Multimedia Tools and Applications
https://doi.org/10.1007/s11042-024-18252-6

Trajectory-based alignment for optical see-through HMD calibration

Lingling Chen[1,2] · Shaohua Zhao[1,2] · Wei Chen[3,4] · Zhongchen Shi[3,4] · Liang Xie[3,4] · Ye Yan[3,4] · Erwei Yin[3,4]

Received: 10 May 2023 / Revised: 7 January 2024 / Accepted: 9 January 2024

Abstract

In order to align the virtual and real content precisely through augmented reality devices, especially in optical see-through head-mounted displays (OST-HMD), it is necessary to calibrate the device before using it. However, most existing methods estimated the parameters via 3D-2D correspondences based on the 2D alignment, which is cumbersome, time-consuming, theoretically complex, and results in insufficient robustness. To alleviate this issue, in this paper, we propose an efficient and simple calibration method based on the principle of directly calculating the projection transformation between virtual space and the real world via 3D-3D alignment. The proposed method merely needs to record the motion trajectory of the cube-marker in the real and virtual world, and then calculate the transformation matrix between the virtual space and the real world by aligning the two trajectories in the observed view. There are two advantages associated with the proposed method. First, the operation is simple. Theoretically, the user only needs to perform four alignment operations for calibration without changing the rotation variation. Second, the trajectory can be easily distributed throughout the entire observation view, resulting in more robust calibration results. To validate the effectiveness of the proposed method, we conducted extensive experiments on our self-built optical see-through head-mounted display (OST-HMD) device. The experimental results show that the proposed method can achieve better calibration results than other calibration methods.

✉ Wei Chen
wei.chen.ai@outlook.com

Lingling Chen
chenling@hebut.edu.cn

Shaohua Zhao
zshua998@163.com

[1] School of Artificial Intelligence and Data Science, Hebei University of Technology, 300130 Tianjin, People's Republic of China

[2] Engineering Research Center of Intelligent Rehabilitation Device and Detection Technology Ministry of Education, 300130 Tianjin, People's Republic of China

[3] Defense Innovation Institute, Academy of Military Sciences (AMS), 100071 Beijing, People's Republic of China

[4] Tianjin Artificial Intelligence Innovation Center (TAIIC), 300130 Tianjin, People's Republic of China

Published online: 03 February 2024

Keywords Augmented reality · Optical see-through head-mounted display · Calibration · Trajectory alignment

1 Introduction

In recent years, the application of optical see-through head-mounted displays has significantly increased with the advancements in augmented reality (AR) technology [1–4]. OST-HMDs are crucial AR devices that overlay virtual objects onto the user's real-world field of view through transparent displays, providing users with immersive augmented reality experiences. However, to achieve seamless integration of virtual and real-world elements, precise rendering of virtual objects in the real-world environment is of utmost importance. Currently, most OST-HMD devices utilize internal tracking systems that accurately determine the transformation between the external environment and their tracking coordinate system. However, the alignment between virtual objects and real-world references within the user's field of view remains a challenge.

To seamlessly blend virtual and physical objects, establishing the transformation relationship between the rendering and tracking systems of these devices becomes necessary. Addressing this requirement presents significant challenges [5, 6] that need to be tackled. The importance of precisely calibrating optical see-through head-mounted display devices extends beyond augmented reality gaming and entertainment applications; it applies to practical applications across various domains. In the medical field [7, 8], accurate calibration of OST-HMD allows doctors to align virtual medical images with the actual patient's anatomy, thereby enhancing surgical precision and safety. In the education sector [9, 10], precise calibration of OST-HMDs can provide students with more realistic virtual learning environments, improving their learning experiences and comprehension. In the industrial domain, by precisely calibrating OST-HMDs, engineers can design and simulate product prototypes in the virtual world, enhancing work efficiency and accuracy. Therefore, understanding and overcoming the challenges involved in precise calibration of optical see-through head-mounted display devices is of paramount importance.

Some traditional methods [11–18] combined the optical lens of the OST-HMD device with the user's eye and modeled it as an off-axis pinhole camera. Its imaging plane corresponded to the translucent virtual screen, and its projection center corresponded to the center of the user's eye. They connected 3D-2D point pairs by aligning the world reference point with the image points on the virtual screen of the OST-HMD device and then calculated the internal and external parameters of the established camera model. These methods, especially when estimating all projection parameters simultaneously, are cumbersome, time-consuming, and highly dependent on operators' skills. On the other hand, using some independent points to calibrate OST-HMD devices focuses on local information rather than the overall field of view, resulting in a weak guarantee of consistent calibration results across the entire field of view. Even though these points are distributed uniformly in the field of view, they remain scattered individuals. If there exist some outliers, they could have a tremendous impact on the calibration results.

In recent years, researchers have proposed a new method for calibration using 3D markers to simplify the calibration process and improve calibration accuracy. The methods proposed in [19, 20] treated the interior of the OST-HMD device as a black box took data from the tracking system as input, and visualized virtual objects in the user's eyes as output. By recording the respective poses of the markers and virtual models when aligned and then directly calculating the projection matrix from real-world 3D space to virtual 3D space, the calculation process was dramatically simplified. This type of 3D-3D approach solves the problem of the need for numerous repeated alignment operations in traditional 3D-2D methods. Additionally, compared to aligning only a single 3D point, aligning the 6DoF pose of 3D markers is more intuitive and reduces the presence of outlier data. However, these methods still have some limitations. On the one hand, using 3D markers to align operations appears to reduce the number of alignment operations required. However, in practice, it is necessary to simultaneously consider the 6DoF of the markers, which makes alignment complex. On the other hand, similar to 3D point data, 3D markers still occupy only a small part of the field of view, making it difficult to ensure consistent calibration across the entire field of view. In addition, these methods are also limited to some extent by the influence of abnormal marker poses, which reduces its robustness.

To address the aforementioned issues, we propose a new calibration method for OST-HMD devices. The proposed method adopts a holistic approach and no longer focuses on the corner information or pose of the marker surface at a single moment. Instead, it records the motion trajectory of the cube marker in the real and virtual worlds. The transformation matrix between the virtual space and the real world is obtained by aligning their motion trajectories. Then the projection matrix required for anchoring the virtual content to the correct position is determined. As the motion trajectory is a holistic entity, it assures the consistency of the calibration across the entire field of view.

