HieraVisVR: Hierarchical Visual Analytics for Motion-Centric VR Playtesting

Yongqi Zhang
Computer Science
George Mason University
Fairfax, Virginia, USA
yzhang59@gmu.edu

Erdem Murat
Computer Science
George Mason University
Fairfax, Virginia, USA
emurat@gmu.edu

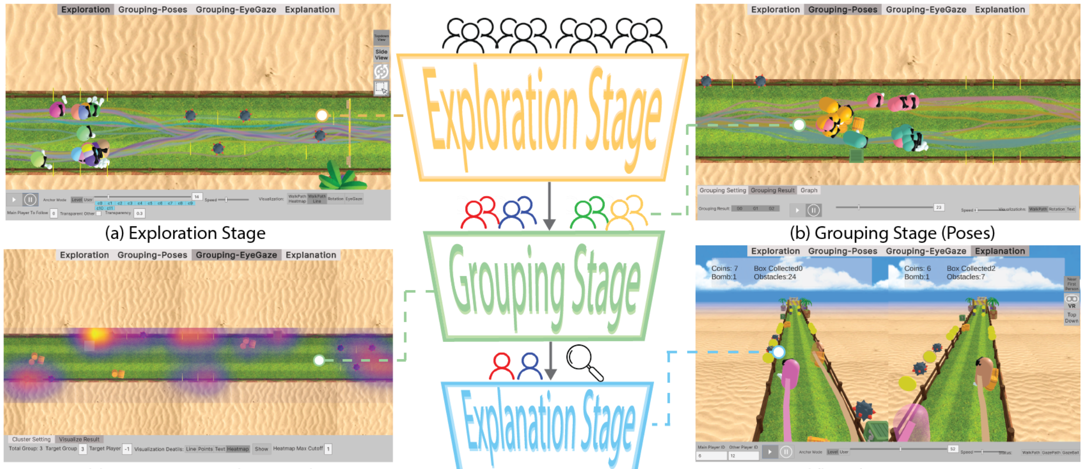
Liuchuan Yu
Computer Science
George Mason University
Fairfax, Virginia, USA
lyu20@gmu.edu

Haikun Huang
Computer Science Department
George Mason University
Fairfax, Virginia, USA
quincy.huanghk@gmail.com

Minsoo Choi
Department of Computer Science
Oklahoma State University
Stillwater, Oklahoma, USA
minsoo.choi@okstate.edu

Christos Mousas
School of Applied and Creative Computing
Purdue University
West Lafayette, Indiana, USA
cmousas@purdue.edu

Lap-Fai Yu
Computer Science
George Mason University
Fairfax, Virginia, USA
craigyu@gmu.edu



(a) Exploration Stage

(b) Grouping Stage (Poses)

(c) Grouping Stage (Eye Gaze)

(d) Explanation Stage

Figure 1: HieraVisVR supports a three-stage playtesting analytics workflow. In the *exploration stage*, designers can use the replayer with anchors to review overall gameplay performance with the aid of motion-based visualization, such as a walking path line (see Figure 1(a)). In the *grouping stage*, they can select gameplay performance-related features (e.g., trajectory, gameplay duration) to identify groups of players they are interested in and analyze their behavior. For example, Figure 1(b) shows three groups of players classified by their walking paths, with the resulting grouped walking paths displayed. Figure 1(c) shows the gaze distribution of a group as a heatmap. In the *explanation stage*, designers can focus on specific players or pairs of players to replay and closely examine their performance. Two players with their current gameplay performance (top canvas) under a near first-person point of view are shown in Figure 1(d). The solid color represents the main player in that camera view, while the other player is represented with a transparent color.

Abstract

Playtesting is widely used in the game industry to identify design flaws and evaluate player experience, yet little research explores how to effectively visualize and analyze playtesting data. This challenge is particularly pronounced in motion-based VR games, which

---

CHI '26, Barcelona, Spain

ACM ISBN 979-8-4007-2278-3/26/04
https://doi.org/10.1145/3772318.3790591

involve physical movements and interactions tracked through multimodal inputs, resulting in complex multidimensional data. To better understand the challenges designers face, we conducted a formative study with 30 practitioners in the VR domain to characterize playtesting workflows and associated tasks. Based on these findings, we present HieraVisVR, a hierarchical visual analytics framework that incorporates body-motion-related data to help designers identify player behaviors and critical game moments, simplifying their workflow. We demonstrate the applicability of HieraVisVR in three different applications and evaluate our system with playtesting experts through an analysis of motion-based game data. The study results suggest that our system enhances playtesters' understanding of the gameplay and improves their data analysis workflow.

CCS Concepts

• **Human-centered computing** → **Interactive systems and tools**.

Keywords

VR, Playtesting workflow

ACM Reference Format:
Yongqi Zhang, Erdem Murat, Liuchuan Yu, Haikun Huang, Minsoo Choi, Christos Mousas, and Lap-Fai Yu. 2026. HieraVisVR: Hierarchical Visual Analytics for Motion-Centric VR Playtesting. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems (CHI '26), April 13–17, 2026, Barcelona, Spain.* ACM, New York, NY, USA, 19 pages. https://doi.org/10.1145/3772318.3790591

1 Introduction

Playtesting is a well-known practice in game research aimed at understanding player experiences in a game. As shown in Figure 2, this process typically involves placing players in a game, recording their behaviors, and collecting immediate feedback. It may also include conducting interviews or administering questionnaires to capture their gameplay experience and emotional responses in depth. Following data collection, a preprocessing stage processes both objective gameplay data and subjective responses into a consistent format (e.g., `CSV`). These data are then loaded into analysis software to be analyzed and visualized. Designers review the obtained insights and update the game design as needed. In addition, playtesting can help identify any game design flaws before releasing to market [22, 42]. There are many types of playtesting that focus on examining different aspects of the games (e.g., stress, accessibility test). In this work, we focus on user experience playtesting, which aims to ensure a smooth and engaging player experience. Currently, there is no standard way to visualize and analyze playtest data. A common practice involves rewatching gameplay and observing individual performance, performing data analysis on aggregated data, and then presenting the results in a formative presentation [28].

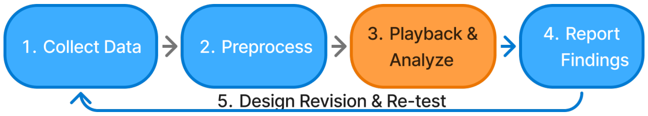


Figure 2: The playtesting workflow.

Motion-based virtual reality (VR) games are receiving unprecedented attention these days: from commercial entertainment (e.g., Beat Saber, Temple Run VR), to health research applications [55], and professional training simulations [23, 56]. However, analyzing player experience in motion-based VR presents unique challenges compared to traditional mouse-and-keyboard games. VR systems collect substantial amounts of multimodal interaction data (e.g., eye gaze and body movement), thus creating rich but complex datasets that encompass both behavioral and motion-related factors. While data analytics approaches such as descriptive and diagnostic analytics offer pathways to understand players' gameplay experiences, the motion-related data itself poses significant analytical hurdles. These multidimensional datasets are inherently time-consuming to process, often requiring specialized preprocessing tools and programs before meaningful analysis can begin. Even after preprocessing, designers and researchers must examine vast amounts of motion data, making it challenging to efficiently identify and extract meaningful patterns that inform design decisions.

These challenges remain unaddressed in current approaches to game analytics. To better understand the current playtesting workflow, we compared the key features of existing third-party tools, common practices identified from our formative phone interviews (see Section 3), and our proposed approach. For the baseline tools, we selected two widely used replay systems that provide general visualization capabilities for specific games: Caldera,[1] a replay tool built on the Call of Duty dataset and adapted into the open-source universal scene description format for visualization; and SCOPE.GG,[2] a replay analysis platform specialized for Counter-Strike 2. Table 1 compares key features across these tools, current practices, and our approach. "Maybe" in the table indicates additional steps or data customization that may be required before the tool can be used. Overall, manual playtesting approaches focus on individual player performance, making it difficult to obtain an overview of the general user experience. Third-party tools provide more basic playback or visualization functions but are often domain-specific (e.g., tied to a particular game) and lack support for group-based analysis or pattern discovery across players.

Meanwhile, research in sports games analytics has progressed from static top-down view [33, 44, 52] to more sophisticated embedded visualizations in augmented sport video [45], simulated environment [25], and live feeds [7]. Yet, how to use embedded visualizations and game analytics to streamline various aspects of the playtest workflow, particularly for motion-based VR games, is still an open question.

In this study, we aim to fill this gap by exploring the design space of visual analytics to assist game designers in reviewing playtest data, facilitating the game analysis workflow. To gain more insights into the general workflow of playtesting VR games and to identify playtesting challenges, we conducted interviews with 30 practitioners in the VR domain. Our findings reveal three challenges they regularly encounter:

1. The lack of a standardized playtesting workflow and inefficiencies in evaluating VR games and applications.
2. Difficulty in extracting insights and evaluating players' experience from a substantial amount of gameplay videos.

---

[1] https://github.com/Activision/caldera
[2] https://SCOPE.GG

Table 1: Comparing playtesting approaches and tools. Both manual playtesting approaches are commonly used by practitioners identified in our formative interviews. Manual Playtesting 1 refers to direct gameplay observation ($n = 21$), while Manual Playtesting 2 involves recording and reviewing gameplay videos ($n = 15$). We also compared two widely-used third-party tools: (1) Caldera, an open-source replay tool built on top of the Call of Duty dataset, and (2) SCOPE.GG, a gameplay visualization tool developed specifically for Counter-Strike 2. In this table, “Maybe” indicates additional steps or data customization that may be required before the tool can be used. Manual playtesting approaches focus on individual performance but provide limited insights into overall user experience and gameplay patterns. Third-party tools support basic playback but are game-specific and require customization for broader uses.

Method / Tool	Individual Playback	Group Playback	In-Game Stats	Event Anchors	Player Grouping	Motion Data Visualization	Cross-Game Applicability
Manual Playtesting 1	✓	×	×	×	×	×	×
Manual Playtesting 2	✓	×	×	×	×	×	×
Caldera	✓	✓	×	Maybe	×	×	✓
SCOPE.GG	✓	Maybe	✓	✓	×	×	×
Ours	✓	✓	✓	✓	✓	✓	✓

(3) Lack of unified tools for examining gameplay moments and identifying behavioral patterns from diverse playtest data.

To address these challenges, we developed HieraVisVR, a hierarchical visual analytics framework that implements a three-stage workflow for reviewing playtest results of motion-centric VR games in a top-down manner. The analytic process begins with the *exploration stage*, where designers gain a general impression of how players as a whole interact with a game, utilizing different viewing angles and a replayer with anchor functionality to observe key gameplay moments. Next, the *grouping stage* allows them to explore different game attributes to categorize players and extract any interesting gameplay patterns from grouping. This stage also includes specialized visualizations for motion-based data, such as player positions and eye gaze. After that, in the *explanation stage*, the user can focus on a specific player of interest to analyze detailed behaviors. Or, the user can select a pair of players to understand how different skill sets of players influence their perception of the game. In this way, our system can process various gameplay data to streamline the playtesting analytics workflow, without the need to switch to other software.

In summary, our contributions comprise the following. First, we conducted a formative study to identify challenges in the playtesting workflow for VR game designers. Second, we proposed a hierarchical visual analytics framework and developed a prototype system built on top of this framework to streamline the playtesting workflow. We demonstrate its utility across three different VR applications. Lastly, we evaluated our system with another group of game researchers to gather insights about the usability and discuss our findings and implications for future research.

2 Related Work

2.1 Playtesting in Games User Research

In game user research [10], several studies define qualitative data as observations and comments collected from players, and quantitative data as gameplay metrics—providing valuable insights into players’ behavior and player-game interaction [9]—or physiological measurements recorded during gameplay [13, 17, 47]. Many researchers have explored visualizing a combination of these data types to gain deeper insights into player experience and behavior.

