Online-Journals.org

iJIM International Journal of Interactive Mobile Technologies

iJIM | eISSN: 1865-7923 | Vol. 19 No. 7 (2025) | OPEN ACCESS

https://doi.org/10.3991/ijim.v19i07.54975

PAPER

Design Principles and Practices for Mobile User Interfaces in High-Fidelity Virtual Reality Environments

Yuanyuan Shen(✉)

School of Information Engineering, Anhui Institute of International Business, Hefei, China

2010103@ahiib.edu.cn

ABSTRACT

With the continuous development of virtual reality (VR) technology, high-fidelity VR environments are increasingly applied in fields such as education, healthcare, and entertainment. As the primary mode of interaction between users and virtual environments, mobile user interfaces (MUI) play a crucial role, where the design's effectiveness directly impacts user immersion and operational efficiency. However, most current MUI designs rely on 2D interface design principles, lacking a tailored design framework for 3D VR environments. In particular, in high-fidelity VR environments, user interaction demands exhibit high dynamics and individuality, which presents challenges for traditional design methods that struggle with adaptability. To address this, this paper first proposes design principles for MUI in high-fidelity VR environments, aiming to provide theoretical guidance and a practical framework for interface design in this field. Secondly, the paper investigates an anchor-point matching method based on simultaneous localization and mapping (SLAM) to solve the dynamic matching problem between user position and interface elements in VR environments. The study shows that by integrating SLAM technology, the accuracy and smoothness of user interaction can be significantly improved. This study enriches the theoretical foundation of user interface design in VR and offers developers more efficient and flexible interaction approaches.

KEYWORDS

high-fidelity virtual reality (VR), mobile user interface (MUI) design, simultaneous localization and mapping (SLAM), anchor-point matching, user interaction

1 INTRODUCTION

With the rapid development of virtual reality (VR) technology, an increasing number of high-fidelity VR environments are being widely applied in fields such as education, healthcare, and gaming [1–3]. In these environments, user immersion and interaction experience have become important criteria for measuring system performance [4–6]. As the bridge between users and the virtual world, the MUI plays a crucial role. Designing an intuitive, efficient, and comfortable MUI not only

Shen, Y. (2025). Design Principles and Practices for Mobile User Interfaces in High-Fidelity Virtual Reality Environments. *International Journal of Interactive Mobile Technologies (iJIM)*, 19(7), pp. 44–57. https://doi.org/10.3991/ijim.v19i07.54975

Article submitted 2024-12-17. Revision uploaded 2025-02-13. Final acceptance 2025-02-18.

enhances the user experience but also increases the system's practicality and interactivity. How to effectively design the MUI in high-fidelity VR has become a major challenge in current VR research [7–10].

Research on MUI design in high-fidelity VR environments is of great significance. First, as VR technology matures, more users are beginning to use VR systems, and ensuring that they can easily and smoothly operate in the virtual world has become a key factor in enhancing the user experience of VR systems [11, 12]. By conducting in-depth research into the design principles of MUI in VR environments, practical design guidelines can be provided for VR device developers, and users can enjoy more natural and convenient interaction methods [13–16]. Second, as VR applications continue to expand, it is especially important to design an MUI that meets the needs of various application scenarios. How to provide consistent and efficient interaction experiences in different VR applications is a problem worth further exploration.

Currently, although some studies have explored user interface design in VR environments, existing methods still have certain limitations. Traditional interface design principles mainly focus on applications in 2D environments and lack adaptability to high-fidelity VR scenarios [17–19]. At the same time, most existing interaction modes are based on fixed control methods, making it difficult to meet the dynamic demands of users in motion for interface interactions [20–22]. In addition, as VR technology continues to evolve, the requirements for precise MUI positioning and real-time interaction are becoming increasingly high. Existing simultaneous localization and mapping (SLAM) technology in VR has not yet been fully optimized, and especially in high-dynamics, mobile user scenarios, achieving precise anchor-point matching remains a problem to be solved.

The primary research content of this paper includes two aspects. First, this paper explores the MUI design principles in high-fidelity VR environments, proposing a series of design guidelines that meet the application needs of VR. Through analyzing user needs in VR environments and combining the current technological developments, this study explores adaptive, intuitive, and user-friendly interface design solutions. Second, this paper investigates an anchor-point matching method based on SLAM, aimed at improving the positioning accuracy of user location and the real-time interaction of the interface in VR systems. This method optimizes SLAM algorithms and anchor-point matching technology to achieve more precise dynamic interaction, providing technical support for MUI in high-fidelity VR environments. The research in this paper not only fills the gap in existing studies but also provides new ideas and directions for future user interface design and technological applications in VR systems.

2 MUI DESIGN PRINCIPLES IN HIGH-FIDELITY VR ENVIRONMENTS

In high-fidelity VR environments, immersion is one of the core goals of user experience. To maintain a high level of immersion, the design of MUI must ensure that visual elements are consistent with other objects in the virtual environment. The design of interface elements needs to be integrated into the overall visual style of the virtual space to avoid abrupt visual disconnections. For example, the colors, textures, and lighting effects of interface elements must be coordinated with the visual environment of the virtual world to ensure that users do not feel discomfort or distraction due to mismatched visuals between the interface and the environment during interaction. Additionally, virtual interface elements should dynamically adjust according to the scene features of the virtual environment, avoiding fixed placement of the interface. They should naturally blend into the user's field of view to enhance the fluidity and naturalness of the interaction.

Spatial perception in VR environments is crucial for user interaction. In high-fidelity VR, users interact not only on a two-dimensional plane but also in three-dimensional space with interface elements. Therefore, MUI design must follow the principle of spatial consistency. Specifically, interface elements need to dynamically adapt according to the user's perspective, position, movement status, and other factors. For example, when the user changes position or turns their perspective in different directions, the interface elements should naturally adjust their position and orientation in line with the user's view, avoiding static interface elements. This ensures spatial consistency with the virtual environment. Furthermore, the actions of the user and the feedback from the interface elements should also maintain a natural correspondence in space. For instance, when the user turns their head or moves their hands, buttons, menus, and other elements in the interface should respond in real time and be displayed in the most comfortable position for the user, avoiding being obstructed or placed outside of the user's interactive range.