Specifically, we use a 10cm × 10cm × 10cm cube as a marker (Fig. 1) and generate a corresponding virtual cube model for alignment operations. The user only needs to perform four alignments, and between every two alignments, the marker only produces a 1DoF change of the position, significantly reducing the difficulty of alignment. A new evaluation experiment was designed to verify the calibration results. The experimental results demonstrate that our proposed method outperforms traditional 3D-3D calibration methods in terms of accuracy, robustness, and data requirements.

The main contributions of our work are as follows:

- A new calibration method for OST-HMD devices is proposed, which obtains the transformation matrix between the virtual space and the real world by aligning the motion

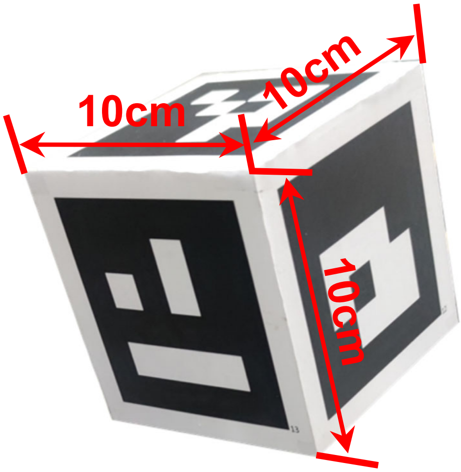


Fig. 1 The cube marker: plastic material. A 10cm × 10cm × 10cm cubic structure fabricated through 3D printing. The surface is adorned with five distinct ArUco codes, while the bottom features a hollow design, facilitating mounting on the calibration devices

trajectories of the markers and virtual models in the tracking system and the rendering system, respectively.
- We design a new calibration procedure for the trajectory-based alignment to simplify the operation of manual alignment.
- A new evaluation method is designed to evaluate calibration accuracy objectively.

2 Related work

Research on achieving the correct calibration of OST-HMD has been conducted over many years. We here provide a more concise overview of the different calibration methods.

The single point active alignment method (SPAAM) [13] is a widely applied method for display calibration due to its simplicity and accuracy. In SPAAM, users collect many 3D-2D point pairs, and then a mapping from the 3D point cloud to its 2D screen coordinates is calculated using Direct Linear Transform (DLT) [21, 22]. However, manual data collection by users inevitably leads to human error. When the level of human error is too high, calibration failure may occur. Repeatedly collecting a large amount of data is necessary to ensure accuracy, which is not user-friendly. Some methods [14, 23–25] have been proposed to improve user interaction and robustness based on SPAAM. Fuhrmann et al. [26] proposed determining the parameters of the virtual viewing frustum by collecting 8 3D-2D point correspondences per eye that define the viewing frustum corner points. Grubert et al. [27] proposed simultaneously collecting multiple 3D-2D point correspondences in a Multiple Point Active Alignment scheme. In this scheme, the user aligns a grid of 9 3D points on a calibration board about 150 cm. While the calibration procedure significantly speeds up the data collection phase, it produces larger calibration errors. Zhang et al. [28, 29] proposed a dynamic SPAAM method that considers eye orientation to optimize the projection mode to adapt the shift of the eye center due to eye orientation. Unlike modeling the mapping from 3D point sets to the 2D screen coordinates as a projection, the DRC method [30] takes the physical model of the optics into account. These methods only considered calibration on a single eye. Stereo-SPAAM [31] simultaneously calibrated both eyes with a stereoscopic OST-HMD. Simultaneous calibration of a stereoscopic OST-HMD is better conditioned by adding the physical constraints of two eyes, such as interpupillary distance (IPD). In contrast to conventional manual calibration techniques, Microsoft HoloLens employs a user-centered calibration method [20]. This method eliminates the need for additional external markers and instead uses finger to align with the screen cursor, enabling manual calibration similar to SPAAM. This approach liberates users from relying on environment-dependent target constraints. By incorporating finger tracking using the Leap Motion controller, it can be easily integrated into most OST-HMD systems, alleviating the burden on both users and application developers.

These 3D-2D calibration methods necessitate a significant amount of 3D point data to ensure their accuracy, resulting in complex and laborious operational procedures and a substantial workload. Typically, this type of method requires 6 alignments for calibration, but in practice, it often demands more than 20 alignments to achieve the desired level of accuracy. In addition, aligning 3D point data with planar symbols cannot guarantee alignment quality. Abnormal 3D data can have a substantial impact on the calibration results, leading to a less robust system. These methods provide a detailed breakdown of the overall tracking and rendering process of the “physical environment-tracking camera-human eye imaging” in

OST-HMD devices and solve for the parameters of each component. These methods model the human eye and the lenses of the OST-HMD device as a camera to represent the rendering system. They calculate the intrinsic parameters of the camera model based on factors such as lens resolution and field of view, which enables the determination of the specific projection positions of virtual models onto the lens screen in the rendering system. By collecting a large amount of 3D-2D point correspondences, the transformation relationship between the virtual camera model and the tracking camera is determined. Different methods may also involve selecting different virtual camera models, contributing to the complexity of their underlying principles.

To address these issues, Azimi et al. [19] proposed a 3D-3D method for OST-HMD calibration. The cube marker was positioned by an optical tracker, and the user was instructed to manually move the cube marker for alignment to complete the calibration. Sun et al. [32] utilized a series of retroreflective spheres as markers for calibration purposes. These methods treated the interior of OST-HMD as a black box and directly calculated the transformation matrix between virtual space and the real world (3D-3D). On this basis, Xue Hu et al. [33] further improved this method to be user-centered. They embedded an RGB-D camera into the OST-HMD device to capture images of the user's palm and generate a point cloud of the palm's edges. Using this point cloud data, they generated virtual markers of the palm shape on the screen. The researchers employed a method called rotation-constrained iterative closest point (rcICP) for point cloud registration to optimize the calibration parameters related to the viewpoint. These methods no longer require establishing a virtual camera model, greatly simplifying the calculation process, but there are still problems, such as difficult alignment and insufficient robustness.

The non-interactive calibration method based on eye capture is also a potential solution [34–37]. Their method measured the eye center online and automatically generated a projection matrix. They used the same pinhole camera model as SPAAM. However, there is little related research work, especially on eye and display models, and further development is needed.

In line with this, a new 3D-3D calibration method based on cube motion trajectory is proposed in this paper. The proposed method can accurately calculate the posture compensation matrix between the virtual space and the real world and correct the misalignment between the real object and its virtual counterpart in the user's eyes.