For example, Mirza-Babaei et al. [29] explored visualizing biometric data, such as skin conductivity levels and facial electromyography measurements, alongside player comments, observational notes, and gameplay metrics. An extension of this work combines physiological data with player trajectories. They use color-coded player movement trajectories to reflect recorded galvanic skin responses, integrating physiological data values into platformer [30]. Moreover, Drenikow et al. [11] proposed a more flexible and interactive toolkit that visualizes player trajectories with additional player facial expression layers.

Another emerging direction is automating playtesting to minimize reliance on human play testers [37]. Since not all game studios, especially indie developers, emphasize or can afford extensive playtesting, automation presents a feasible alternative [28]. Research in automated playtesting includes methods to simulate player behavior, such as predicting player actions with different skill levels and using automation to balance game difficulty.

Much of the existing research focuses on specific game genres such as abstract games or puzzle games [20], with visualization tailored for genre-specific features. In contrast, our work aims to simplify the data analytics workflow for designers by proposing a top-down analytics exploration framework. This framework allows designers to begin with a broad overview of gameplay interactions and progressively dive deeper to discover specific gameplay patterns, identify potential design flaws, and analyze players of interest. Ultimately, it enables the gathering of insights across diverse gameplay data to refine and enhance game design.

2.2 Visual Analysis for Sports and Motion-based Games

There have been research efforts on visual analytics for various sports and motion-based games. We refer readers to [12] and [48] for a more comprehensive review. Early research has focused on extracting meaningful static in-game match data, such as point outcomes and stroke patterns, to provide insights into a single match

Table 2: Comparing analysis workflows across systems.

Category	Our Approach	VIRD [24]	Tac-Miner [49]	EventAnchor [8]
Domain	Visual analytics for VR motion-based games	High-performance badminton coaching analysis	Multi-match table tennis tactic analysis	Interactive sports video annotation
Workflow Type	Hierarchical, top-down	Top-down	Multi-scale visualization	Three-level data acquisition
Stage 1	Exploration: gain a general impression of player interaction	Obtain rallies of interest using summary statistics	Summary view: identify tactics using a steerable projection plot	Object level: detect players, ball, court, and trajectories via computer vision
Stage 2	Grouping: explore attributes to categorize players and extract gameplay patterns	Compare and analyze filtered rallies	Detail view: examine tactic attributes using glyphs	Event level: derive events from ball and player motion, with timestamps serving as anchors
Stage 3	Explanation: analyze detailed behaviors of a player and understand how skill differences shape perception	Investigate game details via shot analysis and multi-angle review	N/A	Context level: summarizes technical and tactical attributes expert annotations
Stage 4	N/A	Verify insights by cross-validating patterns across similar or contrasting rallies	N/A	N/A

performance [36, 51]. Later work extended these ideas by introducing glyph-based visualizations to capture a player's tactics across multiple matches [49] and by developing multi-level visualization systems to support exploration of entire seasons [6].

More recent research has applied data mining techniques to sports data to uncover patterns, particularly using spatiotemporal movement data, which is complex to visualize due to its multidimensional nature [48]. For example, Wu et al. [50] applied pattern mining to racket sports data and identified various tactics and uncovered tactic progression. Polk et al. [35] proposed a visual analytics system that integrated spatio-temporal tracking data of players and the ball with match context. Similarly, Cao et al. [4] developed an action-evaluator system for soccer that segments event streams, links actions to tactics, and predicts the scoring potential and risks associated with each action.

While much of sports research has focused on feature extraction and data mining to advance domain-specific visual analytics, another line of work explores how motion data can be analyzed through immersive and in-situ analytics, particularly using data captured in virtual, augmented, or mixed reality environments. Kloiber et al. [18] explored visualization techniques for VR-specific motion data (e.g., trajectories and teleportation), combining trajectory overviews with avatar-based keyframes to support motion comparison and analysis in VR. Similarly, Nebeling et al. [31] proposed the Mixed Reality Analytics Toolkit for understanding mixed reality experiences. Their toolkit allows non-programmers to instrument MR applications, collect and visualize user interaction data, and provides a set of heuristics and metrics to measure task success. Building on this direction, Buschel et al. [3] introduced a mixed reality toolkit for exploring spatio-temporal datasets, enabling in-situ and collaborative analysis by visualizing 3D trajectories, 2D events, and aggregated data directly in the physical environment where the data was captured. Reipschlager et al. [39] extended this line of work by combining interactive 3D avatars, motion trajectories, and in-situ visualizations (such as gaze points and touch interactions) to provide a richer context for motion analysis. Hubenschmid et al. [14] presented a mixed-immersion visual analytics framework that supports seamless transitions between immersive VR views and desktop-based views, enabling both in-situ and ex-situ analysis for mixed reality (MR) user studies. Finally, Luo et al. [27] extend in-situ data analysis by incorporating the physical environment directly into the visualization and interaction process. Their augmented reality (AR)-based prototype allows analysts to explore, select, and filter human movement data through direct interaction with surrounding objects.

The common goal of these works is to visualize interaction data in-situ, within the original environment where it was captured. Although this method provides a straightforward solution for visualizing motion data, it is often insufficient for game designers tasked with analyzing gameplay data across large player groups. In contrast, our work identifies the challenges game designers face during playtesting, which inspired us to propose a three-stage framework incorporating clustering techniques to support playtesting from different perspectives.

2.3 Embedded Visualization in Sports Games

Embedded visualization has also been widely applied to sports games, where it helps viewers understand the complexities and dynamics of gameplay by overlaying virtual annotations on sports videos. A typical sports game with embedded visualization has a static background (e.g., a basketball court) with some visualizations such as heatmaps for walking paths or graphs for game statistics history (e.g., shot location over seasons) [19, 33].

As computer vision techniques improve, research has advanced sports game-watching experiences and game analytics. For example, Stein et al. [45] utilized computer vision (CV) techniques to extract trajectory data from team sports video so that they could run movement analysis on trajectory data, whose results were embedded in the original video. Deng et al. [8] utilized a machine learning model to identify potential events of interest from videos for viewers to quickly gather important game moments, and to analyze and add annotations to those events. Lin et al. [25] created an interactive design framework to visualize NBA data. Although their interactive system is implemented in a simulated 3D environment, it can potentially be customized for embedded visualization design for regular basketball fans with in-game data. Furthermore, Chen et al. [7] proposed the iball system that can augment basketball video for casual fans, customizing the level of embedded visualization based on the viewers' gaze and preference. Recent research has explored leveraging the immersiveness of VR for sports video analysis, presenting 3D data in a "real 3D" form to provide a more visceral experience and enhance spatial understanding. Ye et al. [53] reconstructed shuttle trajectories using a high-accuracy aerodynamic model and explored visualization designs for interacting with these trajectories in virtual reality. Building on this idea, Lin et al. [24] introduced an immersive analysis system for badminton coaching, supporting multi-scale exploration which starts from a high-level match overview followed by detailed visualizations of individual rallies and shots. The system leverages situated 3D visualizations combined with video playback.

Beyond embedded visualization techniques, several immersive systems adopt top-down analytical workflows that guide users from overview to detailed inspection. For example, VIRD [24], TacMiner [49], and EventAnchor [8] structure analysis around progressive refinement, typically moving from aggregate overviews toward detailed event- or player-level analysis. These systems are designed around domain-specific analytical goals, such as tactical evaluation or event-based performance analysis in sports. See Table 2 for a detailed comparison.

While these approaches support transitions from overview to individual inspection, they generally do not emphasize structured grouping or clustering mechanisms that enable analysts to systematically compare multiple players or reveal cross-player patterns. Moreover, embedded sports analytics systems typically focus on quantifiable performance indicators and spatial representations, such as shot accuracy, tactical strategy, or player positioning. As a result, aggregating gameplay data and jointly analyzing players who progress at different speeds remains challenging during playtesting. These limitations motivate our approach, which explores integrating motion-based behavioral data with in-game performance metrics (e.g., number of collected coins) to better understand player perception and experience in immersive or interactive settings.

3 Interview

To better understand the playtesting workflow and its challenges, we first reached out to practitioners from the VR domain through a university's Game Design program and alumni network. We then asked these practitioners to refer colleagues within the same domain. In the end, we recruited 30 practitioners (i.e., 22 males and 8 females; age range: 22–40) who worked in indie game studios, game companies, or VR production groups. All of them had prior experience in playtesting their VR applications, with 1 to 15 years of experience in game development and playtesting. The distributions by their job titles and working experience are shown in Table 4.

Table 3: Interview Questions for Practitioners.

	Question
Q1.	Could you share your job title, organization, role, and years of experience in this field?
Q2.	What is your overall playtesting approach?
Q3.	What kind of tools do you use, and what features are involved for playtesting?
Q4.	What specific insights do you aim to gather about users, and how do you use these insights to improve design?
Q5.	What challenges or difficulties do you face in the playtesting process?

3.1 Study Setup

We conducted a 30-minute semi-structured phone interview with practitioners. We asked about their general playtesting process, the tools and features they use, the elements they focus on to understand user behaviors, and their method for improving design. Detailed questions are listed in Table 3.

We then analyzed the interview data using a Constructivist Grounded Theory method [5] on the interview data. After transcribing audio into text, all authors gathered to discuss the interview data and create sets of plausible codes. We jointly identified emerging themes and iteratively refined our understanding of the data.

3.2 Findings and Discussion

We present our findings on practitioners' pain points in performing visual analytics for VR playtesting data.

Challenge 1: Lack of standardized playtesting workflow and inefficiencies in VR Testing. Most practitioners noted that their design and development process often involves iterative prototyping under constrained time and resources. Due to the lack of standardization for playtesting, it is typically integrated into the development testing pipeline. Eleven practitioners reported that they usually begin by testing core functionalities themselves before expanding to small groups of users for feedback. When testing core functionalities, ten practitioners mentioned that they often switch between 2D editing tools and deploying the application in a 3D space, which they find inconvenient and tedious. They expressed a desire for a "WordPress-equivalent" for VR content creation—a streamlined and intuitive platform to simplify the iterative design and testing workflow. On the other hand, when playtesting with their target users, the top priorities are ensuring a smooth experience and minimizing as much confusion as possible. As one noted, a 'sticky' experience in a VR app can ensure that users will return

Table 4: Practitioners' Job Titles and Working Experience Distributions.

Job Title	Count	Working Experience	Count
Industrial XR Researcher / Indie Developer	8	1–3 years	8
XR / Instructional / Game Designer / Content Creator	15	4–9 years	15
Project Manager / Technical Lead	4	10+ years	7
Technical Consultant / Advisor	3		

Table 5: Three design goals for assisting designers to streamline the playtesting workflow were revealed from the formative study.

Findings	Design Goals
Challenge 1: Lack of standardized playtesting workflow and inefficiencies in VR testing.	G1: A visual analytic system seamlessly integrated into the current workflow, capable of processing various gameplay data to streamline the playtesting analytics workflow without requiring additional tools.
Challenge 2: Difficulty in identifying & extracting gameplay patterns from a large volume of gameplay video.	G2: Providing support to group players based on criteria to uncover gameplay patterns and identify players of interest.
Challenge 3: Needs for an integrated system to examine specific gameplay moments, identify behavioral patterns, and analyze diverse playtesting data.	G3: Various visualizations to show body-motion related data, with a anchor functionality to highlight important gameplay moments and a gameplay synchronization functionality for player comparisons.

to it (P5). However, each user has a different background experience with VR and a varying tolerance for VR headsets. This makes it difficult to evaluate whether the current guidance and control are suitable for the general audience. Thus, they desire a standardized workflow to evaluate user experiences.