In high-fidelity VR environments, the principle of visual-spatial consistency requires that virtual interface elements maintain dynamic spatial consistency with the user's view and objects in the virtual environment. This means that interface elements must be closely linked to the specific position in the environment and the user's head or hand movements to ensure stability and naturalness. Anchor-point matching technology plays a key role in implementing this design principle. Anchor-point matching assigns specific virtual space coordinates to interface elements, ensuring that these interface elements can automatically adjust their spatial position according to user actions and changes in perspective. For example, when the user walks, turns their head, or changes their view in the virtual environment, anchor-point matching technology can real-time position the interface elements at fixed spatial locations or in relation to specific objects, ensuring that these interface elements remain within the user's line of sight, avoiding drifting or misalignment. This precise spatial matching not only enhances the visual consistency between the interface and the virtual environment but also ensures operational stability during user interaction.

Another important value of anchor-point matching technology in improving the stability of MUI is that it effectively solves the flexibility and accuracy issues of interface elements in user-space interaction in VR. Traditional user interface design often relies on fixed two-dimensional layouts, but in VR environments, interface elements need to dynamically adjust according to the user's spatial position, movement trajectory, and changes in perspective. Through anchor-point matching, interface elements are "fixed" at the spatial anchor points required by the user, ensuring that regardless of how the user turns or moves, the interface remains in the appropriate position for interaction. For example, when using gesture controls, buttons, menus, and other interface elements adjust in real time according to the user's hand position and do not shift due to the user's movement or change in posture.

3 ANCHOR-POINT MATCHING METHOD BASED ON SLAM IMPLEMENTATION

The application of the anchor-point matching method based on SLAM in high-fidelity VR environments is crucial for ensuring that MUI elements maintain consistency with the physical space in the virtual environment through precise spatial localization and map construction, thereby stabilizing the user interaction experience. In the front-end of SLAM, the three-dimensional point cloud data obtained through scanning provides initial geometric information of the VR environment, which can be used as the initial positioning basis for the anchor points of the interface elements. Through the front-end scan matching technology, the user's position is linked to specific three-dimensional coordinates in the environment, enabling the interface elements to dynamically adjust their position in virtual space based on the user's actual location, perspective, and dynamic changes in real time. This process ensures the visual-spatial consistency between the interface elements and the physical space in the virtual environment, meaning that no matter how the user moves or changes their perspective, the interface elements will consistently align with the actual objects in the environment and the user's interaction needs, avoiding drifting or misalignment of the interface elements in virtual space. On the back-end, SLAM's graph model is used to further optimize these initial anchors point positions through probabilistic constraints and sparse matrix optimization, adjusting the user's movement trajectory and map construction for global consistency. During the loop closure phase, the SLAM system analyzes the front-end user interaction data and back-end subgraph information to identify and correct map deviations caused by accumulated errors. This optimization process ensures the accuracy of anchor-point matching and prevents inconsistencies between the interface elements and the virtual environment caused by the accumulation of errors during long-term interaction.

3.1 Scan matching inserted into subgraphs

In high-fidelity VR environments, the stability of the MUI relies on accurate pose estimation and dynamic map updating, with the scan matching inserted into subgraphs being one of the key technologies for achieving this goal. Based on the SLAM method, scan matching is used to align the point cloud data from laser scans with an existing map, providing accurate spatial information for the user interface. In a VR environment, the user's pose $\gamma = (a, b, \varphi)^T$ determines their position and orientation in virtual space. Through scan matching technology, each laser scan's generated point set $G = \{g_u\}$ is aligned with the previous submap. Each new laser scan data is matched with previous data, and the position of the scan frame is adjusted through the coordinate transformation S_γ, ensuring that the point cloud data is accurately inserted into the current subgraph, forming a complete and more stable virtual map. The coordinate transformation formula is as follows:

$$T_u(\gamma) = S_t o = \begin{pmatrix} COS\varphi & -SIN\varphi \\ SIN\varphi & COS\varphi \end{pmatrix} \begin{pmatrix} g_a \\ g_b \end{pmatrix} + \begin{pmatrix} a \\ b \end{pmatrix} \qquad (1)$$

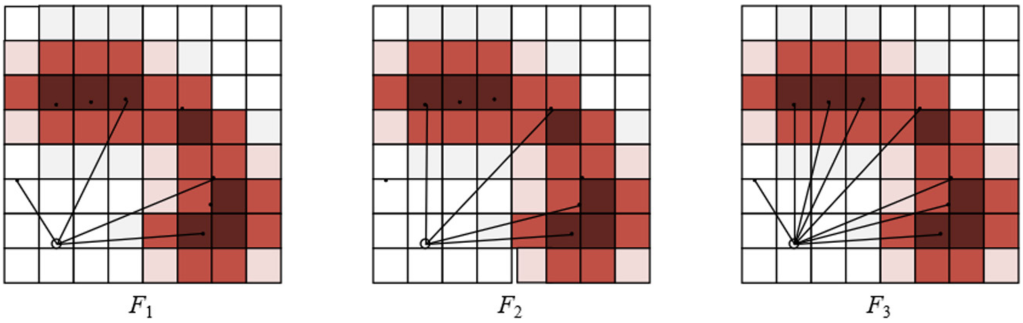

Fig. 1. Example of subgraph scale increase

The introduction of the subgraph concept, through the divide-and-conquer method, divides the global map into multiple smaller subgraphs, effectively reducing the accumulation of errors and thus improving the precision of scan matching.

Figure 1 shows an example of the increase in subgraph scale. When the user moves within the virtual environment, the system calculates the transformation relationship between each pair of consecutive scan frames, matching the current frame's laser scan data to previous frames, thereby generating a new subgraph. As the user continues to move, new scan data is inserted into the corresponding subgraph, with each subgraph representing a small area of the virtual environment. As the subgraphs continuously expand and update, the system can precisely build a gradually optimized global map, ensuring spatial consistency between the interface elements in the VR environment and the real environment. Particularly in high-fidelity VR, the continuous growth of subgraphs allows the virtual environment to respond more precisely to the user. Whenever the user changes their position or perspective, the interface elements are updated based on the real-time scan data, thereby avoiding drift or misalignment of interface elements.