3 Proposed method

In continuity with the insights gained from the preceding section on related work, Section 3.1 lays the groundwork by elucidating the fundamental principles that underpin the 3D-3D calibration method. By establishing a solid theoretical foundation, we pave the way for a nuanced comprehension of the intricacies involved in our proposed methodology. Building upon this theoretical framework, Section 3.2 introduces our innovative approach. Our method strategically employs the cube trajectory to expedite the calibration process, ensuring a harmonious blend of speed, accuracy, and robustness. This trajectory, a central component of our methodology, serves as a key enabler for overcoming challenges identified in the existing literature. Throughout this section, we delve into the specifics of our approach, providing a detailed exposition of its principles and rationale.

3.1 Basic principle of 3D-3D calibration method

In this 3D-3D calibration method, the head display's interior was treated as a black box, and detailed differentiation between various parameters of the virtual rendering camera was no longer required. The 3D visualization model projected by the virtual rendering camera is used as the output, and the estimated marker pose by the tracking camera is used as the input. A comprehensive projection matrix T is used to represent the complex transformation process in the black box. In fact, all the parameters of the human-eye camera model established using the traditional 3D-2D calibration method are integrated into the parameter matrix T.

Assuming that there is a 3D point O_1 in the real world and its corresponding rendered virtual point O_2, the transformation relationship between them can be represented as follows:

$$O_2 = T O_1. \tag{1}$$

Two pieces of information can be easily acquired: the pose of the cube marker in the coordinate system of the OST-HMD tracking camera (Fig. 2(a)); and the pose of the virtual cube model in the coordinate system of the rendering camera (Fig. 2(b)). Through the active alignment operation in Fig. 2(c), a good alignment between the cube marker and the virtual model can be achieved in 6DoF.

To clarify the principle, we define the transformation representation between two coordinates. Consider the coordinate systems A and B. Let ${}^{A}_{B}R$ represent the rotation matrix from A to B, and ${}^{A}_{B}t$ represent the translation vector from A to B. The transformation from A to B can be expressed as:

$$B = {}^{A}_{B}RA + {}^{A}_{B}t. \tag{2}$$

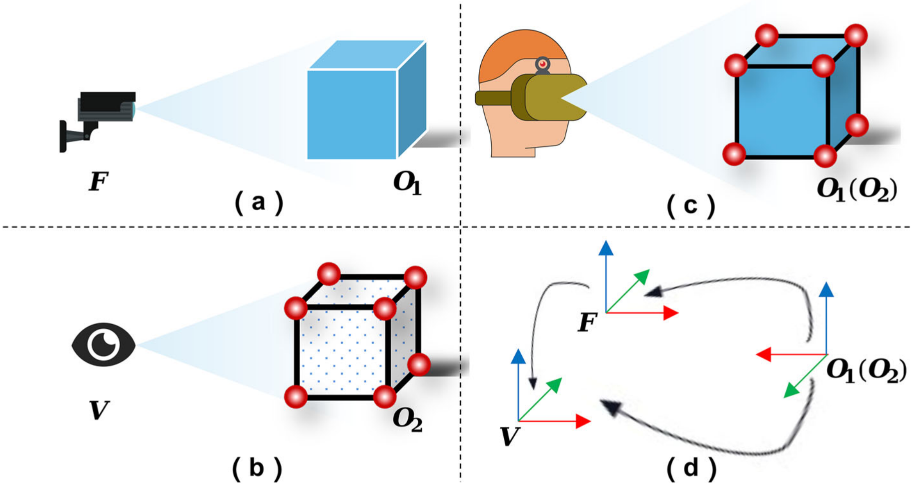


Fig. 2 Principle of the traditional 3D-3D calibration method. (a) Estimation of the pose of the cube marker O_1 by tracking camera F. (b) Rendering of virtual model O_2 within the field of view of camera V. (c) Manual alignment of cube marker O_1 with virtual model O_2. (d) Relative positional relationships among tracking camera F, rendering camera V, cube marker O_1, and virtual model O_2 in the aligned state

Figure 2(d) illustrates the relative positions of various coordinate systems in the calibration system, while their interrelationships can be described as follows:

$$\begin{cases} F = {}^{O_{1i}}_{F}R O_{1i} + {}^{O_{1i}}_{F}t \\ V = {}^{O_{2i}}_{V}R O_{2i} + {}^{O_{2i}}_{V}t \quad i = 1, \ldots, n, \\ O_{1i} = O_{2i} \end{cases} \tag{3}$$

where F stands for the coordinate system of the tracking camera, while V represents that of the virtual rendering camera. $O_{1i}(O_{2i})$ refers to the object coordinate system of the cube marker (virtual cube model) during the i_{th} alignment, and n denotes the total number of alignment operations.

The parameters we need to calibrate for OST-HMD are the transformation relationship between the real world and the virtual projection space. The real-world coordinate system can be defined using the tracking camera of OST-HMD, while the virtual projection space can be described using the rendering camera of OST-HMD. Thus, the parameter that needs to be calibrated is a transformation matrix, denoted as ${}^{F}_{V}T$, which represents the transformation from F to V:

$${}^{F}_{V}T = \begin{bmatrix} {}^{F}_{V}R & {}^{F}_{V}t \\ 0 & 1 \end{bmatrix}, \tag{4}$$

$$V = {}^{F}_{V}RF + {}^{F}_{V}t. \tag{5}$$

By combining (5) with (3), we obtain:

$$\begin{cases} {}^{F}_{V}R_i = {}^{O_i}_{V}R \, {}^{O_i}_{F}R^{-1} \\ {}^{F}_{V}t_i = {}^{O_i}_{V}t - {}^{O_i}_{V}R \, {}^{O_i}_{F}R^{-1} \, {}^{O_i}_{F}t \end{cases} \tag{6}$$

Calculate the average of multiple alignments as the final calibration result:

$$\begin{cases} {}^{F}_{V}R = AVE\left(\left[{}^{F}_{V}R_1, {}^{F}_{V}R_2, \ldots, {}^{F}_{V}R_n\right]\right) \\ {}^{F}_{V}t = AVE\left(\left[{}^{F}_{V}t_1, {}^{F}_{V}t_2, \ldots, {}^{F}_{V}t_n\right]\right) \end{cases} \tag{7}$$

From our perspective, aligning all six degrees of freedom of the cube simultaneously can still present a challenge, particularly in cases where manual operation is involved. Furthermore, the thickness of the virtual model framework itself can cause occlusion, resulting in slight differences in the pose of markers under the same alignment effect (particularly noticeable in depth). Minor distortions exist at the edges of the optical lenses in the head-mounted display, causing slight deformations in the projection of virtual models at the edge positions, preventing complete alignment. The pose estimation algorithm using surface QR code tracking cubes also has inherent random errors. These errors can significantly impact the calibration results by introducing individual outliers.