Challenge 2: Difficulty in extracting meaningful insights from a large volume of gameplay videos to effectively assess player experience. Most practitioners have emphasized the importance of observing player behaviors to gather insights about player experience. Typically, twenty-one practitioners reported observing gameplay directly, while fifteen reviewed recorded gameplay videos. Three practitioners mentioned they would share gameplay videos with clients to gather initial feedback. One explained *"Showing data is extremely critical to convince clients to adopt my new solution (i.e., VR training app)"* (P6). Seventeen others found the recording useful for investigating very specific problems.

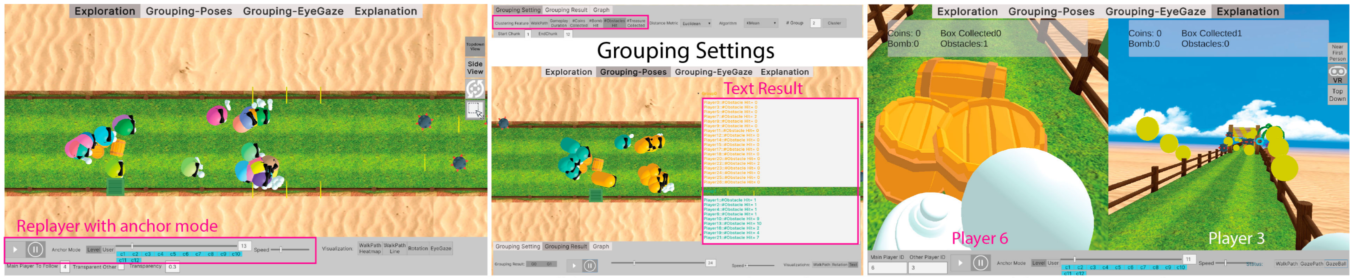
However, when scaling up to larger volumes of players, issues arise. One practitioner noted that *"When I scaled up the test to 50 players, it became a time-consuming and costly task because I needed to watch so many players for a long time."* (P17). Three practitioners mentioned that the fidelity of VR headsets differs from testing in the play mode in Unity, requiring them to first record in the VR headset and then export the data, which they found troublesome. Additionally, thirteen practitioners expressed interest in observing how users interact with their applications from a first-person perspective. Thus, they sought tools that could help them quickly identify user interests and analyze engagement experiences.

Challenge 3: Needs for integrated system to examine specific gameplay moments, identify behavioral patterns, and analyze diverse playtesting data. Thirteen practitioners expressed the importance of examining player behaviors at specific game moments and identifying gameplay patterns as part of their playtesting analysis process. Their focus is twofold: on one hand, determining whether the current game mechanics or guidance in the game/application help players follow the expected gameplay; on the other hand, determining whether there are any unexpected events during interactions that may cause confusion or reveal bugs. One practitioner noted, *"I first find out the 'not so obvious' factors in the player experience, then I will identify patterns in how players interact with the game/UI."* (P2). When it comes to debugging, another practitioner emphasized, *"It is important for me to locate key moments in the gameplay video where bugs occur."* (P5).

In addition to video replays, twelve practitioners use a mix of qualitative and quantitative methods to gather feedback. This includes in-game body-related data (e.g., hand-tracking) and verbal feedback (e.g., survey, rating) during or after testing. The tools used for analysis are diverse, with practitioners often relying on familiar options such as Excel, Google Sheets, or in-house tools developed by their company. Ultimately, they believe that it would be highly beneficial to have a toolkit that combines all these functions and simplifies the data analysis workflow while supporting player profiling to examine player performance.

3.3 Design Specifications

From the identified challenges and needs, we derived three design goals (Table 5) that a playtesting framework should satisfy. First, it should seamlessly integrate with existing workflows and provide an overview of diverse gameplay data without requiring additional



(a) Exploration Stage (b) Grouping Stage (Poses) (c) Explanation Stage

Figure 3: Overview of how a designer could analyze playtesting data for a motion-based VR game. Here, the designer goes through all stages hierarchically to document observations for a specific gameplay level of Reflex. (a) The process begins with the exploration stage, where the designer uses the replayer in anchor mode (highlighted by a pink rectangle) or jumps to a specific chunk of interest (using the blue button in the replayer) to review the performance of all players. For example, upon jumping to chunk 1, the designer observes that some players were significantly delayed due to repeatedly bumping into an obstacle barrel. (b) To identify all players who bumped into the obstacle, she switches to the grouping stage (poses), where she can specify clustering criteria for grouping. She selects "obstacle hits" as the feature and uses k-means as the clustering algorithm, setting the number of groups to two. Next, she moves to the grouping results sub-tab to review the outcome. Upon replaying the grouped data, she observes that players were separated as expected. (c) To further understand why this obstacle was challenging for one group of players, she transitions to the explanation stage. Here, she selects Player 6 from the group with frequent obstacle collisions and Player 3 from the other group. She then replays and compares their performances via a VR point-of-view mode. Intuitively, she notices that Player 6 spent a significant amount of time looking at the controller and attempting to use it to move around the obstacle.

tools (G1). Second, it should enable grouping of players based on criteria to uncover patterns and identify players of interest (G2). Finally, it should support detailed investigation through visualizations of body-motion related data, event anchors to highlight important gameplay moments, and synchronization playback for player comparisons (G3).

To achieve these goals, we propose a top-down hierarchical framework for VR playtest analysis. Guided by principles from human-computer interaction (HCI) on hierarchical interaction and information seeking frameworks [34, 43], our design supports progressive exploration of playtesting data to better understand player experience while fitting naturally into practitioners' existing workflows.

The analytic process begins with the **exploration stage**, which offers a high-level overview of how players as a whole interact with the game. Designers can examine gameplay from multiple camera views or jump to important game moments to form an overall impression (G1, G3). Once an initial understanding is established, designers move to the **grouping stages**, where they can identify subsets of players of interest (e.g., those who get stuck) by using pose-based or gaze-based grouping mechanisms (G2, G3). Finally, in the **explanation stage**, designers can closely examine individual players, or compare pairs, using synchronized playback views with embedded visualizations. This stage enables the investigation of player experiences, potential bugs, and the causes of unexpected behaviors (G3). We describe detailed implementation from Section 5.1 to Section 5.3.

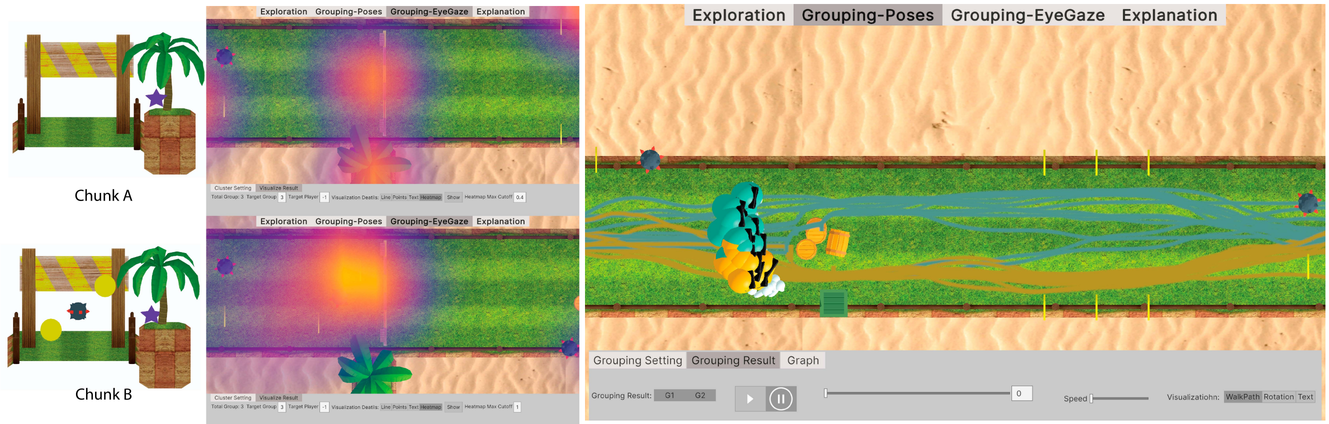
4 System Overview

In this section, we first introduce our illustrative game, *Reflex*, and describe the types of data collected during gameplay. We then introduce three key stages of HieraVisVR and explain how these stages guide designers in reviewing the playtest results (Section 4.2- 4.4). We follow a hypothetical designer, Alice, who aims to evaluate how novice players interact with this game and whether the controls are intuitive for them, as shown in Figure 3. The detailed implementation of each stage of HieraVisVR, illustrated using the Reflex, is described in Section 5, with two additional example demonstrations provided in Section 9.

4.1 Illustrative VR Game: *Reflex*

We use a motion-based VR game, *Reflex*, as an illustrative example. It mimics the gameplay mechanics of the popular Kinect game *Reflex Ridge*. The original game has been used for virtual reality training research [16]. Reflex is a representative motion-centric VR game, requiring players to perform full-body physical movements such as squats, jumps, lateral movements, and reaching gestures to navigate the virtual environment and interact with game objects. These interactions produce rich multimodal data, including hand and body trajectories, eye gaze, and discrete event markers, making it an ideal dataset for demonstrating the analytics workflow of HieraVisVR.

Building on the original gameplay, our VR adaptation requires players to physically move and dodge obstacles while using controllers to collect coins and treasure boxes. The game was played on the Meta Quest Pro. We recruited 27 players with prior experience in VR gaming to complete a three-minute gameplay session. During each session, we logged player movement, eye-gaze data, and in-game performance metrics such as coin collections and obstacle



(a) Use Case 2

(b) Use Case 3

Figure 4: Illustration of Use Cases 2 and 3. (a) Use Case 2 shows two nearly identical chunks being analyzed to uncover design intentions. Using heatmap visualizations of eye-gaze for these chunks, more attention is observed on the bar and the star on the island in chunk A. In contrast, in chunk B, the three elements in front of the bar drew most of the attention, distracting players from the intended design elements. (b) In Use Case 3, designers can use the Grouping stage to categorize players based on their walking paths.

hits. The collected data were exported as CSV files and subsequently loaded into HieraVisVR for analysis and demonstration. Both the illustrative game and the hierarchical visual analytics framework were implemented in the Unity 3D engine [46].

4.2 Use Case 1: Identifying Gameplay Patterns

Figure 3 exemplifies how Alice identifies gameplay patterns using HieraVisVR.

Exploration Stage. Alice begins by using the replayer to observe the gameplay from start to finish several times. She immediately notices that some players fell far behind others and struggled with the first obstacle. Using the replayer's anchor mode, she jumps directly to the chunk containing this obstacle. Based on her observation, she realizes that players repeatedly bumped into obstacles every time they encountered one.

Grouping Stage. To identify whether other players were experiencing the same difficulty, Alice uses the grouping feature by selecting "obstacle hit" as the grouping criterion and setting the total number of groups to two. This reveals a group of players who frequently get stuck. She then uses the group replayer to observe and identify any interesting patterns within this group. The replayer reveals that all players in this group struggled primarily with the first obstacle and spent the most time there. However, she cannot determine why this chunk was challenging for them, prompting her to proceed to the next stage for deeper insights.

Explanation Stage. Switching to the first-person point of view in the replayer, Alice discovers the root cause: these players were looking at their controllers, attempting to use their controllers instead of their bodies for movement, contrary to expectation. This observation clarifies the issue, letting her identify that players are unfamiliar with the movement mechanics.

4.3 Use Case 2: Uncovering Design Intention

This time, Alice is interested in two similar chunks (see Figure 4): chunk A contains a bar and an island with a purple star, while chunk B is almost identical, except that there are three additional objects in front of the bar. Alice wants to determine whether players noticed the purple star in these chunks, as it aligns with her design intention. She selects the grouping stage focused on eye-gaze data, using fixation sequences as the grouping criterion for chunk A. By enabling the heatmap visualization, she observes gaze patterns across different groups. She repeats this process for chunk B. Her findings reveal that the three objects in front of chunk B distracted players from noticing the island with the purple star. She documents this observation for further design refinement.

