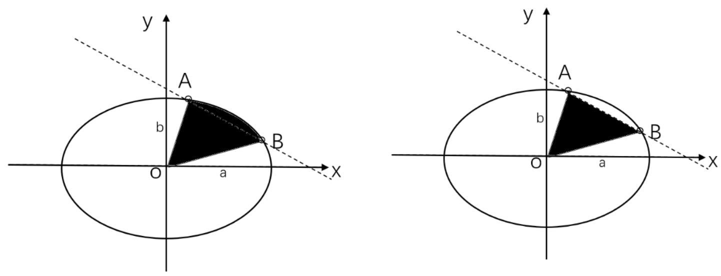

Figure 22: Two points case 3

can use an elliptical arc and its corresponding elliptical sector (Fig22 left) minus the area of the triangle whose vertices are the origin O(0,0), and the arc endpoints A and B (Fig22 right).
For the area of Fig22 left, we can use polar-coordinate to represent the $x = r\cos\theta$ and $y = r\sin\theta$. And we know the equation $\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$, so we can get the new one:

$$r^2 = \frac{a^2b^2}{b^2(\cos\theta)^2 + a^2(\sin\theta)^2} \tag{29}$$

The area of this sector is

$$A(\beta,\alpha) = \int_{\beta}^{\alpha} \frac{1}{2}r^2 d\theta = \int_{\beta}^{\alpha} \frac{(a^2b^2/2)d\theta}{b^2(\cos\theta)^2 + a^2(\cos\theta)^2} \tag{30}$$

An anti-derivative of the integrand is

$$F(\theta) = \frac{ab}{2}\left[\theta - \tan^{-1}\left(\frac{(b-a)\sin 2\theta}{(b+a)+(b-a)\cos 2\theta}\right)\right] \tag{31}$$

The area of the elliptical sector is therefore [16]

$$A(\beta,\alpha) = F(\alpha) - F(\beta) \tag{32}$$

For the region of triangle (the Fig22 right), we can use the formula below, x, y is the coordinates of points A, B

$$A_{triangle} = \frac{1}{2}|x_1y_0 - x_0y_1| \tag{33}$$

Eventually, we can get the area of the aimed region (Fig21 left) as below. And for the region of the other part (Fig21 right) we can get it in the same method.

$$A_{arc} = A(\beta,\alpha) - A_{triangle} \tag{34}$$

5.2.3 For the four intersecting points case

Like the figure below (Fig23), the two ellipses intersecting at four points A, B, C and D. To get the area of this region, we need to divide it into two parts, the blue region and the black region. For the blue region (Formula 34), we can just deal it with the same method described above (I should mentioned here the four blue regions were not necessarily equal, it means we should use the formula above four times.). And for the black region, obviously, it was a polygon. We already get the coordinates of intersecting points A, B, C and D. And we can use the formula (Formula 35) below to get it's area [22]. Eventually we can get the intersecting area of the four intersecting points case, the formula (36) is following:

$$A_{polygon} = \frac{1}{2}(x_1y_2 - x_2y_1 + x_2y_3 - x_3y_4 + ... + x_{n-1}y_n - x_ny_{n-1} + x_ny_1 - x_1y_n) \quad (35)$$

$$A_{4points} = A_{arc_{AB}} + A_{arc_{BD}} + A_{arc_{DC}} + A_{arc_{CA}} + A_{polygon} \quad (36)$$

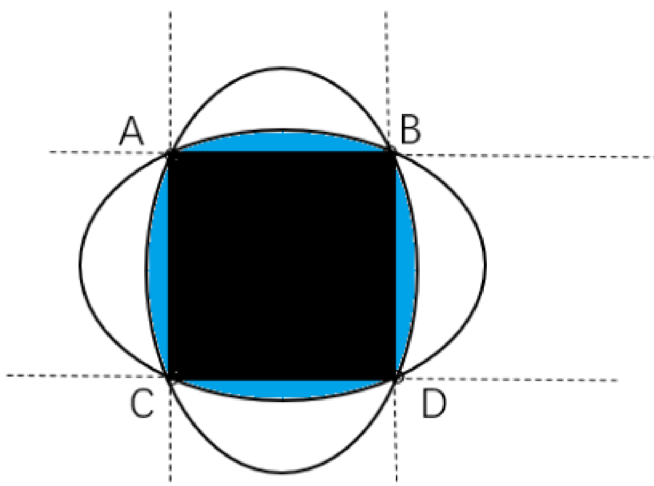

Figure 23: Four points case

6 Conclusion

In the beginning of this report, we introduced the background of the eye-tracking analysis domain. What is eye tracking, how does eye tracker work and so on. And then we enumerate the present methods for analyzing the eye-tracking data and demonstrated the ups and downs for these methods. There are lots of related works in two dimensional environment and we classify it as four categories, they respectively are Grid-based method, Sample-based method, Direct Measures, and Recurrence-based method. The Grid-based measures compares the scanpath in a simple way but how to define the region of interest is a trade-off problem. The Sample-based methods is sensitive to temporal but it still exist the same trouble as Grid-based methods. These two methods can be adapted to three dimensional environment if we consider the Z axis information and have a good way to define the region of interest in 3D environment. The Direct measurement methods simply compare the fixations between two scanpath, and the Recurrence-based measures can provide an overall measurement for two scanpath.