3.2 Gaussian-Newton optimization to reduce errors

In the anchor-point matching method based on SLAM, as the user moves and scan data is continually added, the accumulation of errors between the gradually formed subgraphs occurs. To ensure the stability of the MUI in high-fidelity VR environments, it is necessary to correct the poses of these subgraphs through an effective optimization method. The Gaussian-Newton optimization is a commonly used algorithm for solving such nonlinear least-squares problems. Its core objective is to eliminate map deviations caused by accumulated errors by minimizing the pose errors, thus ensuring the stability and accuracy of the user interface.

During the Gaussian-Newton optimization process, an initial pose γ needs to be given first, and the new pose change $\Delta\gamma$ is estimated using scan data or previous subgraph information. The core of this step is to define and minimize the observation error caused by pose errors between subgraphs. In VR environments, such errors typically manifest as deviations between interface elements and the actual positions of objects in the virtual environment, which may lead to misalignment or uncoordinated experiences during user interaction. Assuming the cubic spline smoothing function is represented by L, the optimized pose is:

$$\gamma^* = \underset{\gamma}{\operatorname{argmin}} \sum_{u=1}^{v} \left(1 - L\left(T_u(\gamma)\right)\right)^2 \tag{2}$$

First, given the initial pose γ, define and estimate $\Delta\gamma$, and minimize the observation error, we have:

$$\sum_{u=1}^{v} \left(1 - L\left(T_u(\gamma + \Delta\gamma)\right)\right)^2 \to 0 \tag{3}$$

To eliminate these errors, a first-order Taylor expansion $L(T_u(\gamma+\Delta\gamma))$ is used to linearize the errors; that is, assuming that the pose change $\Delta\gamma$ is small, the error change can be approximated as a linear relationship between the pose error and the pose change. This step transforms the complex nonlinear problem into a simpler linear problem, making it easier to solve.

$$\sum_{u=1}^{v} \left(1 - L\left(T_u(\gamma)\right) - \nabla L\left(T_u(\gamma)\right) \frac{\partial T_u(\gamma)}{\partial \gamma} \Delta\gamma\right)^2 \to 0 \tag{4}$$

Next, the Gaussian-Newton method iteratively optimizes the poses of each subgraph, gradually correcting the errors. Specifically, the Gaussian-Newton method calculates the Jacobian matrix of the errors and uses it to update the pose Δε, thereby gradually reducing the error. In each iteration, the algorithm continuously adjusts the pose of the subgraph until the error is minimized. The advantage of the Gaussian-Newton method is its ability to gradually converge through multiple iterations, effectively reducing the misalignment of interface elements in the virtual environment caused by the accumulation of errors. Suppose the Hessian matrix is represented by G, and the formula for solving $\Delta\gamma$to minimize it using the Gaussian-Newton method is:

$$\Delta\gamma = G^{-1}\sum_{u=1}^{v}\left[\nabla L\left(T_u(\gamma)\right)\frac{\partial T_u(\gamma)}{\partial\gamma}\right]^S\left[1-L\left(T_u(\gamma)\right)\right] \tag{5}$$

The Hessian matrix formula is:

$$G = \left[\nabla L\left(T_u(\gamma)\right)\frac{\partial T_u(\gamma)}{\partial\gamma}\right]^S\left[\nabla L\left(T_u(\gamma)\right)\frac{\partial T_u(\gamma)}{\partial\gamma}\right] \tag{6}$$

Suppose the gradient of the subgraph is approximated by ∇L, then by combining with Eq. (1), we have:

$$\frac{\partial T_u(\gamma)}{\partial\gamma} = \begin{pmatrix} 1 & 0 & -SIN(\varphi)g_a & -COS(\varphi)g_b \\ 0 & 1 & COS(\varphi)g_a & -SIN(\varphi)g_b \end{pmatrix} \tag{7}$$

3.3 Pose loop closure and map correction

In high-fidelity VR environments, the stability of the user interface directly depends on accurate pose estimation and real-time dynamic map updates. As the user moves through the virtual space, the accumulation of errors may distort the map, affecting the position of interface elements and the smoothness of interactions. To solve this issue, pose loop closure and map correction are critical steps to ensure the accuracy of the map and the stability of the interface in the VR environment.

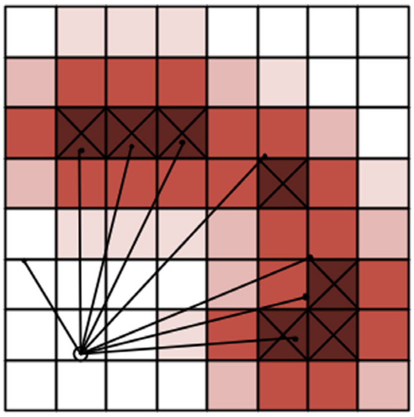

Fig. 2. Map drift detection

Pose loop closure refers to the system's ability to detect the similarity between the current user position and previously visited positions when the user returns to an earlier location in the VR environment. By matching scan frames, the system identifies the loop closure. Specifically, pose loop closure is based on the previously constructed subgraphs and scan data. It compares the estimated pose of the new scan frame with the poses in the existing subgraphs to find a loop closure. In this way, the system can effectively identify and correct map drift caused by pose estimation errors, as shown in Figure 2. In practice, loop closure detection quantifies the error using relative pose constraints γ_{uk} and covariance matrix $\sum_{uk}$. Suppose the number of accumulated subgraphs is denoted as l, and the number of correctly matched scans is v. The poses of the subgraphs and scan poses in the global coordinate system can be represented as $R^l = \left\{\gamma_u^l\right\}_{u=1,\dots,l}$ and $R^t = \left\{\gamma_k^t\right\}_{k=1,\dots,l}$, then the following equation holds:

$$\underset{R^l,R^t}{\operatorname{argmin}} \frac{1}{2} \sum_{uk} \vartheta\left(R^2\left(\gamma_u^l, \gamma_k^t; \sum\nolimits_{uk}, \gamma_{uk}\right)\right) \tag{8}$$