3.2 Calibration with cube trajectory

We introduce a novel 3D-3D calibration method that addresses the limitations of the method described in Section 3.1.

The proposed method only requires the user to perform four alignment operations to complete the basic calibration of the OST-HMD device. The difference lies in the fact that, unlike traditional methods, each subsequent alignment in our proposed method only moves the cube marker along one axis by a fixed step while maintaining the rotation angle of the cube marker unchanged, except for the initial alignment operation that accounts for the six degrees of freedom of the cube. This means that in the subsequent alignment operations, only one degree of freedom of the marker needs to be considered, making the alignment operation easy. We further reduce the information that needs to be collected by only focusing on the position information of the cube marker during each alignment. In addition, the method of using single alignment data for calibration and then jointly optimizing multiple calibration results is no longer adopted. Instead, the overall motion trajectory formed by multiple alignment data is directly utilized, making it easier and more thorough to weaken the impact of individual abnormal data on calibration results and improve the robustness of the calibration method. Furthermore, in designing the motion trajectory, we try to cover the entire visual field range, using the integrity of the motion trajectory to ensure the consistency of calibration results throughout the entire visual field. Finally, by aligning the motion trajectories of cube markers in both the real world and virtual rendering space to calculate their relative poses, we represent the transformation relationship between the real world and virtual space. Our method offers simpler operations and computations.

In the actual calibration process, to further improve accuracy, the number of alignment operations will be increased to seven (The traditional 3D-3D calibration method also performs seven alignment operations to facilitate the comparison of calibration results), including an initial position, moving the cube along each axis twice, and fitting the resulting points into three intersecting lines. The overall motion trajectory looks like a three-dimensional coordinate system, which will be referred to as the trajectory coordinate system hereafter. Figure 3(a) is a schematic diagram of the alignment result, displaying the distribution shape of the motion trajectory of the designed cube marker during the calibration process. The blue cube represents the real cube marker, while the black frame and red corner points constitute a virtual model that shares the same motion trajectory as the real object. Figure 3(b) depicts

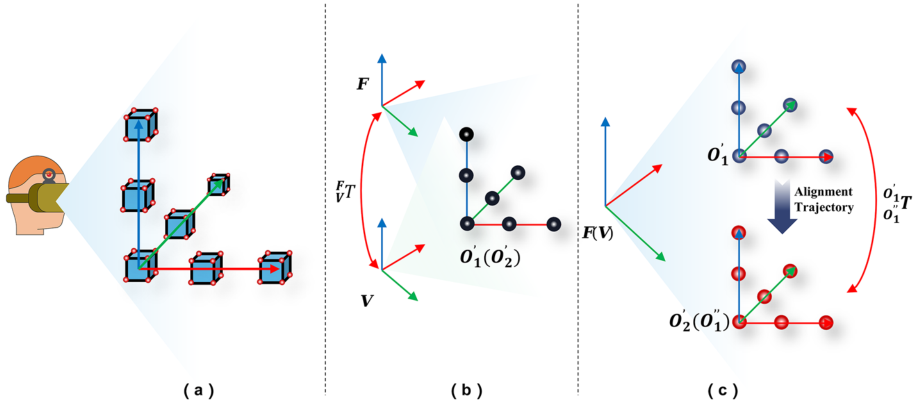


Fig. 3 Principle of our calibration method. (a) Distribution of the motion trajectory of the cube marker during the data acquisition process. The blue cube represents the real cube marker, while the black framework and red corner points constitute a virtual model sharing the same motion trajectory as the real cube marker. (b) Relative positional relationships among the trajectory coordinates of the cube marker $O_1^{'}$, virtual model trajectory coordinate system $O_2^{'}$, tracking camera F, and rendering camera V. (c) Assuming the coincidence of tracking camera F and rendering camera V, the two trajectories separate. Realignment of the trajectories allows for measuring the offset generated during the alignment process, representing the deviation between the tracking and rendering cameras

the relative position between the trajectory coordinate system $O_1^{'}$ of the cube marker and the tracking camera F, as well as the relative positioning between the trajectory coordinate system $O_2^{'}$ of the virtual cube model and the rendering camera V. To simplify the data collection process to only include positional information during alignment while disregarding rotation, the cube is represented by a black sphere in the illustration.

During the data collection process, we start by projecting the virtual model onto a selected position, then we move the cube marker to align it. We collect the position information of the cube marker and the virtual model when they are perfectly aligned. However, it is quite challenging to calculate the transformation relationship between the tracking camera and the rendering camera solely based on each point position in the trajectory. This is especially true when determining the rotational transformation between them. Therefore, we assume that there is no deviation between the real world and the virtual projection space, meaning the coordinate systems of the tracking camera and the rendering camera are perfectly aligned. As shown in Fig. 3(c), we place the previously collected position information of the marker and its corresponding virtual model in the same coordinate system, which was obtained in the F and V coordinate systems separately. As a result, $O_1^{'}$ and $O_2^{'}$ no longer coincide. Then, we adjust the movement trajectory of the cube marker to align it with the motion trajectory of the virtual model, bringing $O_1^{'}$ and $O_2^{'}$ back into coincidence at position $O_1^{''}$.

Throughout the alignment process, the transformation matrix ${}^{O_1^{'}}_{O_1^{''}}T$ generated by the movement of the cube marker's trajectory coordinate system can be considered equivalent to the transformation relationship ${}^{F}_{V}T$ from the tracking camera to the rendering camera. As shown in (8) and (9):

$$O_1^{''} = {}^{O_1^{'}}_{O_1^{''}}T\,O_1^{'}, \tag{8}$$

$${}^{F}_{V}T = {}^{O_1^{'}}_{O_1^{''}}T, \tag{9}$$

where $O_1^{'}$ is the coordinate system of the cube marker's trajectory before movement, and $O_1^{''}$ is the coordinate system of the cube marker's trajectory after movement.