4.4 Use Case 3: Analyzing Players' Decision-Making

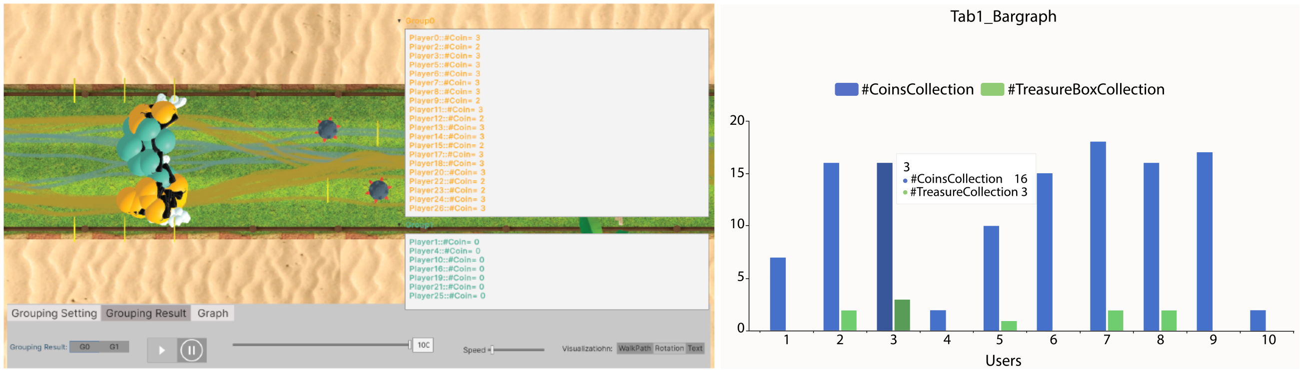
Understanding player preferences at key game moments is also part of Alice's analysis. Figure 4 shows an example. This time, she focuses on chunk 2, where players faced an obstacle, chose a side, and aimed to maximize coin collection. She moves to the grouping pose stage, selects a chunk range from 1 to 3, sets the grouping criteria to walking paths, and targets two groups. After grouping, players are divided into two color-coded groups. Alice can then replay either group's performance, observe their behavior, and record her findings.

5 Hierarchical Visual Analytics System

Our framework was implemented using the Unity 3D engine, with clustering algorithms, including K-Means, HDBSCAN, and subspace clustering [54], implemented in Python. A socket-based connection was used to enable communication between Unity and Python.

5.1 Exploration Stage

In response to G3, we designed a replayer with two anchor modes to highlight important gameplay moments: the level-anchor replayer and the event-anchor replayer. As a game level consists of a



```
Player0::#Coin= 3
Player2::#Coin= 2
Player3::#Coin= 3
Player5::#Coin= 3
Player6::#Coin= 3
Player7::#Coin= 3
Player8::#Coin= 3
Player9::#Coin= 2
Player11::#Coin= 3
Player12::#Coin= 2
Player13::#Coin= 3
Player14::#Coin= 3
Player15::#Coin= 2
Player17::#Coin= 3
Player18::#Coin= 3
Player20::#Coin= 3
Player22::#Coin= 2
Player23::#Coin= 2
Player24::#Coin= 3
Player26::#Coin= 3
```

```
Player1::#Coin= 0
Player4::#Coin= 0
Player10::#Coin= 0
Player16::#Coin= 0
Player19::#Coin= 0
Player21::#Coin= 0
Player25::#Coin= 0
```

Figure 5: Illustration of the Grouping Stage (Poses). Left: Designers can examine grouping results in text or through the group replayer. Right: Designers can plot selected gameplay features in a separate window.

sequence of chunks, the level-anchor replayer focuses on chunk-specific events within the game level. For example, the Reflex game consists of eleven chunks, represented as eleven buttons beneath the slider in the level-anchor replayer (see Figure 3(a)). Similarly, the event-anchor replayer encodes important predefined events recorded during gameplay, such as player interactions with game objects, significant game movements, and player task execution. For example, the treasure box collection is an event specified by the designer in the Reflex game, as it provides players with speedy effects for a limited time. Moreover, when replaying for a large number of players, overlapping players in the replay becomes unavoidable. To address this issue, we added a "Transparent Others" option, allowing designers to focus on a specific player of interest during massive replays (see Figure 6).

In addition to replaying game performance, body-motion-related data is crucial to provide designers with a better understanding. Since Reflex captures the player's body movement and eye gaze trajectory, we explore the design space for visualizing multidimensional data tailored to the purpose of this stage. This includes walking path visualization, walking path heatmap, eye gaze visualization, and head rotation.

The walking path visualization and walking path heatmap are designed to display the trajectory of each player with a different color (see Figure 1(a)). These visualizations help designers understand player decisions at specific moments or identify the most frequently visited areas. Eye gaze visualization is designed to reflect the sequence of fixations and their duration, represented by lines and points. This can reveal player attention during gameplay. Head rotation visualization highlights players' head movement. As shown in Figure 6 (right), the amount of head rotation is represented by an arc shape, with orange indicating a leftward rotation and blue indicating a rightward rotation.

To offer more flexibility in this replay, we have included a camera control panel to support basic control (e.g., zoom in and out, translation, and rotation) as well as camera view switching during the replay. This stage aims to provide game designers with an overview of player performance in a game and to enhance their understanding of motion-based game data through visualization. To achieve this, we incorporate three key components: a replayer with anchor modes, a camera control panel, and motion-based visualization features.

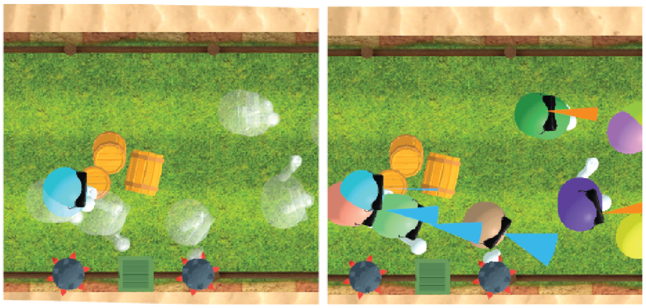


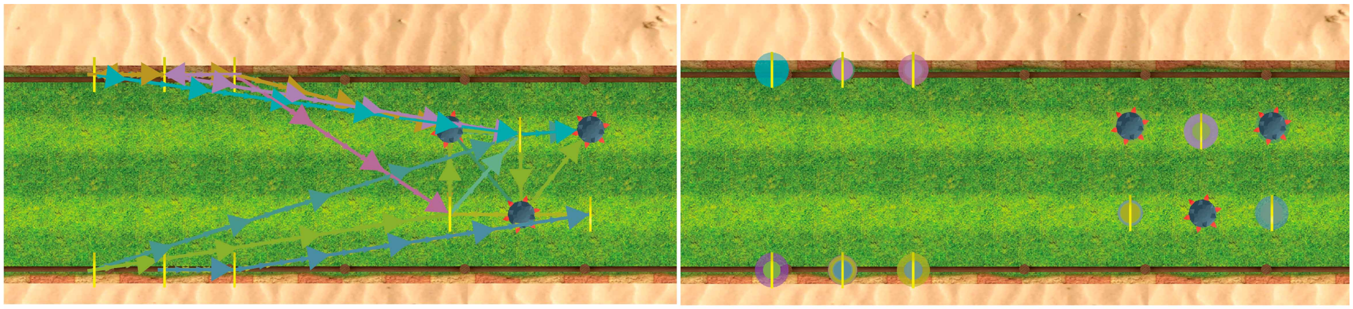
Figure 6: In the exploration stage, designers can either highlight a player of interest (left) or display all players during the replay, with rotation visualization (right).

5.2 Grouping Stage

The goal of this stage is to help designers identify and extract similar gameplay patterns for their analysis. We separate this stage into two categories: poses and eye gaze, as these two types of data represent different aspects of gameplay behaviors.

Grouping Stage (Poses). This focuses on general player behaviors as reflected in players' walking paths and gameplay performance. It consists of three sub-tabs: Grouping Settings, Grouping Results, and Graph. Designers begin their analysis in the Grouping Settings tab by selecting specific game performance-related features (e.g., number of coin collections) they want to explore for patterns. Here, they can specify their preferred number of groups and choose from various clustering algorithms, including KMeans, HDBSCAN, and subspace clustering [54]. HieraVisVR can support additional clustering algorithms as needed. For more focused analysis, designers can narrow their investigation to a specific game moment by defining chunk ranges.

After the cluster setting is complete, HieraVisVR will preprocess the gameplay data based on the selected features, which typically include a mixture of temporal data (e.g., poses) and discrete data (e.g., game duration). Inspired by previous action recognition frameworks [15], HieraVisVR uses a covariance descriptor to represent



Line Visualization

Point Visualization

```
▾ Group1
player0 with ['CoinsOnly1'_'Coin4', 'CoinsOnly1'_ 'Coin6', 'CoinsOnly1'_ 'Coi
player6 with ['CoinsOnly1'_'Coin2', 'CoinsOnly1'_ 'Coin4', 'CoinsOnly1'_ 'Coi
▾ Group2
player2 with ['CoinsOnly1'_'Coin1', 'CoinsOnly1'_ 'Coin2', 'CoinsOnly1'_ 'Coi
player3 with ['CoinsOnly1'_'Coin1', 'CoinsOnly1'_ 'Coin2', 'CoinsOnly1'_ 'Coi
player7 with ['CoinsOnly1'_'Coin1', 'CoinsOnly1'_ 'Coin3', 'CoinsOnly1'_ 'Coi
player8 with ['CoinsOnly1'_'Coin1', 'CoinsOnly1'_ 'Coin2', 'CoinsOnly1'_ 'Coi
player11 with ['CoinsOnly1'_'Coin1', 'CoinsOnly1'_ 'Coin2', 'CoinsOnly1'_ 'Co
player12 with ['CoinsOnly1'_'Coin1', 'CoinsOnly1'_ 'Coin2', 'CoinsOnly1'_ 'Co
▾ Group3
player1 with ['CoinsOnly1'_'Coin1', 'CBSwipeUp1'_ 'Coin2',]
player4 with ['CoinsOnly1'_'Coin5', 'CoinsOnly1'_ 'Coin4', 'CBSwipeUp1'_ 'C
player5 with ['CoinsOnly1'_'Coin2', 'CoinsOnly1'_ 'Coin3', 'CoinsOnly1'_ 'Coi
player10 with ['CBSwipeUp1'_'Coin2', 'CBSwipeUp1'_ 'Bomb3', 'CBSwipeUp
player13 with ['CoinsOnly1'_'Coin5', 'CoinsOnly1'_ 'Coin6', 'CBSwipeUp1'_ '
```

Text

Heatmap

Figure 7: Different styles of eye gaze visualizations are shown for a selected group in the grouping stage. Each player in the group is represented by a unique color. The arrows in the line visualization indicate the sequence of gaze movements, while the size of the circles in the point visualization reflects gaze duration. Text provides detailed grouping sequences, and the heatmap reveals the overall distribution of eye gaze patterns.

each player's body trajectory. Additional gameplay data, such as normalized duration, is appended to the descriptor, forming an embedding vector that encodes both trajectory and gameplay information. This vector is then processed by a Python server to execute the selected grouping algorithm. The results are then displayed in the Grouping Results tab, where designers can examine the visual and textual representations of different groups. A replayer with walking path and rotation features is encoded for designers to investigate, and resulting groups are color-coded for easy distinction (see Figure 5 (left)).

Additionally, in the Graph tab, designers can plot gameplay performance data, similar to how they would in Excel or any third-party tool. In Figure 5 (right), we can see that the designer has plotted players' coin collections alongside treasure box collections.