These constraints are added during the optimization process, helping the system determine whether a new scan frame matches an existing subgraph or scan frame. The loop closure is confirmed by comparing the features in the scan data. Suppose the Huber loss function is represented by ϑ, the residual of this constraint is calculated as:

$$R^2\left(\gamma_u^l, \gamma_k^t; \sum\nolimits_{uk}, \gamma_{uk}\right) = r\left(\gamma_u^l, \gamma_k^t; \gamma_{uk}\right)^S \sum\nolimits_{uk}^{-1} r\left(\gamma_u^l, \gamma_k^t; \gamma_{uk}\right) \tag{9}$$

$$r\left(\gamma_u^l, \gamma_k^t, \gamma_{uk}\right) = \gamma_{uk} - \begin{pmatrix} E_{\gamma_u^l}^{-1}\left(s_{\gamma_u^l} - s_{\gamma_k^t}\right) \\ \gamma_{u;\varphi}^l - \gamma_{k;\varphi}^t \end{pmatrix} \tag{10}$$

When the subgraph construction is completed, it is further incorporated into the loop closure detection. In high-fidelity VR, loop closure detection not only relies on the similarity of the user's pose but also uses nonlinear optimization methods to ensure precise matching. At regular intervals, the *Ceres* nonlinear optimization library is used for global optimization, further reducing the impact of errors. In this way, pose loop closure detection in the virtual environment not only ensures the stability of the user interface but also effectively corrects map distortion caused by user path curvature or error accumulation.

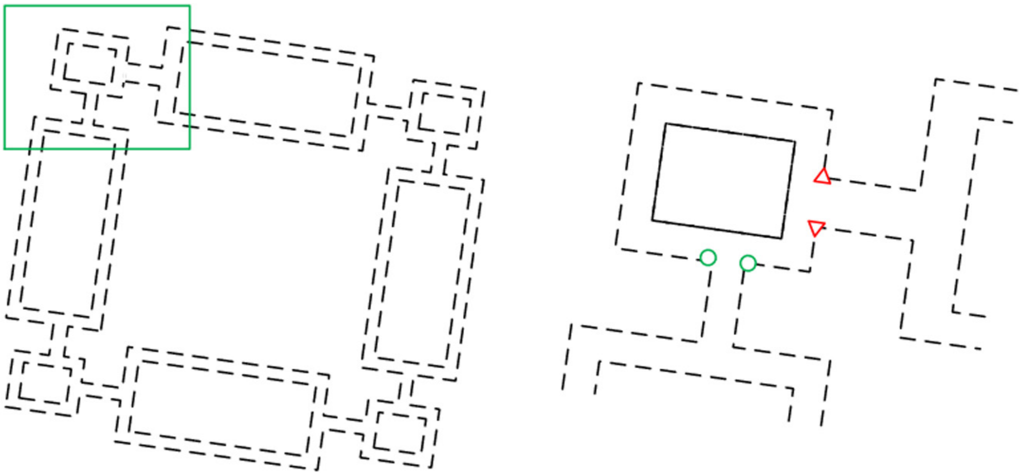

Fig. 3. Example of global and local maps after loop closure optimization

The core goal of pose loop closure detection is to correct errors by re-estimating the user's position in the map, thus performing map correction. When the system detects that the user has returned to a previously visited area, it compares the new scan frame with previous ones. By matching the two different time poses as identical, the system eliminates the effect of errors. For example, if the user moves from starting point A along a path to reach position C, due to accumulated errors, the path from A to C on the map appears distorted. However, through loop closure detection, the system finds that the features near point C match those at point A, thus considering A and C to be the same point. By matching these two positions, the system can correct the entire map's deviation. Loop closure correction not only eliminates path errors but also performs global optimization on the map, ensuring that the user interface elements in the virtual environment remain consistent with their real-world positions, avoiding misalignment or inconsistency caused by error accumulation.

In the map correction process, nonlinear least-squares optimization algorithms are used to optimize the poses of all subgraphs and scan data in a unified manner, ensuring the accuracy of the global map. By treating each subgraph's pose as an optimization variable, the system calculates new constraints and performs optimization at each loop closure, gradually eliminating the map distortion caused by error accumulation. Figure 3 shows an example of the global and local maps after loop closure optimization. Figure 4 provides a comparison of the loop closure optimized mobile path.