The estimation of the cube's 3D positional information at the alignment moment is achieved by attaching two-dimensional markers (ArUco) [38] on the surfaces of the cube. The specific procedure is depicted in Fig. 4: Initially, the regions of the two-dimensional markers are detected. Subsequently, the two-dimensional pixel coordinates of the four corners of each detected marker are recorded, and the corresponding ID number of each two-dimensional marker is determined by consulting the marker dictionary. A coordinate system for the cube object is defined with the centroid of the cube as the origin. Based on the ID number of the two-dimensional marker, the face of the cube on which the marker is located and the three-dimensional position of the four corners in the cube coordinate system are determined. In the experiment, a cube with a side length of 10cm is used, and the marker has a side length of 8cm. Taking marker 10 as an example, it is positioned on the cube's surface along the positive z-axis direction, with corner coordinates of $z = 0.05m$ $(x = \pm 0.04m, x = \pm 0.04m)$. Finally, the pose information of the cube marker is determined by employing the Perspective-n-Point (PnP) algorithm [39] in conjunction with the three-dimensional to two-dimensional point correspondence.

A series of corresponding 3D position information is collected for the cubic marker and the virtual model, forming their respective trajectories. The trajectories are then aligned using the Iterative Closest Point (ICP) registration method. The ICP registration algorithm iteratively optimizes the transformation parameters, such as translation and rotation, between the cubic

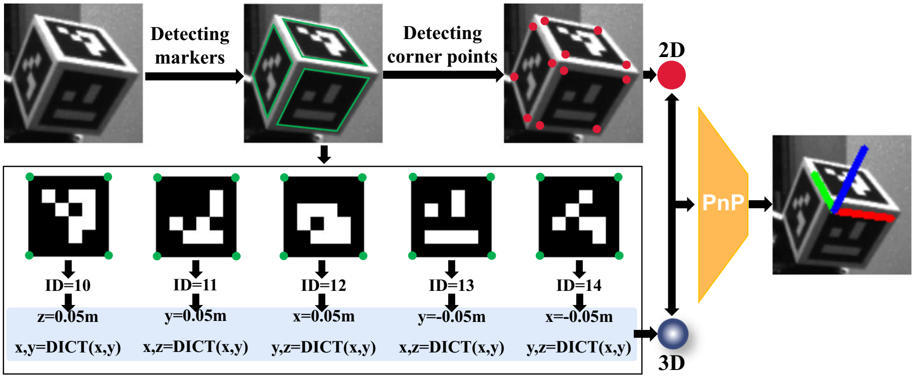


Fig. 4 Principle of cube marker localization. Firstly, the detection of ArUco code regions is performed. Subsequently, the 2D pixel coordinates of the four corners of each detected ArUco code are recorded, and the corresponding ID number for each ArUco code is determined by referencing the appropriate dictionary. The object coordinate system of the cube marker is defined with the centroid of the cube as the origin. Based on the ID number, the faces of the cube to which each ArUco code belongs and the 3D positions of the four corners in the cube coordinate system are determined. Finally, the PnP algorithm is employed, utilizing 3D-2D point correspondences, to determine the pose information of the cube marker

marker and the virtual model to minimize the distance between corresponding points. This ensures the alignment of the trajectories and produces an accurate registration result.

Compared to Section 3.1, the calibration method proposed in this section boasts significant advantages, primarily in three aspects: Firstly, the proposed method utilizes a cube motion trajectory to calibrate OST-HMD devices, mitigating the impact of individual anomalies in the cube pose information on the calibration accuracy and enhancing the system's robustness. Secondly, the movement trajectory closely connects the position information of the cube markers at each node, constructing an overall structure that results in a consistent calibration effect of the different areas in visual fields. Finally, it simplifies the collection of data information, thereby reducing the complexity of alignment operations, preventing repetitive and intricate manual operations that introduce errors, and ultimately enhancing calibration efficiency.

4 Implementation

To provide a comprehensive description of the proposed calibration method, we will introduce the hardware devices used in the calibration process and outline the specific calibration workflow in this section.

4.1 Hardware and software

Our calibration method applies to OST-HMD devices with different rendering principles. As illustrated in Fig. 5, we utilized a self-made OST-HMD in our experiments, which is equipped with a tracking system comprising two fish-eye cameras located at the front of a plastic housing for external object positioning. The display system includes two translucent optical lenses integrated under the plastic housing to render the virtual model projection.

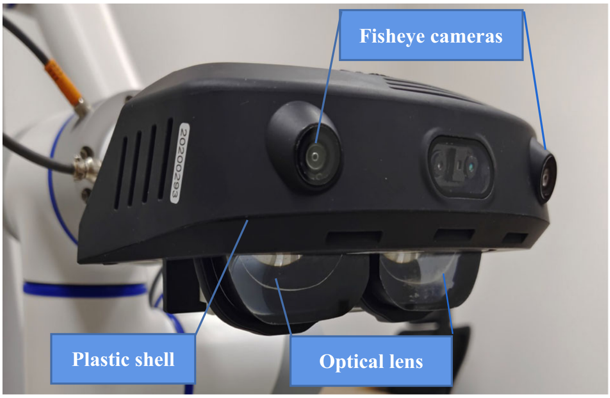


Fig. 5 Self-made OST-HMD. Equipped with a tracking system consisting of two fisheye cameras, positioned at the front of the plastic housing for external object localization. The display system comprises two semi-transparent optical lenses integrated beneath the plastic casing, facilitating the projection of virtual model images

In the experiment, a 10cm × 10cm × 10cm cube was employed as a marker, with distinct two-dimensional codes affixed to each of its six surfaces, to facilitate accurate pose estimation by the OST-HMD tracking cameras. As shown in Fig. 6, a virtual 3D model identical in shape and dimensions to the cube marker was generated for use during alignment procedures. To prevent any false alignment resulting from symmetry, we differentiated each face of the virtual cube model using different colors and established a one-to-one correspondence between those faces and the two-dimensional codes on the marker's surface.