Grouping Stage (Eye Gaze). This grouping stage focuses on eye gaze data, where designers can analyze players' interests and attention during gameplay. Similar to the grouping stage (poses), designers initiate the grouping by selecting eye gaze features to explore, specifying the number of groups for the results, and choosing a clustering algorithm. For gaze feature options, we focus on the sequence of game objects as gaze fixations, along with other characteristics of these fixations (e.g., average fixation duration), which will be extracted from raw gaze data once the cluster settings are complete. Then HieraVisVR constructs embedding vectors from the sequence of fixation locations using Sequence Graph Transform [38]. Any additional eye gaze features are normalized and appended to the end of the embedding vectors.

Moreover, as eye gaze features differ from gameplay performance and walking paths by nature, we use a different set of clustering techniques. We support Agglomerative Clustering [32], FasterMSC clustering [21], PAM clustering [40, 41] for eye gaze analysis. Unlike the previous stage, the visualization of eye gaze results is static without the replay functionality. Instead, the results are presented through text descriptions alongside visual representations using lines, points, and heatmaps (see Figure 7). Designers can view data for a specific player, a group, or an entire population. Generally, point and line visualizations are useful for examining individual player eye gaze data as they denote gaze duration and sequence of fixations. Heatmaps help quickly grasp the distribution of attention and gaze patterns, while the text provides insights into gaze patterns.

5.3 Explanation Stage

This stage helps designers examine a specific player or a pair of players in-depth to understand their performance. Similar to the exploration stage, it features a replayer with anchor modes, a camera switch panel, and motion-based visualization features specifically tailored for this stage. The camera switch panel is enhanced with options for first-person point of view and near-first-person point of view, allowing designers to observe the gameplay from the players' perspective.

As the replayer in this stage focuses on one or two players, it includes three embedded visualizations to assist designers in analyzing eye gaze and movement data in detail: walking paths, eye gaze paths, and eye gaze balls. The walking path visualization

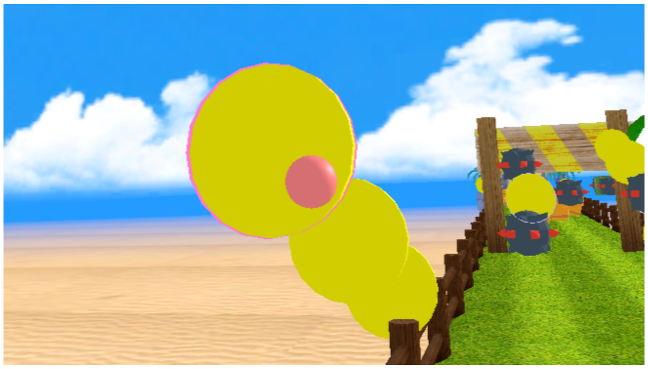


Figure 8: Gaze ball visualization.

is similar to the one in the previous stage, showing the player's movement trajectory.

The eye gaze path visualization highlights game objects as circles, where the size of each circle represents the duration of the gaze. These circles are connected by lines to indicate the sequence of gazes. The eye gaze ball visualization features a red ball whose position updates in real-time during the replay, representing the player's actual gaze position dynamically (see Figure 8).

In addition, this stage supports a side-by-side comparison, as shown in Figure 1(d). In this view, the primary player is displayed in a solid color within their camera perspective, while the other player appears as transparent, allowing for easy differentiation. Real-time gameplay performance metrics (e.g., number of coins collected) are also dynamically updated for the selected players, providing immediate insight into their actions and progress.

6 Expert User Study

We conducted an expert user study with professional game testers, focusing on two research objectives: (1) How does the three-stage workflow support playtest analysis? and (2) How does each visualization help playtest analysis? We designed realistic tasks mimicking actual playtesting workflows and collected qualitative feedback and quantitative measures to assess the effectiveness and usability of the HieraVisVR.

6.1 Setup

Participants. We recruited five professional game testers through our university's alumni network to evaluate our system (four males and one preferred not to disclose the gender; ages ranged from 20 to 26). This group is different from the participants of the formative study. Two participants had 2 years of VR game development and testing experience (E3, E4); two had 3–5 years of experience (E2, E5); and one had over 5 years of experience (E1). Table 6 shows a summary of their experience. Most participants had used screen recording tools to capture the gameplay. A few mentioned writing their own scripts to test their games. All participants had experience using testing tools provided by their companies to evaluate games.

Procedure. The entire study lasted approximately 90 minutes. We began with a 20-minute introduction and a warm-up session to familiarize participants with the system. Following this, participants began their evaluation with our system, using Reflex as the example game for assessment. Our formative interviews revealed that participants typically have specific questions in mind before testing. To mimic this workflow, we designed a VR testing questionnaire for participants to guide them in evaluating a game. The details of the testing questionnaire are provided in Table A1 in the Appendix A. Participants were also encouraged to explore freely if they had additional questions in mind. This evaluation section took about 30 minutes. Finally, participants engaged in a 40-minute interview to discuss the usability and effectiveness of our system. In this interview, they also completed Likert scale questions (partially adapted from the SUS questionnaire [2]). Each participant was compensated $75 for their time.

Table 6: Participants' professional backgrounds for the expert study.

	Expert Experience
E1	QA tester of desktop and VR games (6 years of experience)
E2	Design director for indie games (3 years of experience)
E3	QA tester at a company (2 years of experience)
E4	Level designer at a small indie game studio (2 years of experience)
E5	Game designer and programmer (3.5 years of experience)

7 General Feedback and User Ratings

Figure 9 shows an overview of the questions and ratings given by the participants for HieraVisVR. In Q1, "The system assists me in evaluating the quality of the game" ($M = 4.60$, $SD = 0.55$). All participants agree that our system is "feature-rich" and helps them inspect different aspects of a game. E2 commented *"it isolates very specific key events, which is a big thing in game design. It helps me identify where players got stuck."* E3 mentioned *"This system aggregates data very well. With heatmap and walking path visualizations, I don't need to watch individual gameplay to understand game difficulty."* E4 expressed *"I could see how players interacted with the game and quickly identified places where they could go wrong, and where they go right. So if I were to want to increase the quality of the game or the effects of it on specific target audiences, it would be extremely useful."*

In Q6, "I felt confident about understanding the quality of a game" ($M = 4.00$, $SD = 0.00$), most participants felt confident in using our system to evaluate the quality of any motion-based game. They found that our system captures the basic testing needs and makes it easier to compare and contrast different gameplay experiences. They mentioned that VR gaming experiences are highly individualized and, as designers, they prefer to play the game themselves first. Afterward, they would use our system to complement their testing process.

System Usability. In response to Q2, "I found the system very cumbersome to use" ($M = 2.60$, $SD = 0.55$), most participants disagreed or remained neutral, indicating that our system is not particularly cumbersome due to its straightforward UI. However, some usability issues were highlighted. Almost all participants were unfamiliar with the grouping algorithms provided, and they had to refer back to the documentation given in the tutorial, or showed uncertainty when selecting the appropriate grouping algorithm to use. They expressed a desire for more detailed documentation on that. E5 suggested adding tooltips or pop-up boxes to reduce the need to refer back to documentation for understanding.

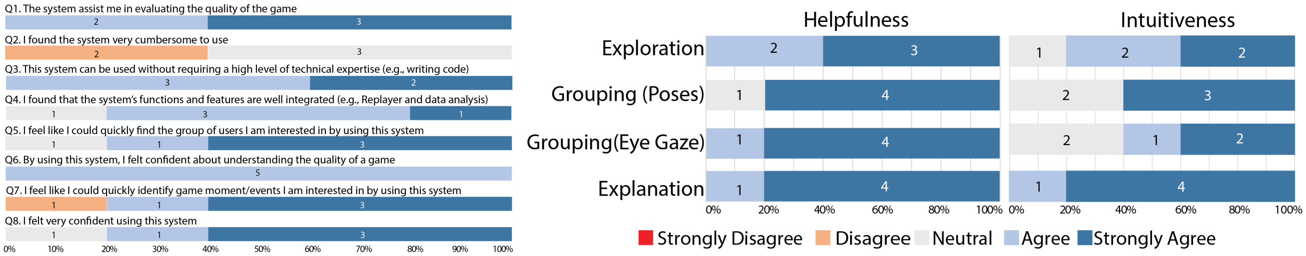


Figure 9: Stack bars showing the participants' ratings on HieraVisVR in the expert interview.

Another issue was overlapping players in the replay scene during the exploration stage, making it difficult to identify players of interest. Although a selection tool was provided to allow designers to click and view player name tags, participants still found this process somewhat cumbersome. One suggestion was to display all players in a collapsible list panel, allowing designers to click and highlight a specific player. Another suggestion involved using hoverable name tags to highlight a player when hovering automatically over overlapped players directly in the replay scene (E3 and E5).

Technical Expertise. When answering the question "This system can be used without requiring a high level of technical expertise (e.g., writing code)." (Q3, $M == 4.40$, $SD = 0.55$), all participants agree or strongly agree, generally because our system does not require any coding input. E2 said *"I can imagine designers from our team can figure this out and work through it without knowing too much about grouping algorithms such as k-means."* and he added *"The only downside is that I wish there were tooltips for explaining clustering algorithms and metrics."*

In Q8, "I felt very confident using this system" ($M = 4.40$, $SD = 0.89$), most participants found it easy to understand and use after a brief introduction. However, E2, who rated it a 3.00, mentioned that there was a learning curve at first when it came to understanding different grouping algorithms. After using the system for a while, he became confident in applying it to different games.

System Functionality and Features. In response to Q4, "I found the various functions/features (e.g., replayer and camera view) in this system well integrated" ($M = 4.00$, $SD = 0.71$), and Q7, "I feel like I could quickly identify game moments/events I am interested in by using this system" ($M = 4.60$, $SD = 0.55$), most participants found the replayer with level and event anchors particularly useful. For example, some participants commented, *"I can jump to a point where people get stuck"* or *"see how people generally progress across a chunk"* (E2, E3, and E4). However, two participants expressed a desire for more finite control over the timeline when using the anchor. E5 noted, *"When I use the event anchor, I expect to see the moment before an event happens."* Moreover, E2, who gave a 3 in Q4, commented on the missing flexibility to change the camera view in the grouping stage.

In Q5, "I feel like I could quickly find the group of players I am interested in by using this system" ($M = 4.40$, $SD = 0.89$), most participants agreed that by grouping players based on their performances, they could quickly identify players of interest. E5, who rated it a 3, expressed a desire for more advanced metrics to guide the grouping of multiple player performance features. He suggested adding flexibility to combine contradictory features into a single metric; for example, with a higher coin collection being favorable and a shorter gameplay duration being preferred. By providing such metrics, the system could better reflect overall player performances.

Workflow Comparison. In answering the question, "If you have to evaluate the game, would you use your existing workflow (software), or would you consider using this system?", four out of five participants preferred integrating our system into their existing workflow, as the workflow proposed by our system aligns well with their current practices. They found the visualizations provided by our system particularly valuable for analyzing large datasets. E5 mentioned, *"There is still a need to test by myself first and directly talk with some target players to gather initial insights."* However, he believed that after collecting data, using our system for visualization was helpful and easier than his current workflow. With some adjustments, he noted that the system could even help him gather statistics to present to clients. In contrast, E3 mentioned that he prefers using our system primarily when analyzing a large amount of playtesting data, as our system allows for comprehensive observation of player behaviors in a single view.