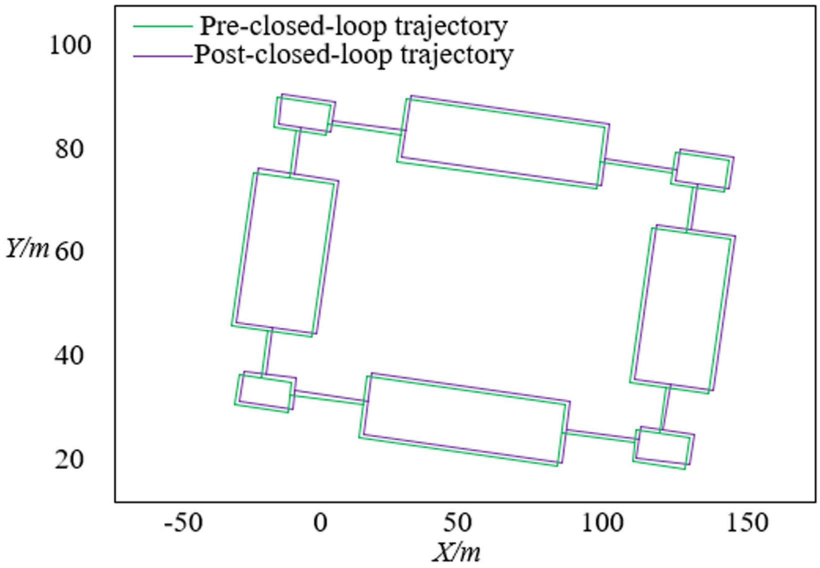

Fig. 4. Comparison of loop closure optimized mobile path

4 EXPERIMENTAL RESULTS AND ANALYSIS

According to the experimental data shown in Figure 6, the absolute trajectory error of the user's movement path along three dimensions (x, y, z) exhibits certain variations. Specifically, in the x-axis direction, the error fluctuates significantly over time, with a noticeable increase at 30 seconds and 40 seconds, where the maximum error reaches –0.13 m. At 15 seconds, the error is 0, after which it gradually changes to –0.04 m. Overall, the error shows fluctuations over longer time periods. The error in the y-axis direction remains relatively stable, with only slight increases at 30 seconds and 35 seconds. Most of the errors stay within a small range (–0.1 m to 0.13 m), and the fluctuation is minimal, especially between 25 seconds and 35 seconds, where the error remains close to 0. The z-axis error also shows some fluctuation but is relatively consistent, with an error range of approximately 0 to 0.12 m. Except for the period between 20 and 30 seconds, where the error remains relatively stable, there is noticeable fluctuation during other times. From the analysis of the experimental data, it can be seen that the user's position error in the VR environment does not show an obvious growth trend, and the error fluctuations are relatively stable. Particularly, the errors along the y-axis and z-axis are smaller, indicating that the positioning accuracy of the VR system is ideal in these two directions. Figure 5 displays an example of the matched path using the method described in this paper.