Beside these, we also analysis some works in three dimensional environment and we proposed a new method based on the previous works. There are some shortcomings in the methods proposed by our previous team. First is they didn't consider the distance information of headset and gaze, it means the comparison results will not change even if the two observers changes the distance sharply but just keep the same angle. And the second shortcoming is all the calculation is linear, it means with the angle changed the comparison results will also changed in the same speed.

To solve these two problems, we proposed a new method based on the fovea of human visual system and the projection of eyesight seeing. We think the comparison results should based on the projection ellipse of human eyesight seeing. To make the problem simple we only consider a very primary scenario, the observers were looking at the same object but with different angles and distance. This means we keep the gaze point fixed. To get the area of the projection ellipse, we also assume we already know the norm vector of the plane which the gaze point belongs. After this, we analysis the geometric relationship between the two observers and the gaze point, verified that the area of the projected ellipse is possible to get. Furthermore, we argue that if we can get the intersecting area of the two ellipses, we can average the values of the proportions of the intersecting area in each ellipse as the final comparison result. And we also demonstrated that it is feasible to get the intersecting area in a mathematical way.

7 Perspectives

There still are lots of works to do in the future. We only focus on a very simple scenario and we assumed we already know the norm vector of the plane. In some complex scenario, it is very hard to get the norm vector of all the plane, and the computational complexity is comparatively high than the previous method. So, how to get the norm vector and make this method works in some complex scenario is a good direction for the future works, and how to diminish the computational complexity is also a problem worthing to deal with.

References

[1] Nicola C. Anderson, Walter F. Bischof, Kaitlin E. W. Laidlaw, Evan F. Risko, and Alan Kingstone. Recurrence quantification analysis of eye movements. *Behavior*

Research Methods, 45(3):842–856, Sep 2013.

[2] Saul B. Needleman and Christian D. Wunsch. A general method applicable to search for similarities in amino acid sequence of 2 proteins. *Journal of molecular biology*, 48:443–53, 04 1970.

[3] Filipe Cristino, Sebastiaan Mathôt, Jan Theeuwes, and Iain Gilchrist. Scanmatch: A novel method for comparing fixation sequences. *Behavior research methods*, 42:692–700, 08 2010.

[4] Mathôt S. Theeuwes J. Gilchrist I. D. Cristino, F. Scanmatch: A novel method for comparing fixation sequences. *Behavior Research Methods*, (42(3)):692–700, 2010.

[5] Andrew T. Duchowski Gerd Marmitt. Modeling visual attention in vr: Measuring the accuracy of predicted scanpaths. *The Eurographics Association*, 2002.

[6] HTC. Htcvive. https://www.vive.com/cn/.

[7] hyperphysics. The fovea centralis. http://hyperphysics.phy-astr.gsu.edu/hbase/vision/retina.html.

[8] imotions. Eye tracker technology. https://imotions.com/blog/eye-tracking-work/.

[9] Halszka Jarodzka, Kenneth Holmqvist, and Marcus Nyström. A vector-based, multidimensional scanpath similarity measure. In *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*, ETRA '10, pages 211–218, New York, NY, USA, 2010. ACM.

[10] Halszka Jarodzka, Kenneth Holmqvist, and Marcus Nyström. A vector-based, multidimensional scanpath similarity measure. In *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*, ETRA '10, pages 211–218, New York, NY, USA, 2010. ACM.

[11] Ruddock K. H. Wooding D. S. Mannan, S. K. Automatic control of saccadic eye movements made in visual inspection of briefly presented 2-d images. *Spatial Vision*, 9(3):363–386, 1995.

[12] Microsoft. hololens. https://www.microsoft.com/en-us/hololens.

[13] mro.network. worldviz. https://www.mro-network.com/emerging-technology/collaborative-virtual-reality-platform-announced.

[14] AlanKingstone Walter F. Bischof NicolaC.Anderson, FraserAnderson. A comparison of scanpath comparison methods. *Behav Res*, 322(47):1377–1392, 2014.

[15] Uri Hasson Asif A. Ghazanfar Stephen V. Shepherd, Shawn A. Steckenfinger. Human-monkey gaze correlations reveal convergent and divergent patterns of movie viewing. *Current biology*, 20:649–56, 03 2010.

[16] Geomatric tools. The area of intersecting ellipses. https://www.geometrictools.com/Documentation/AreaIntersectingEllipses.pdf.

[17] Wikipedia. Augmented reality — Wikipedia, the free encyclopedia, 2019. AR.

[18] Wikipedia. Electrooculography — Wikipedia, the free encyclopedia, 2019. EOG.

[19] Wikipedia. Ellipse — Wikipedia, the free encyclopedia, 2019. Ellipse.

[20] Wikipedia. Eye tracking — Wikipedia, the free encyclopedia, 2019. eye tracking.

[21] Wikipedia. Virtual reality — Wikipedia, the free encyclopedia, 2019. VR.

[22] WolframMathWorld. The area of polygon. http://mathworld.wolfram.com/PolygonArea.html.

[23] A. L. (1967) Yarbus. *Eye Movements and Vision.* Originally published in Russian 1962, New York: Plenum, 1962.