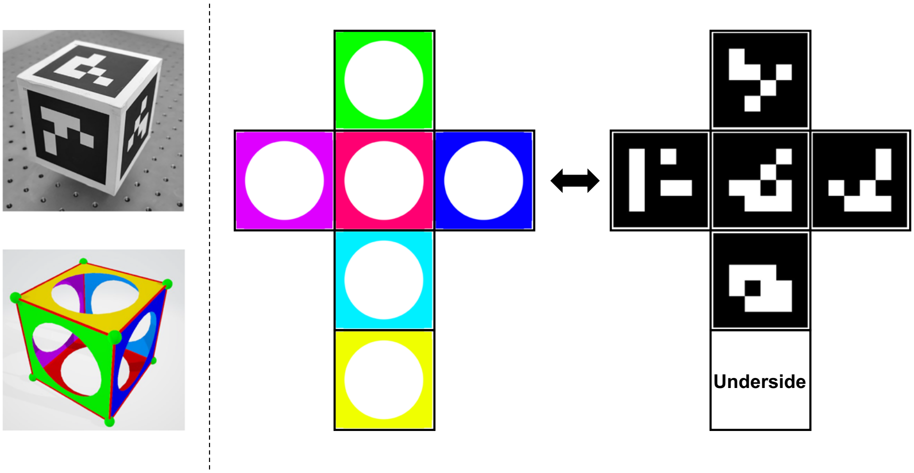


Fig. 6 Cube marker and virtual model. The cube model showcased in detail in Fig. 1 occupies the top left corner. At the bottom left, a virtual 3D model identical in shape and size to the cube marker is generated for use in the manual alignment process. Different colors are employed to differentiate each face of the virtual cube model. A one-to-one correspondence is established between these colors and the ArUco codes on the surface of the Cube marker, as shown on the right side of the image

Figure 7 shows the overall experimental environment. The left half of the setup utilizes a 3D printing jig to affix the OST-HMD device to the mechanical arm. Additionally, a RealSense D435i camera replaces the need for a human eye to record the alignment effect and is henceforth referred to as the observation camera. In the right half of the setup, a cube marker is securely mounted on the rotary translation table. The rotating translation platform moves the marker during alignment operations, thus ensuring smooth and precise movement during the alignment process.

During the implementation process, we primarily utilized the OpenCV-4.6.0 and Eigen-3.3.7 libraries for marker tracking, matrix operations, and numerical optimization. Additionally, we employed the Open3D-0.13.0 library to handle tasks such as trajectory generation, reading 3D point data, visualization, and alignment during the trajectory alignment process.

Incorporating external devices, such as a robotic arm, to assist in the calibration process of OST-HMD can enhance the ease and accuracy of alignment operations, while also aiding in the efficient recording of experimental data. The proposed calibration method is also suitable for the absence of these devices: operators holding markers can also successfully calibrate OST-HMD devices.

4.2 Operation process

The entire calibration work is divided into two main steps: (1) Calibrating the tracking camera's intrinsic parameters to provide a more accurate tracking effect. (2) Estimating the relative pose of the tracking camera and the virtual rendering camera to project virtual objects accurately.

4.2.1 Tracking camera calibration

Tracking cameras are calibrated using traditional calibration techniques that require acquiring images of the various poses of a checkerboard calibration board. This paper used the camera calibration toolbox in Matlab.

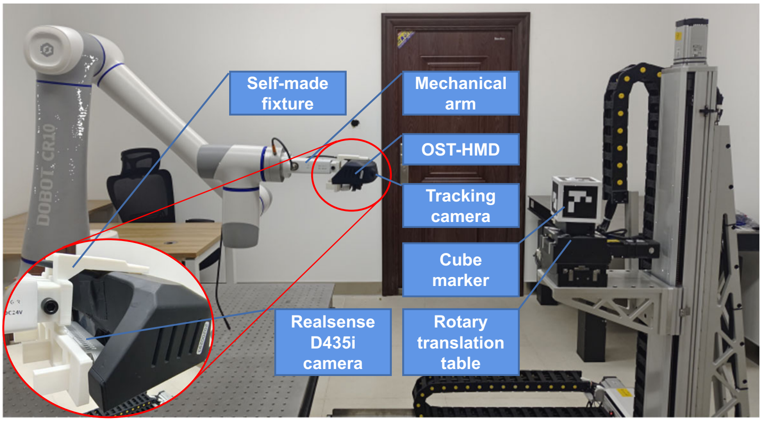


Fig. 7 Experimental environment. In the left half of the setup, the OST-HMD device is securely mounted on a robotic arm using a self-made fixture produced through 3D printing. The RealSense D435i camera, referred to as the observation camera, is employed instead of the human eye to record alignment effects. In the right half of the setup, the cube marker is affixed to a rotary and translation table to control its movement