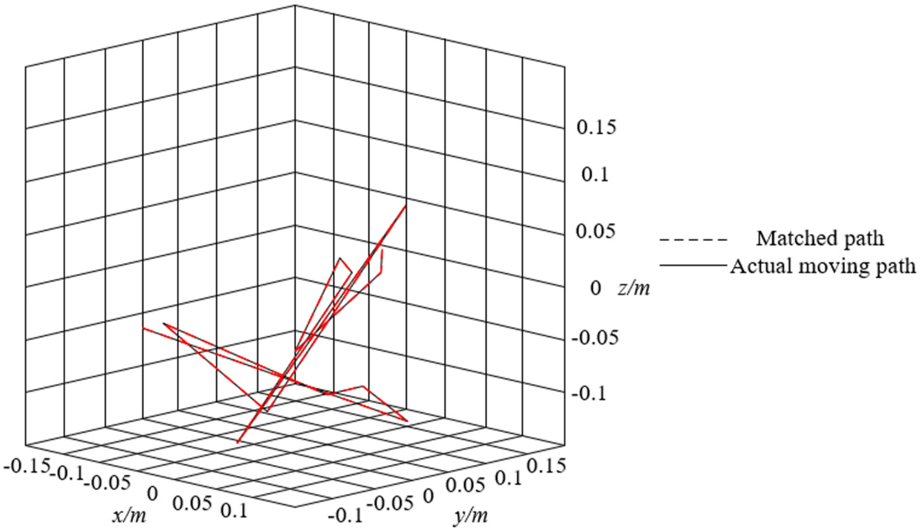

Fig. 5. Example of matched path using the method described in this paper

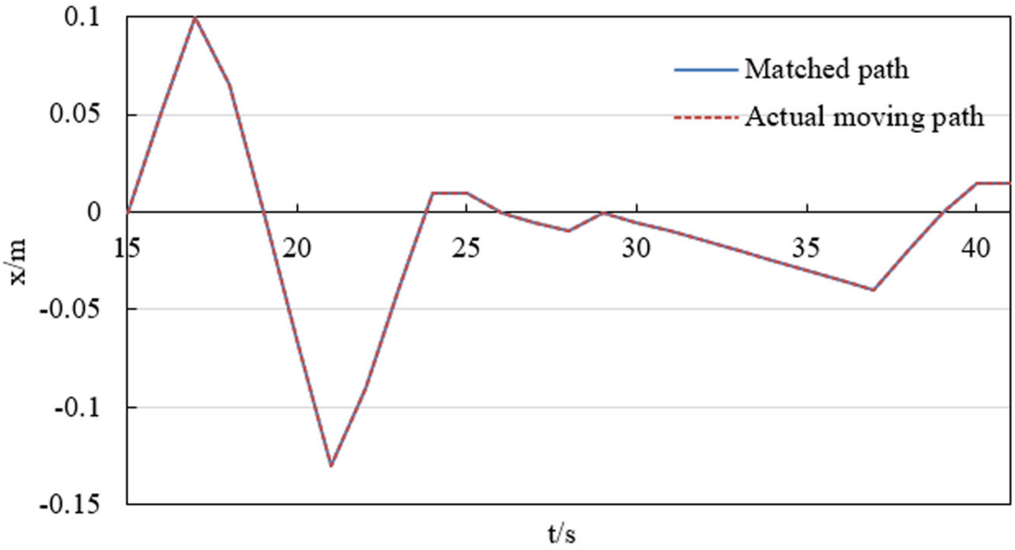

Fig. 6. *(Continued)*

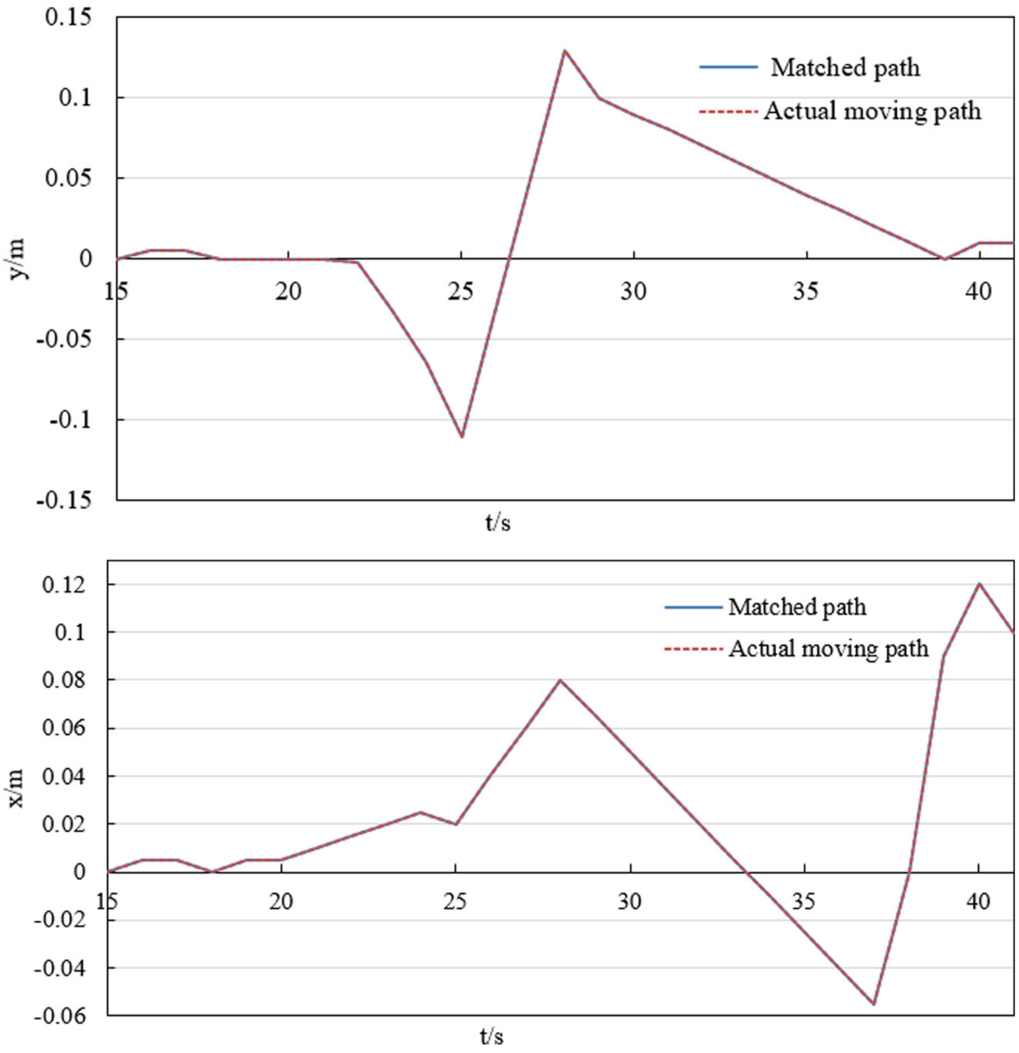

Fig. 6. Absolute trajectory error of the user's movement path

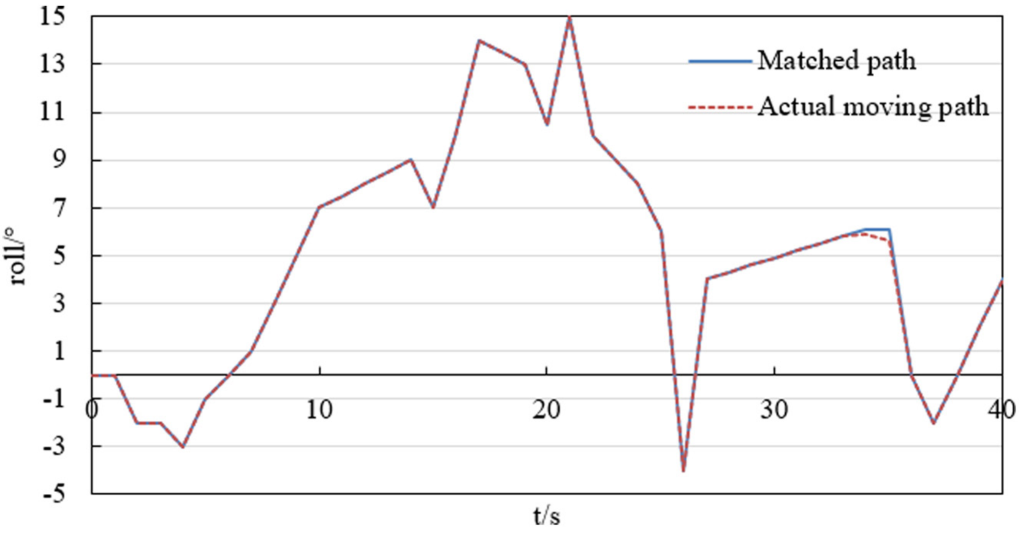

Fig. 7. *(Continued)*

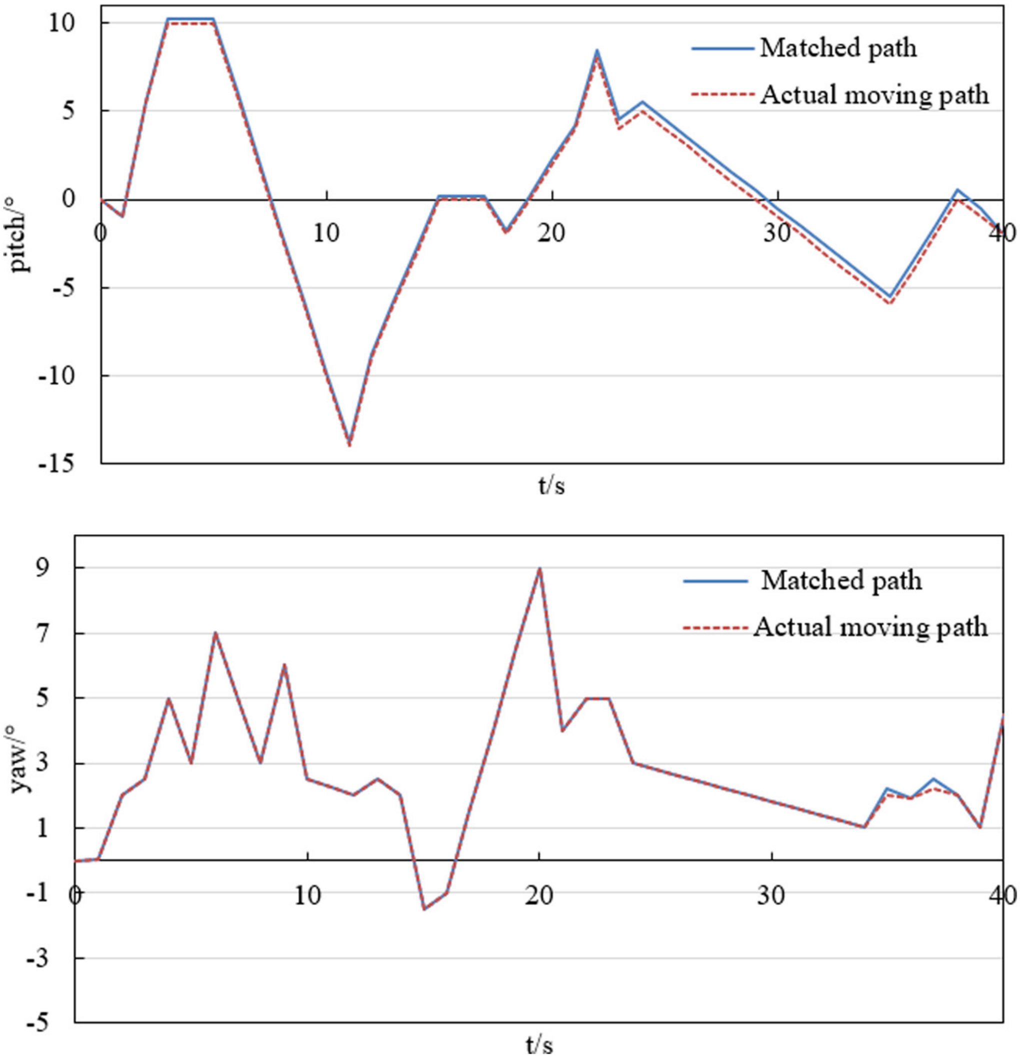

Fig. 7. User pose absolute trajectory error

According to the experimental data given in Figure 7, the absolute trajectory errors of the user's pose in the three rotation angles (roll, pitch, and yaw) exhibit certain fluctuations over time. For the roll axis, the error variation is relatively small, with the overall fluctuation range between –4 m and 15 m. In the early stage (0 to 20 seconds), the error remains relatively stable, with the maximum error occurring at 30 seconds and 40 seconds, being –3 m and 4 m, respectively. The error on the pitch axis is more significant, with a fluctuation range from –13.8 m to 10.2 m. Particularly between 20 seconds and 30 seconds, the error changes more drastically, and between 30 seconds and 40 seconds, the fluctuation tends to converge, becoming stable. The fluctuation of the yaw axis error is relatively small, with a change range of –1.5 m to 9 m, but there are periods (such as at 10 seconds and 30 seconds) where the error is somewhat abrupt, with the maximum error being 9 m. Overall, the error fluctuations on all three axes show instability, especially in the error changes on the pitch axis, where there are large fluctuations. Combined with the experimental data, it can be seen that in the user's pose estimation process in the VR system, although the error changes are relatively stable in the roll and yaw directions, the error fluctuation in the pitch axis is more pronounced, indicating poorer positioning accuracy in this direction. Specifically, the larger fluctuations in the pitch direction may be related to factors such as sensor accuracy in the VR system and the positioning errors of the SLAM algorithm in the vertical direction.

Table 1. Matching efficiency and accuracy for different matching schemes

Method	Matching Time	Accuracy /%
EKF SLAM	17s/iteration	82.6
Fast SLAM	6.7s/iteration	85.4
The Proposed Method	4.1s/iteration	92.3

According to the experimental data of different matching schemes in Table 1, significant differences were observed among the three methods in terms of matching time and accuracy. The EKF SLAM method had a matching time of 17 seconds per iteration, with an accuracy of 82.6%. The Fast SLAM method had a shorter matching time of 6.7 seconds per iteration, but the accuracy improved to 85.4%. The method proposed in this paper, however, demonstrated the best balance between matching time and accuracy, with a matching time of 4.1 seconds per iteration and the highest accuracy of 92.3%. These results indicate that the method proposed in this paper not only significantly outperforms EKF SLAM and Fast SLAM in terms of time efficiency but also exhibits a notable advantage in accuracy.