8 Discussion

8.1 How does the Three-Stage Workflow Support Playtest Analysis?

The three-stage workflow was received positively, with the explanation stage most preferred. In rating the helpfulness and intuitiveness of each stage (see Figure 9), the explanation stage received the highest ratings, followed by the exploration stage, and then the grouping stage. These ratings reflect participants' typical debugging and analysis habits–examining gameplay on a case-by-case basis. As E1 explained, *"Usually, I would go directly to the places where my code tends to break to ensure smooth gameplay first. After debugging, I will look at the entire gameplay to gain a broader understanding."* Participants felt that the current implementation of the explanation stage, including the existing camera views and visualizations, met most of their needs for close comparison. As E2 noted, *"Even though I don't often examine players side-by-side, this is exactly what I expect."* He added, *"This could be useful for comparing two different play styles in a strategic game."*

Exploration stage supports not only pattern discovery, but also client communication and cross-platform analysis. Most participants were positive about the exploration stage, noting that it provides *"a greater picture of how players interact with the game,"* helps them understand play styles, and offers *"a starting point to* *create useful groups."* As E1 mentioned, *"I want to test on other headsets or hardware and then put them together to see whether there is any gameplay difference across different platforms."* Others saw value in using the aggregated gameplay statistics in client-facing presentations: *"I will use the statistical results from the survey and this aggregated gameplay data to persuade my client."* (E5)

Grouping stage is more helpful for performance analysis if the grouping result presentation is more informative. Most participants were introduced to grouping algorithms for the first time. They found this grouping method effective for quickly categorizing players, such as identifying the best and worst performers. They agreed that the grouping feature options were diverse and made it easy to visualize any in-game data. However, E3 and E5 wanted more detailed documentation explaining the differences between grouping algorithms, including their pros, cons, and general use cases. E5 suggested adding the ability to sort groupings based on performance metrics (e.g., lowest performance) for more efficient performance analysis.

On one hand, participants appreciated the current graphing feature as it allowed them to observe the general distribution of key metrics—such as gameplay duration and player rotation—which were difficult to interpret from replaying the session frame by frame. On the other hand, their expectations for the grouping stage extended beyond simply replaying gameplay. They expressed a desire for more informative and structured presentations of the grouping results. As E3 noted, *"I'm more interested in seeing the grouping results in a summary so that I can write my report more easily."*

Grouping by eye gaze enables deeper behavioral diagnosis but highlights the need for guidance. All participants reported that grouping by eye gaze is a helpful method for analyzing players' behaviors, especially for understanding the root causes of in-game struggles. As E4 commented, *"I can understand how they played the game from the eye gaze perspective of the players who got stuck."*

However, since eye tracking is a relatively new technology in VR, this feature also revealed a knowledge gap. Participants were still learning how to utilize eye gaze data effectively and expressed uncertainty about the analytical features. *"I am not sure what to expect when grouping based on the average duration of fixation,"* said E3, who gave a low rating for intuitiveness for the grouping by eye gaze feature. This highlighted a clear need for better documentation. He added, *"I can imagine this could potentially be helpful for a shooting game to see whether players identify potential enemies in the bush."*

8.2 How does each Visualization help Playtest Analysis?

Walking path visualization supports behavioral comparison, but its relevance varies with the analysis stage. The general usage of walking paths is in the grouping stage visualization, which reveals how grouped players take different paths, highlighting distinctions between skillful and less skillful players, and showcasing their different approaches to the game. As E2 noted, *"It serves the purpose of helping me filter out the group of people I'm interested in from the rest."* To better support result interpretation, E5 suggested adding an average walking path for each group to better summarize overall behaviors.

In other stages, the feature's role shifted. During the initial exploration stage, it served as a supplement to heatmaps, helping to analyze players' decision-making changes and, with a highlighting option, focus on specific players when paths overlapped. Interestingly, by the final explanation stage, participants had differing opinions based on their goals. While three participants wanted the walking path for detailed comparisons, the other two felt it was no longer their focus, as they had already used it to draw conclusions in earlier stages.

Heatmaps are useful for highlighting both positive and negative areas, while the rotation visualization mostly serves as a supplementary cue. Participants found heatmaps useful for observing high-level gameplay patterns. They commented that heatmaps were effective in "highlighting both positive and negative areas" (i.e., dead spots) and in revealing players' intentions by showing where they got stuck. The primary critique was a usability issue noted by E4, who found the default colormap could blend with the game environment and suggested an option to change the color pattern for better contrast. In contrast, opinions on the rotation visualization were more diverse. While some testers found it to be a useful cue for tracking general directions, others felt it was limited or made redundant by the more precise and insightful data from the eye gaze grouping stage. This highlights how the utility of a general visualization can be debated when a more specific visualization is also available.

A preference for real-time feedback over static summaries is observed. When analyzing player focus in the final explanation stage, participants strongly preferred the dynamic "gaze ball" visualization over the static "gaze path" because it revealed the exact game element a player was looking at in real-time as the replay progressed. This live feedback allowed for deeper insights, as E2 noted, *"I was also able to see the distance of how far they were looking during gameplay."* To further enhance this real-time analysis, E1 suggested that if object names, like 'coin 1,' could pop up when looked at, it would make it *"clearer for me to draw conclusions."*

In contrast, opinions on the static gaze path visualization were mixed, largely because some felt its functionality was made redundant by the more immediate gaze ball. Others, however, saw potential use cases for this summary view, particularly with a top-down perspective. E3 highlighted its context-dependent value, remarking that in the final explanation stage, the path was *"not overly cluttered... so I could analyze it without needing to rewatch the entire replay,"* suggesting its utility as a quick, summary feature when a full replay is not needed.

9 Additional Examples

To illustrate the general applicability of our framework, we adapted it to two different applications: a VR fire evacuation training and a VR escape room experience. These applications serve as additional examples to demonstrate how our framework could be integrated across VR experiences and genres with different user interactions, game mechanics, and objectives. They are not evaluated at the same depth as our illustrative game, Reflex.

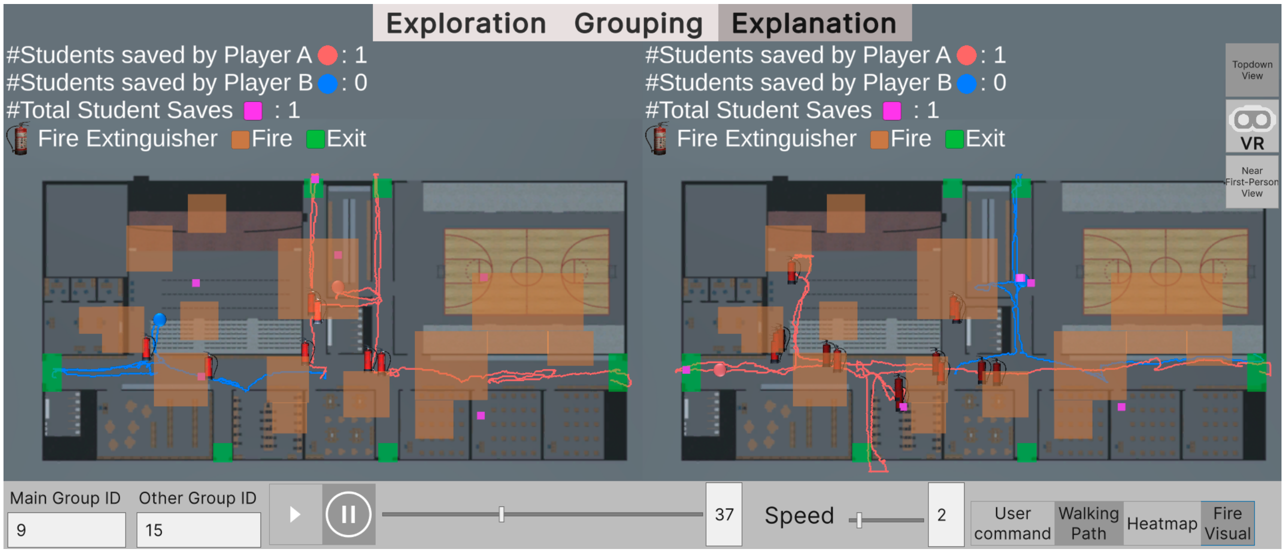


Figure 10: A virtual fire evacuation training application [26] implemented with our system. A comparison playback of two pairs of players and their rescue strategies is shown. In addition to walking path visualization and heatmap visualization, our framework can incorporate customized annotations for specific applications. Here, we visualize fire blocks in the scene. Although two pairs of players distribute tasks equally among themselves, we can still observe their different rescue strategies.

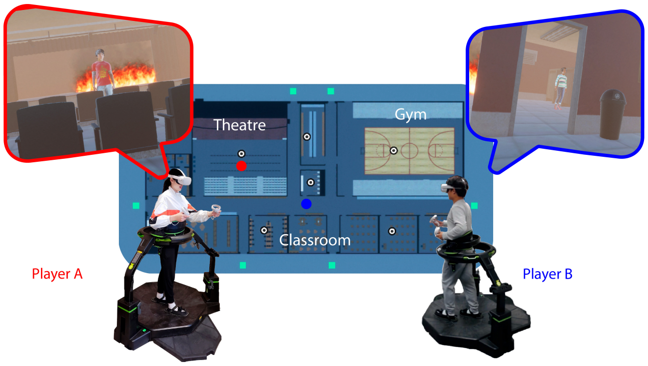


Figure 11: The original VR fire evacuation training application [26].

We first applied our framework to a VR fire evacuation training application, which is based on a previous work on a virtual reality fire evacuation training drill [26]. In their work, the researchers explored how collaborative behaviors changed as they increased the difficulty of fire evacuation training scenarios. They conducted the experiment in VR and recruited 54 players grouped into 27 pairs. For each pair of players, they were asked to wear a VR headset and walk on the VR treadmill to complete a 10-minute gameplay session. Their goal was to guide five virtual students out of a school where a simulated fire emergency occurred. Initially, one student is located in the theater, another is in the gym, and the remaining three are in the classroom (see Figure 11). Several in-game metrics were collected for evaluation, including completion times, players' trajectories, extinguisher usages, and command times.

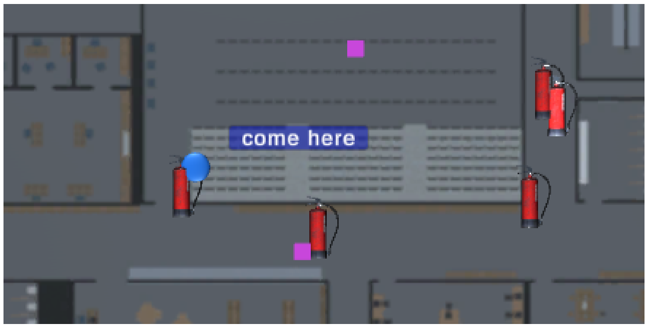


Figure 12: Player commands shown in the explanation stage.

9.1 VR Fire Evacuation Training

Figure 10 provides a top-down view of the replay during the explanation stage, offering a side-by-side comparison of two pairs of players (Group 9 and Group 15) participating in a fire drill training scenario at the same difficulty level.

Replayer. In this replay, each player is represented by a different color, allowing us to observe their rescue strategies and collaboration behaviors through the replay. The players from Group 9 (left) decided to split the rescue tasks by location: one player was responsible for the left side of the building, while the other managed the right side. Both players prioritized gathering students together before leading them to the exit. On the other hand, Group 15 (right) employed a similar strategy; however, player A preferred rescuing students one by one and leading them to the nearest exit. As player A completed their tasks earlier, they assisted player B in rescuing the last student.

Additional Visualization. In addition to the walking path visualizations, we provide custom annotations for this application, such

(a) Task 1 (b) Task 2 (c) Task 3

Figure 13: Gaze heatmaps showing how players visually explored the room while searching for three key items. (a) In Task 1, attention was spread across the bookshelf, long table, and desk areas. (b) In Task 2, the gaze focused on higher regions of the bookshelf. (c) In Task 3, the gaze concentrated on darker areas such as the drawers and the region under the table.

as extinguisher usages and players' commands. In Figure 10, we can observe that fire extinguishers were displayed when players first encountered the fire. As Figure 12 shows, player B issued a command to a virtual student in the theater, instructing him to follow. Please refer to the supplementary video for more details.