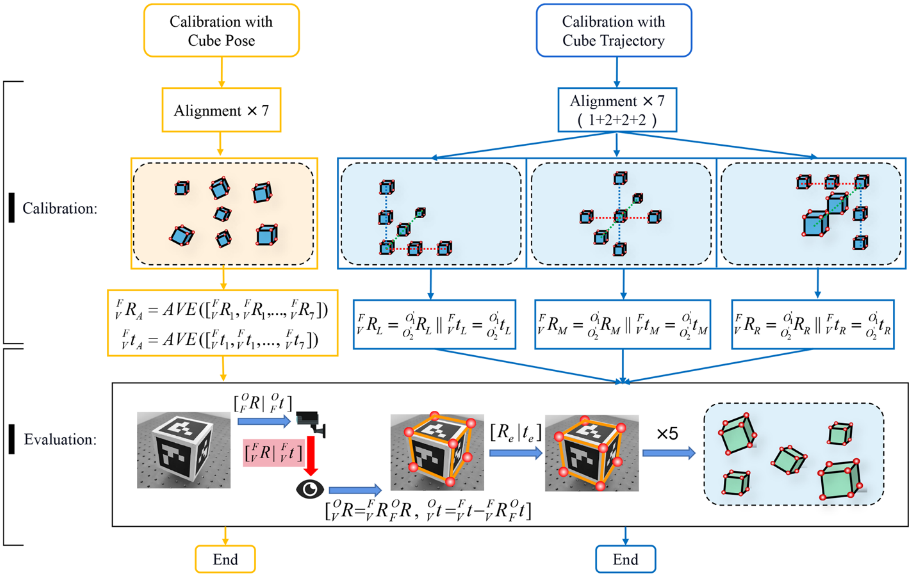

Calibration with Cube Pose
Calibration with Cube Trajectory
Alignment × 7
Alignment × 7 (1+2+2+2)
Calibration:
$^{F}_{V}R_A = AVE([^{F}_{V}R_1, ^{F}_{V}R_1, ..., ^{F}_{V}R_7])$
$^{F}_{V}t_A = AVE([^{F}_{V}t_1, ^{F}_{V}t_1, ..., ^{F}_{V}t_7])$
$^{F}_{V}R_L = ^{O_1'}_{O_2'}R_L \parallel ^{F}_{V}t_L = ^{O_1'}_{O_2'}t_L$
$^{F}_{V}R_M = ^{O_1'}_{O_2'}R_M \parallel ^{F}_{V}t_M = ^{O_1'}_{O_2'}t_M$
$^{F}_{V}R_R = ^{O_1'}_{O_2'}R_R \parallel ^{F}_{V}t_R = ^{O_1'}_{O_2'}t_R$
Evaluation:
$[^{O}_{F}R| ^{O}_{F}t]$
$[^{F}_{V}R| ^{F}_{V}t]$
$[^{O}_{V}R = ^{F}_{V}R ^{O}_{F}R, ^{O}_{V}t = ^{F}_{V}t - ^{F}_{V}R ^{O}_{F}t]$
$[R_e | t_e]$
×5
End
End

Fig. 8 Calibration work and evaluation experimental process. The yellow box represents the conventional method of calibration, which directly utilizes the pose of markers. The blue box signifies our proposed calibration method, which leverages the motion trajectories of markers. We have explored three different trajectories in the figure. The black box encompasses the evaluation experiment for our designed reprojection virtual model

4.2.2 Render camera calibration

i) Calibration with cube pose During the data collection process, the virtual model is first randomly projected into the user's field of view in different positions and poses (Fig. 8). The cube marker is then moved using a rotating translation platform, and the alignment effect is observed in real-time through an observation camera. When it is determined to be fully aligned, as shown in Fig. 9, the alignment is confirmed by pressing the Enter key. At this point, two important pieces of information are recorded. The first is the pose of the cube marker

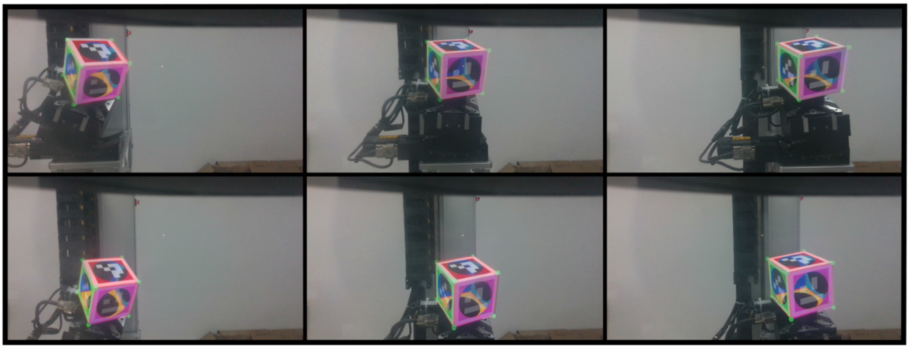


Fig. 9 Alignment effect of cube marker and virtual model

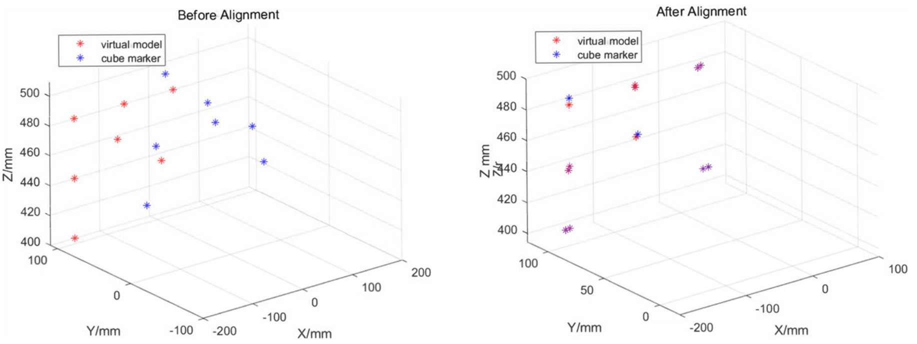


Fig. 10 Alignment effect of trajectory L. Red star * denotes a series of positions for the virtual model, while blue star * represents a series of positions for the cube marker. The left image illustrates the relative positional relationship between the two trajectories before the alignment operation, and the right image showcases the overlapping effect of the two trajectories after the alignment operation

in the coordinate system of the tracking camera, denoted as $\left[{}^{O_1}_{F}R \,|{}^{O_1}_{F}\, t\right]$. The second is the pose of the virtual cube model in the coordinate system of the rendering camera, denoted as $\left[{}^{O_1}_{V}R \,|{}^{O_1}_{V}\, t\right]$. Using the (3) in Section 3.1 of the paper, the relative position and orientation between the cube marker and virtual cube model, denoted as $\left[{}^{F}_{V}R_1 \,|{}^{F}_{V}\, t_1\right]$, are calculated. To ensure a uniform distribution of the virtual cube model in different positions and poses, the alignment operation is repeated seven times.

ii) Calibration with cube trajectory In our proposed method, the posture of the cube marker is fixed. Once the last alignment state is determined, the virtual cube model is translated by a fixed step length along a specific coordinate axis of the rendering camera. The proposed method still requires performing the alignment operation seven times. However, in each iteration, only the position information of the cube marker in the alignment state is collected, while ignoring the rotation of the cube marker. This position information is used to establish the trajectory coordinate system described in Section 3.2. The transformation matrix is then calculated by aligning the two trajectory coordinate systems. To collect data from different positions within the field of view, we aim to capture data from the left, middle, and right sides. These positions correspond to trajectory L, trajectory M, and trajectory R, respectively. You can refer to Fig. 8 for a visual representation of these different positions

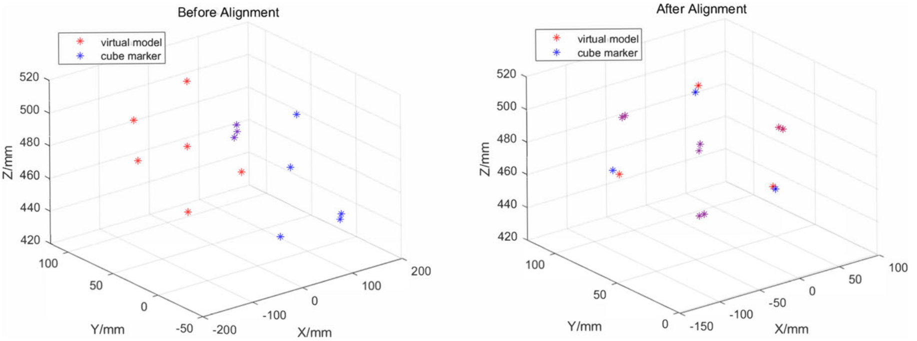


Fig. 11 Alignment effect of trajectory M

within the field of view. Figures 10, 11 and 12 respectively depict the trajectories L, M, and R along with their alignment effects.