5 CONCLUSION

This paper focuses on the design of MUI in high-fidelity VR environments and anchor point matching methods based on SLAM, aiming to improve the accuracy and real-time performance of user interaction in VR systems. First, in terms of interface design, this paper analyzes user requirements in VR environments and, combined with current technological developments, proposes a series of interface design guidelines that meet the needs of VR applications, emphasizing adaptable and intuitive design solutions. Through these design principles, this paper provides theoretical support and practical guidance for user interfaces in future VR systems. Second, in response to the challenges of user position tracking in VR, this paper investigates anchor point matching technology based on the SLAM algorithm, optimizing the system's positioning accuracy and interaction response speed. Experimental data verification shows that the SLAM optimization method proposed in this paper exhibits significant advantages in improving the system's real-time performance and accuracy, enabling more precise user position tracking, and greatly enhancing the smoothness of VR interactions.

In summary, this study provides innovative technical support for MUI design and positioning in high-fidelity VR systems, achieving significant results in improving system accuracy and real-time performance. The optimized SLAM algorithm and anchor point matching technology significantly reduce matching time and improve accuracy, offering more stable and efficient solutions for dynamic interaction in VR environments. The main value of this study lies in its effective response to core issues in VR technology, particularly in application scenarios that require high positioning accuracy and real-time performance, which has important practical significance. However, there are certain limitations in this study. For instance, the study mainly focuses on the optimization of traditional SLAM algorithms and does not explore the potential integration of emerging deep learning methods with SLAM in depth. Moreover, this paper does not address the impact of more complex VR environments (such as multi-user interaction in large-scale scenarios) on system performance. Therefore, future research could explore the combination of deep learning and SLAM to further enhance the system's adaptability and intelligence. Additionally, the application scenarios of the research could be further expanded to test its performance in more complex and diverse VR environments, further improving and optimizing the overall performance of VR systems.