9.2 VR Escape Room

This game mimics the escape room genre, a popular game category on Steam VR. At the beginning of this game, players are placed in a room filled with various pieces of furniture, which is divided into four distinct areas, as shown in Figure 14. They are asked to solve three poem-based puzzles, identify the locations of three hidden items, and search for the items in the room in order to escape. Unlike the prior illustrative motion-based VR game, Reflex, where body movements and rapid responses are the main focus, this game features visual scanning and exploratory searching. We recruited 14 participants to play the game, each completing it within five minutes. During gameplay, we collected in-game metrics such as completion time, player trajectories, and eye gaze data.

Eyegaze Grouping. Gaze patterns are a primary focus in this type of game as they reveal player behaviors and inform designers in adjusting the game's difficulty level. HieraVisVR supports game designers in understanding how players visually scan the environment to complete tasks. Figure 13 presents heatmaps of gaze patterns throughout the searches for three key items, revealing how players visually explored the room. In Task 1, where the hint directed players to search near books, attention was dispersed across multiple areas, including the bookshelf, the surface of the office desk, and the pile of books in the corner of the table. In Task 2, where a more direct hint to search in higher areas was provided, gazes were more concentrated in elevated regions of the bookshelf. In Task 3, the hint suggested searching in dark places; accordingly, the heatmap highlights regions near the corner of the office drawer and underneath the long table, where players spent more time searching.

HieraVisVR also assists designers in identifying gameplay patterns. For example, in Task 3, two distinct search behaviors were observed. Players who correctly interpreted the puzzle hint focused their search on lower areas such as the drawers and the region under the table. They typically found the item in one run. In contrast, players who misunderstood the underlying clue searched in areas where they believed items would most likely appear, resulting in repeated back-and-forth exploration. Figure 15 shows visualizations of gaze trajectories of different search behaviors by HieraVisVR.

Figure 14: Top-down view of Escape Room.

10 Limitations and Future Work

Bridging Self-Directed Testing and Data Analysis. Both formative and expert studies show that VR designers prefer to explore applications or games independently during playtesting. While HieraVisVR simplifies post-playtesting data analysis, there is strong interest in advanced visualization and analytics to support self-directed testing. Inspired by previous immersive, situated analytic research [14, 18], we propose extending the framework to enable real-time annotation during gameplay, with annotations visible during playback. In this way, designers could annotate important observations while watching players in action. This kind of direct, multimodal testing can further enhance the depth and efficiency of analysis.

Intelligent Suggestions for Algorithm Selection. When presenting advanced grouping techniques to game designers, it is important to recognize that most designers may not have a strong mathematical background. Therefore, more detailed explanations


(a) Efficient Search Trajectories (b) Repetitive Search Trajectories

Figure 15: Gaze trajectories of players with (a) efficient search behaviors and (b) repetitive search behaviors.

are needed, possibly including examples of each grouping technique. For instance, if a designer wants to group players based on their performances, the system should suggest an appropriate algorithm based on the features he has selected, rather than requiring him to investigate which algorithm would be the most effective. This would streamline the process and make it more accessible to designers with varying levels of expertise. Moreover, game designers often want to investigate design intentions for specific locations using eye gaze features. A prediction model can be used to detect anomalies, such as whether a particular area is intentionally eye-catching or unintentionally distracting to players [1].

Replayer with the Anchor Mode. Participants emphasized the importance of the event anchors, which allow them to instantly jump to important gameplay moments and save them time watching entire replays. Yet, many agreed that the functionality of this replayer still needs to be improved. We discuss three ideas.

First, the type of events in the replayer should be flexible. In a large complex game with a substantial amount of events (e.g., death, shot), showing all events in the main timeline could overwhelm designers. There should be flexibility in allowing the designer to choose what types of events to anchor to and display, tailoring the replayer to their specific analytical needs. Second, the replayer progress bar can be individualized. Currently, the replayer has a single progress bar that controls all players simultaneously. For comparing two players in the explanation stage, there could be two sets of event anchors, one for each player, so the designer can focus on one player as the main player and view the player's events. Third, the types of events should be customizable and controllable. One designer mentioned that, although rare, he would like the ability to redefine an event (e.g., changing its start or end frame) or combine two events, with the replayer adapting to those changes. As Participant P5 noted, *"Sometimes I want to see a few frames before that event happens because that is the moment I really care about."* The rotation visualization received mixed feedback during our evaluation. While some participants found it useful for understanding general directional trends, others felt it was limited or overshadowed by the more precise eye gaze data available in the grouping stage. In future work, we plan to explore alternative encodings, such as directional arrows or alternative color schemes, to better represent head rotation.

Incorporating Emotion-Aware Metrics. Our system currently focuses on motion-related gameplay data, such as locomotion, eye-tracking, and pose, which are crucial for motion-centric playtesting. While this provides a strong foundation, we acknowledge that a comprehensive understanding of player experience can benefit from other data types. For future work, we could extend our system's capabilities to incorporate emotional responses (e.g., electroencephalography and heartbeat data) by integrating with external devices. More importantly, these new data streams will largely present as time-series data, a format our current approach is already well-equipped to visualize and group. For discrete physiological responses, such as enjoyment scale ratings, our system can readily graph these within the existing grouping stage.

Expanding the Expert Study Participant Pool. In the current expert study, we invited five game testers and conducted in-depth interviews based on the Reflex gameplay data. Expanding the study to include a larger and more experienced group of VR testing experts, along with additional VR applications such as the fire evacuation training example, would help validate whether the observed insights generalize to other VR applications. With a larger participant pool, we could also conduct statistical analyses to further assess the effectiveness of the proposed hierarchical playtesting analysis workflow across a broader range of VR scenarios.

11 Conclusion

This work explores visual analytics for playtesting in VR motion-based games. We interviewed 30 VR practitioners to identify designers' challenges and propose HieraVisVR, a hierarchical visual analytics framework for top-down playtesting analysis. Designers can visualize motion data, anchor key moments, group players, and compare behaviors. We demonstrate HieraVisVR on two motion-based VR games and a training application, evaluating its usability through an expert study. Expert feedback validated the three-stage workflow's effectiveness, with participants noting that the system helps them "identify where players got stuck" and provides "a greater picture of how players interact with the game." These insights can help refine HieraVisVR and future VR playtesting research.

Acknowledgments

We are grateful to the anonymous reviewers for their constructive comments. This project was supported by the National Science Foundation (award numbers: 2430673, 2418236, and 1942531). Lap-Fai Yu is supported by the GMU's IDIA P3 Faculty Fellowship on this project.

References

[1] Elham Azizi and Loutfouz Zaman. 2024. AstroBug: Automatic Game Bug Detection Using Deep Learning. *IEEE Transactions on Games* 16, 4 (2024), 793–806. doi:10.1109/TG.2024.3402626

[2] John Brooke et al. 1996. SUS-A quick and dirty usability scale. *Usability evaluation in industry* 189, 194 (1996), 4–7.

[3] Wolfgang Büschel, Anke Lehmann, and Raimund Dachselt. 2021. Miria: A mixed reality toolkit for the in-situ visualization and analysis of spatio-temporal interaction data. In *Proceedings of the 2021 CHI conference on human factors in computing systems*. 1–15.

[4] Anqi Cao, Xiao Xie, Mingxu Zhou, Hui Zhang, Mingliang Xu, and Yingcai Wu. 2023. Action-Evaluator: A Visualization Approach for Player Action Evaluation in Soccer. *IEEE Transactions on Visualization and Computer Graphics* 30, 1 (2023), 880–890.

[5] Kathy Charmaz. 2008. Constructionism and the grounded theory method. *Handbook of constructionist research* 1, 1 (2008), 397–412.

[6] Wei Chen, Tianyi Lao, Jing Xia, Xinxin Huang, Biao Zhu, Wanqi Hu, and Huihua Guan. 2016. Gameflow: narrative visualization of NBA basketball games. *IEEE Transactions on Multimedia* 18, 11 (2016), 2247–2256.

[7] Zhutian Chen, Qisen Yang, Jiarui Shan, Tica Lin, Johanna Beyer, Haijun Xia, and Hanspeter Pfister. 2023. iball: Augmenting basketball videos with gaze-moderated embedded visualizations. Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems, 1–18.

[8] Dazhen Deng, Jiang Wu, Jiachen Wang, Yihong Wu, Xiao Xie, Zheng Zhou, Hui Zhang, Xiaolong Zhang, and Yingcai Wu. 2021. Eventanchor: Reducing human interactions in event annotation of racket sports videos. Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems, 1–13.

[9] Anders Drachen and Alessandro Canossa. 2009. Towards gameplay analysis via gameplay metrics. Proceedings of the 13th international MindTrek conference Everyday life in the ubiquitous era, 202–209.

[10] Anders Drachen, Pejman Mirza-Babaei, and Lennart E Nacke. 2018. *Games user research*. Oxford University Press.

[11] Brandon Drenikow and Pejman Mirza-Babaei. 2017. Vixen: interactive visualization of gameplay experiences. In *Proceedings of the 12th International Conference on the Foundations of Digital Games*. 1–10.

[12] Meng Du and Xiaoru Yuan. 2021. A survey of competitive sports data visualization and visual analysis. *Journal of Visualization* 24 (2021), 47–67.

[13] Nour Halabi, Günter Wallner, and Pejman Mirza-Babaei. 2019. Assessing the impact of visual design on the interpretation of aggregated playtesting data visualization. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*. 639–650.

[14] Sebastian Hubenschmid, Jonathan Wieland, Daniel Immanuel Fink, Andrea Batch, Johannes Zagermann, Niklas Elmqvist, and Harald Reiterer. 2022. Relive: Bridging in-situ and ex-situ visual analytics for analyzing mixed reality user studies. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. 1–20.

[15] Mohamed E Hussein, Marwan Torki, Mohammad A Gowayyed, and Motaz El-Saban. 2013. Human action recognition using a temporal hierarchy of covariance descriptors on 3d joint locations. In *Twenty-third international joint conference on artificial intelligence*.

[16] Eun-Kyu Ji and Sang-Heon Lee. 2016. Effects of virtual reality training with modified constraint-induced movement therapy on upper extremity function in acute stage stroke: a preliminary study. *Journal of physical therapy science* 28, 11 (2016), 3168–3172.

[17] Jun H Kim, Daniel V Gunn, Eric Schuh, Bruce Phillips, Randy J Pagulayan, and Dennis Wixon. 2008. Tracking real-time user experience (TRUE) a comprehensive instrumentation solution for complex systems. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*. 443–452.

[18] Simon Kloiber, Volker Settgast, Christoph Schinko, Martin Weinzerl, Johannes Fritz, Tobias Schreck, and Reinhold Preiner. 2020. Immersive analysis of user motion in VR applications. *The Visual Computer* 36, 10 (2020), 1937–1949.

[19] Marcos Lage, Jorge Piazentin Ono, Daniel Cervone, Justin Chiang, Carlos Dietrich, and Claudio T Silva. 2016. Statcast dashboard: Exploration of spatiotemporal baseball data. *IEEE computer graphics and applications* 36, 5 (2016), 28–37.

[20] Injung Lee, Hyunchul Kim, and Byungjoo Lee. 2021. Automated playtesting with a cognitive model of sensorimotor coordination. In *Proceedings of the 29th ACM International Conference on Multimedia*. 4920–4929.

[21] Lars Lenssen and Erich Schubert. [n. d.]. Clustering by Direct Optimization of the Medoid Silhouette. Int. Conf. on Similarity Search and Applications, SISAP 2022. https://doi.org/10.1007/978-3-031-17849-8_15

[22] Luis Levy and Jeannie Novak. 2009. *Game development essentials: Game QA & testing*. Delmar Learning.