6 REFERENCES

[1] R. Alsalameen, L. Almazaydeh, B. Alqudah, and K. Elleithy, "Information technology students' perceptions toward using virtual reality technology for educational purposes," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 17, no. 7, pp. 148–166, 2023. https://doi.org/10.3991/ijim.v17i07.37211

[2] H. D. Sharma, Y. Misra, S. Kumar, B. M. Rao, and B. Ch, "Expanding an education-based collision detection system created on virtual reality and augmented reality," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 17, no. 17, pp. 108–120, 2023. https://doi.org/10.3991/ijim.v17i17.42831

[3] H. A. Al-Jundi and E. Y. Tanbour, "A framework for fidelity evaluation of immersive virtual reality systems," *Virtual Reality*, vol. 26, pp. 1103–1122, 2022. https://doi.org/10.1007/s10055-021-00618-y

[4] T. Mizuho, T. Narumi, and H. Kuzuoka, "Effects of the visual fidelity of virtual environments on presence, context-dependent forgetting, and source-monitoring error," *IEEE Trans. Vis. Comput. Graph.*, vol. 29, no. 5, pp. 2607–2614, 2023. https://doi.org/10.1109/TVCG.2023.3247063

[5] A. Sudiarno, D. A. Amanullah, and R. A. Akbar, "The measurement of evacuation effectiveness regarding dynamic evacuation routing system (DERS) in high-rise building using virtual reality simulation," *International Journal of Safety and Security Engineering*, vol. 12, no. 1, pp. 115–122, 2022. https://doi.org/10.18280/ijsse.120114

[6] A. F. Macnamara, K. Bird, A. Rigby, T. Sathyapalan, and D. Hepburn, "High-fidelity simulation and virtual reality: An evaluation of medical students' experiences," *BMJ Simulation & Technology Enhanced Learning*, vol. 7, no. 6, pp. 528–535, 2021. https://doi.org/10.1136/bmjstel-2020-000625

[7] H. Dammak, M. Riahi, and F. Moussa, "Adapting mobile user interfaces based on users' application usage behaviours," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 44, no. 1, pp. 48–66, 2023. https://doi.org/10.1504/IJAHUC.2023.133453

[8] C. J. Lin and S. H. Ho, "The development of a mobile user interface ability evaluation system for the elderly," *Applied Ergonomics*, vol. 89, p. 103215, 2020. https://doi.org/10.1016/j.apergo.2020.103215

[9] Z. Qu and J. Sun, "Adaptive mobile web interface: User readiness in context," *International Journal of Mobile Communications*, vol. 10, no. 2, pp. 132–149, 2012. https://doi.org/10.1504/IJMC.2012.045670

[10] P. H. J. Chong, P. L. So, P. Shum, X. J. Li, and D. Goyal, "Design and implementation of user interface for mobile devices," *IEEE Transactions on Consumer Electronics*, vol. 50, no. 4, pp. 1156–1161, 2004. https://doi.org/10.1109/TCE.2004.1362513

[11] S. Guan, "Computer-aided design methods in virtual reality product development," *Advances in Multimedia*, vol. 2022, no. 1, pp. 1–11, 2022. https://doi.org/10.1155/2022/9489650

[12] P. Kuna, A. Hašková, and Ľ. Borza, "Creation of virtual reality for education purposes," *Sustainability*, vol. 15, no. 9, p. 7153, 2023. https://doi.org/10.3390/su15097153

[13] K. B. Park and J. Y. Lee, "New design and comparative analysis of smartwatch metaphor-based hand gestures for 3D navigation in mobile virtual reality," *Multimedia Tools and Applications*, vol. 78, pp. 6211–6231, 2019. https://doi.org/10.1007/s11042-018-6403-9

[14] M. Kim, J. Lee, C. Jeon, and J. Kim, "A study on interaction of gaze pointer-based user interface in mobile virtual reality environment," *Symmetry*, vol. 9, no. 9, p. 189, 2017. https://doi.org/10.3390/sym9090189

[15] V. Biener *et al*., "PoVRPoint: Authoring presentations in mobile virtual reality," *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, no. 5, pp. 2069–2079, 2022. https://doi.org/10.1109/TVCG.2022.3150474

[16] F. Liarokapis and E. Conradi, "User interfaces for mobile navigation," *Library Hi Tech*, vol. 25, no. 3, pp. 352–365, 2007. https://doi.org/10.1108/07378830710820934

[17] E. Chang and T. S. Dillon, "Knowledge-based and template-based perceptual design of a user interface," *Engineering Intelligent Systems for Electrical Engineering and Communications*, vol. 7, no. 2, pp. 97–108, 1999.

[18] Z. Ji, W. H. Huang, and X. Zhang, "Design and implementation of a game interface interaction on smartphone," *Journal of Intelligent & Fuzzy Systems*, vol. 34, no. 2, pp. 923–931, 2018. https://doi.org/10.3233/JIFS-169386

[19] Z. Wei and J. Nie, "Research on interface design based on user's mental model driven by interactive genetic algorithm," *International Journal of Bio-Inspired Computation*, vol. 17, no. 1, pp. 42–51, 2021. https://doi.org/10.1504/IJBIC.2021.113364

[20] Z. Guan, Y. Li, H. Wang, and G. Dai, "A comparison of free interaction modes for mobile layout system," in *Handheld and Ubiquitous Computing, HUC 2000,* in Lecture Notes in Computer Science, P. Thomas and H. W. Gellersen, Eds., Springer, Berlin, Heidelberg, vol. 1927, 2000, pp. 63–71. https://doi.org/10.1007/3-540-39959-3_5

[21] J. Sang, T. Mei, Y. Q. Xu, C. Zhao, C. Xu, and S. Li, "Interaction design for mobile visual search," *IEEE Transactions on Multimedia*, vol. 15, no. 7, pp. 1665–1676, 2013. https://doi.org/10.1109/TMM.2013.2268052

[22] C. Lim and L. Jin-Hwa, "The effects of task modality and type on Korean EFL learners' interactions," *Journal of Asia TEFL*, vol. 12, no. 2, pp. 87–123, 2015.

7 AUTHOR

Yuanyuan Shen, holds a master's degree and is a dual-qualified teacher, recognized as a promising newcomer on her school's teaching stage. Her research focuses on animation and virtual reality, and she primarily teaches courses in "3D Modeling," "3D Animation," and "Film and Television Special Effects Production." She has successfully guided students to win first-place awards in the "Virtual Reality Design and Production" category at the National Vocational College Skills Competition for three consecutive terms. She has also won first-place awards at the Anhui Province Higher Vocational College Teaching Ability Competition for two consecutive years. Additionally, she has repeatedly led students to achieve first and second-place awards in the Anhui Province College Students' Original Animation Contest. Shen has received one special-grade award, two first-grade awards, and one second-grade award for provincial teaching achievements. She has led and participated in multiple provincial and school-level quality engineering projects (E-mail: 2010103@ahiib.edu.cn; ORCID: https://orcid.org/0009-0006-3535-7538).