[23] Wanwan Li, Haikun Huang, Tomay Solomon, Behzad Esmaeili, and Lap-Fai Yu. 2022. Synthesizing Personalized Construction Safety Training Scenarios for VR Training. *IEEE Transactions on Visualization and Computer Graphics* 28, 5 (2022), 1993–2002. doi:10.1109/TVCG.2022.3150510

[24] Tica Lin, Alexandre Aouididi, Chen Zhu-Tian, Johanna Beyer, Hanspeter Pfister, and Jui-Hsien Wang. 2023. VIRD: immersive match video analysis for high-performance badminton coaching. *IEEE transactions on visualization and computer graphics* 30, 1 (2023), 458–468.

[25] Tica Lin, Zhutian Chen, Yalong Yang, Daniele Chiappalupi, Johanna Beyer, and Hanspeter Pfister. 2022. The quest for omnioculars: Embedded visualization for augmenting basketball game viewing experiences. *IEEE transactions on visualization and computer graphics* 29, 1 (2022), 962–971.

[26] Huimin Liu, Minsoo Choi, Liuchuan Yu, Alexandros Koilias, Lap-Fai Yu, and Christos Mousas. 2022. Synthesizing Shared Space Virtual Reality Fire Evacuation Training Drills. In *2022 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*.

[27] Weizhou Luo, Zhongyuan Yu, Rufat Rzayev, Marc Satkowski, Stefan Gumhold, Matthew McGinity, and Raimund Dachselt. 2023. Pearl: Physical environment based augmented reality lenses for in-situ human movement analysis. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–15.

[28] Pejman Mirza-Babaei, Naeem Moosajee, and Brandon Drenikow. 2016. Playtesting for indie studios. Proceedings of the 20th International Academic Mindtrek Conference, 366–374.

[29] Pejman Mirza-Babaei, Lennart E Nacke, John Gregory, Nick Collins, and Geraldine Fitzpatrick. 2013. How does it play better? exploring user testing and biometric storyboards in games user research. Proceedings of the SIGCHI conference on human factors in computing systems, 1499–1508.

[30] Pejman Mirza-Babaei, Günter Wallner, Graham McAllister, and Lennart E Nacke. 2014. Unified visualization of quantitative and qualitative playtesting data. In *CHI'14 Extended Abstracts on Human Factors in Computing Systems*. 1363–1368.

[31] Michael Nebeling, Maximilian Speicher, Xizi Wang, Shwetha Rajaram, Brian D Hall, Zijian Xie, Alexander RE Raistrick, Michelle Aebersold, Edward G Happ, Jiayin Wang, et al. 2020. MRAT: The mixed reality analytics toolkit. In *Proceedings of the 2020 CHI Conference on human factors in computing systems*. 1–12.

[32] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[33] Hannah Pileggi, Charles D Stolper, J Michael Boyle, and John T Stasko. 2012. Snapshot: Visualization to propel ice hockey analytics. *IEEE Transactions on Visualization and Computer Graphics* 18, 12 (2012), 2819–2828.

[34] Peter Pirolli and Stuart Card. 2005. The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. In *Proceedings of international conference on intelligence analysis*, Vol. 5. McLean, VA, USA, 2–4.

[35] Tom Polk, Dominik Jäckle, Johannes Häußler, and Jing Yang. 2019. CourtTime: Generating actionable insights into tennis matches using visual analytics. *IEEE Transactions on Visualization and Computer Graphics* 26, 1 (2019), 397–406.

[36] Tom Polk, Jing Yang, Yueqi Hu, and Ye Zhao. 2014. Tennivis: Visualization for tennis match analysis. *IEEE transactions on visualization and computer graphics* 20, 12 (2014), 2339–2348.

[37] Vahid Ranandeh and Pejman Mirza-Babaei. 2023. Beyond Equilibrium: Utilizing AI Agents in Video Game Economy Balancing. In *Companion Proceedings of the Annual Symposium on Computer-Human Interaction in Play*. 155–160.

[38] Chitta Ranjan, Samaneh Ebrahimi, and Kamran Paynabar. 2022. Sequence graph transform (SGT): a feature embedding function for sequence data mining. *Data Mining and Knowledge Discovery* 36, 2 (2022), 668–708.

[39] Patrick Reipschläger, Frederik Brudy, Raimund Dachselt, Justin Matejka, George Fitzmaurice, and Fraser Anderson. 2022. Avatar: An immersive analysis environment for human motion data combining interactive 3d avatars and trajectories. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. 1–15.

[40] Erich Schubert and Peter J Rousseeuw. 2019. Faster k-medoids clustering: improving the PAM, CLARA, and CLARANS algorithms. In *Similarity Search and Applications: 12th International Conference, SISAP 2019, Newark, NJ, USA, October 2–4, 2019, Proceedings 12*. Springer, 171–187.

[41] Erich Schubert and Peter J Rousseeuw. 2021. Fast and eager k-medoids clustering: O (k) runtime improvement of the PAM, CLARA, and CLARANS algorithms.

Information Systems 101 (2021), 101804.
[42] Charles P Schultz and Robert Denton Bryant. 2016. *Game testing: All in one.* Mercury Learning and Information.
[43] Ben Shneiderman. 2003. The eyes have it: A task by data type taxonomy for information visualizations. In *The craft of information visualization.* Elsevier, 364–371.
[44] Manuel Stein, Johannes Häußler, Dominik Jäckle, Halldór Janetzko, Tobias Schreck, and Daniel A Keim. 2015. Visual soccer analytics: Understanding the characteristics of collective team movement based on feature-driven analysis and abstraction. *ISPRS International Journal of Geo-Information* 4, 4 (2015), 2159–2184.
[45] Manuel Stein, Halldor Janetzko, Andreas Lamprecht, Thorsten Breitkreutz, Philipp Zimmermann, Bastian Goldlücke, Tobias Schreck, Gennady Andrienko, Michael Grossniklaus, and Daniel A Keim. 2017. Bring it to the pitch: Combining video and movement data to enhance team sport analysis. *IEEE transactions on visualization and computer graphics* 24, 1 (2017), 13–22.
[46] unity.com. 2006. https://unity.com/
[47] Günter Wallner and Simone Kriglstein. 2013. PLATO: Understanding Gameplay Data Through Visualization. In *CHI Games User Research Workshop.* 5.
[48] Günter Wallner, Marnix Van Wijland, Regina Bernhaupt, and Simone Kriglstein. 2021. What players want: information needs of players on post-game visualizations. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems.* 1–13.
[49] Jiachen Wang, Jiang Wu, Anqi Cao, Zheng Zhou, Hui Zhang, and Yingcai Wu. 2021. Tac-Miner: Visual tactic mining for multiple table tennis matches. *IEEE Transactions on Visualization and Computer Graphics* 27, 6 (2021), 2770–2782.
[50] Jiang Wu, Dongyu Liu, Ziyang Guo, Qingyang Xu, and Yingcai Wu. 2021. TacticFlow: Visual analytics of ever-changing tactics in racket sports. *IEEE Transactions on Visualization and Computer Graphics* 28, 1 (2021), 835–845.
[51] Yingcai Wu, Ji Lan, Xinhuan Shu, Chenyang Ji, Kejian Zhao, Jiachen Wang, and Hui Zhang. 2017. iTTVis: Interactive visualization of table tennis data. *IEEE transactions on visualization and computer graphics* 24, 1 (2017), 709–718.
[52] Yingcai Wu, Xiao Xie, Jiachen Wang, Dazhen Deng, Hongye Liang, Hui Zhang, Shoubin Cheng, and Wei Chen. 2018. Forvizor: Visualizing spatio-temporal team formations in soccer. *IEEE transactions on visualization and computer graphics* 25, 1 (2018), 65–75.
[53] Shuainan Ye, Chen Zhu-Tian, Xiangtong Chu, Yifan Wang, Siwei Fu, Lejun Shen, Kun Zhou, and Yingcai Wu. 2020. Shuttlespace: Exploring and analyzing movement trajectory in immersive visualization. *IEEE transactions on visualization and computer graphics* 27, 2 (2020), 860–869.
[54] Chong You, Chun-Guang Li, Daniel P Robinson, and René Vidal. 2016. Oracle based active set algorithm for scalable elastic net subspace clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition.* 3928–3937.
[55] Yongqi Zhang, Biao Xie, Haikun Huang, Elisa Ogawa, Tongjian You, and Lap-Fai Yu. 2019. Pose-Guided Level Design. In *Proceedings of the 37th Annual ACM Conference on Human Factors in Computing Systems.*
[56] Yongqi Zhang, Chuan Yan, Haikun Huang, Simon Su, and Lap-Fai Yu. 2024. Mixed-Integer Programming for Adaptive VR Workflow Training. In *International Conference on Human-Computer Interaction.* Springer, 325–344.

A Additional Demographic Details About Players for VR Games

We implemented our framework on two motion-based VR games (Reflex and Escape Room) and one VR training application. We recruited participants via emails sent across the university and selected individuals with prior VR gameplay experience.

For the Reflex game, we recruited 27 players (ages: 19-35, $M = 25.92$, $SD = 3.99$), with 17 male and 10 female participants, for a three-minute gameplay session. During each session, we logged player movement, eye-gaze data, and in-game performance metrics such as the number of collected coins and obstacle hits.

For the escape room game, we recruited 14 players (ages: 19-30, $M = 24.78$, $SD = 3.64$), including 10 male and 4 female participants, to play the game. Each participant completed the game within five minutes. During gameplay, we collected in-game metrics such as completion time, player trajectories, and eye gaze data.

In prior research on VR fire evacuation training, Liu et al. [26] collected data from 27 groups (54 participants; 34 males and 20 females). Participants were between 17 and 30 years old ($M = 19.96$, $SD = 2.88$). The researchers recorded several in-game measurements to analyze participants' collaborative behaviors, including completion time, completion time offset, trajectory length, distance between participants, extinguisher counts, and the number of commands issued.

Expert User Study To evaluate the effectiveness and usability of HieraVisVR, we conducted a user study with five expert game testers. To mimic this workflow, we designed a VR testing questionnaire (Table A1) to guide participants to evaluate a game.

Table A1: Tasks given to game testers for evaluating the HieraVisVR system.

Exploration Stage
1. Identify player(s) who play very differently from the rest of the group.
2. Identify player(s) who bump into obstacles frequently.
3. Identify player(s) who look around a lot.
4. Identify player(s) who are sitting or get stuck at the bar.
5. Identify any area in a chunk that has not been visited.
6. Identify any area that has been visited but shall not be (potentially a bug).
7. Identify any area with obstacles that has been visited.
Grouping (Poses) Stage
1. Focus on Chunks 1-4: Identify a group of players who excelled at the game. These players should have demonstrated exceptional performance by collecting many coins, avoiding obstacles, dodging bombs, and collecting boxes effectively.
2. Focus on Chunks 1-4: Identify a group of players who performed poorly in the game. These players frequently bumped into obstacles, struggled with game mechanics, and failed to achieve key objectives.
3. Focus on Chunks 1-4: Identify a group of players who consistently prioritized collecting boxes regardless of other game elements.
Grouping (Eye Gaze) Stage
1. Focus on chunk 4: What differences can be observed in gaze patterns between the groups in this chunk?
2. Focus on chunk 10: How do the gaze patterns vary between the groups in this chunk?
3. Comparison of chunk 4 and chunk 10: Are there any distinct or shared gaze patterns comparing these two chunks?
Explanation Stage
1. What does best performance refer to in your opinion? Can you form two groups with the best performance and the worst performance?
2. Select Player X from the best performance group. Select Player Y from the worst performance group. Can you tell why Player X performed better than Player Y?