5 Experimental verification for calibration

To showcase the accuracy of our calibration method, a dedicated evaluation experiment was designed, as depicted in Fig. 8(Evaluation). In this experiment, the cube marker can be placed at any position within the field of view. To correct the pose of the marker obtained from the tracking camera ($\left[{}^{O}_{F}R \mid {}^{O}_{F}t\right]$), we utilize the calibrated transformation matrix ${}^{F}_{V}T$ (as defined in (4)). This transformation matrix enables us to adjust the pose of the marker accurately.

Obtain the actual pose of the 3D model anchored in the virtual rendering space:

$$\begin{cases} {}^{O}_{V}R = {}^{F}_{V}R\,{}^{O}_{F}R \\ {}^{O}_{V}t = {}^{F}_{V}t \; - \; {}^{F}_{V}R\,{}^{O}_{F}t \end{cases} \tag{10}$$

This evaluation method requires collecting five additional data and calibrating the transformation matrix from the above seven sets of aligned data to project the cube model in a virtual scene. From the observation camera, it is intuitive to see the alignment effect between the reprojection model and the cube markers. The traditional 3D-3D method and our method are implemented on the same OST-HMD device for a more fair comparison. Figure 13 shows the qualitative results of some calibration experiments.

5.1 Quantitative evaluation indicators

Considering that many OST-HMD devices cannot easily access the pixel coordinates of their translucent screens, we have designed more intuitive and widely adaptable indicators to analyze experimental results. Use a rotating translation platform to obtain the 6DoF information of the cube marker in real time, move the cube marker to align the corrected virtual model again, and record the data before and after the realignment (Fig. 14). Then calculate the posture error between the corrected virtual model and the cube marker.

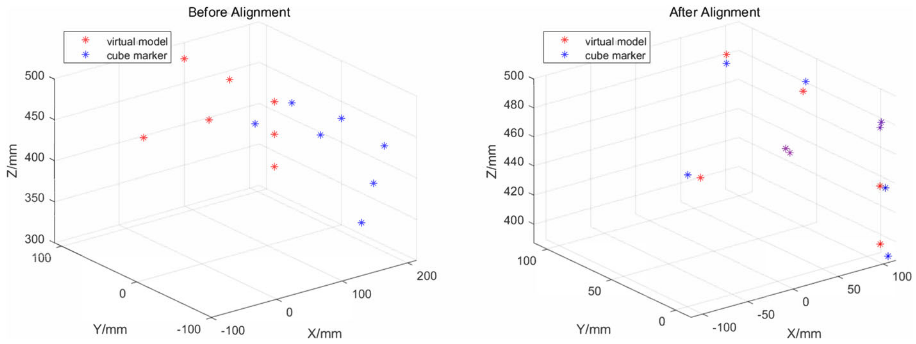


Fig. 12 Alignment effect of trajectory R

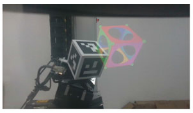
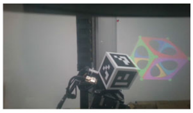
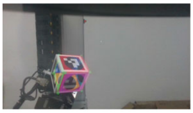
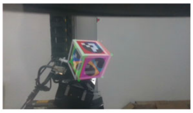
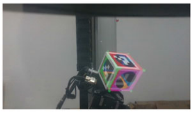
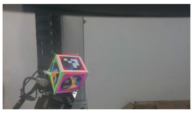
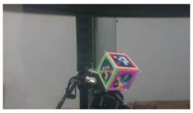
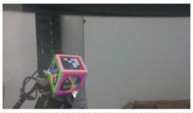
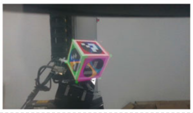
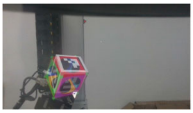
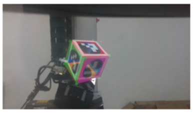
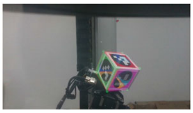
Method		Sample 1	Sample 2	Sample 3
No calibration				
Calibration with Cube Pose				
Calibration with Cube Trajectory	Track L			
	Track M			
	Track R			

Fig. 13 Qualitative results of calibration experiments: 3D visualization demonstration of the alignment effect between virtual and real models using various calibration methods. The results are presented from top to bottom, showcasing the uncalibrated state, calibration using traditional methods, and calibration using our proposed method (utilizing trajectories L, M, and R respectively)

Equation (11) shows the indicators used to describe the calibration accuracy:

$$\begin{cases} O_{3D}^{err} = \sqrt{\triangle x^2 + \triangle y^2 + \triangle z^2} \\ Lg = \arccos \dfrac{tr\left(R_2 R_g^T\right) - 1}{2} \end{cases}, \tag{11}$$

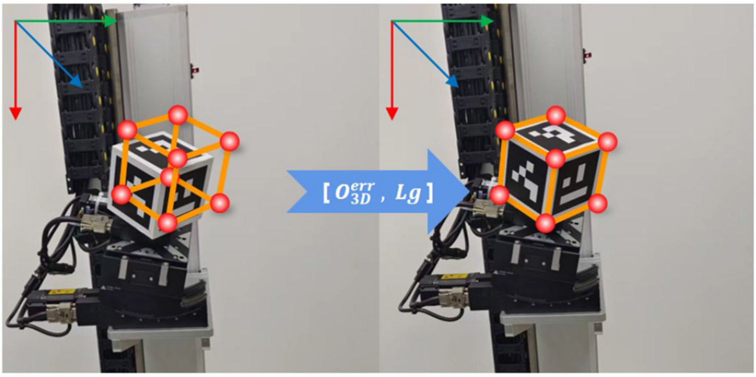


Fig. 14 Schematic diagram of realignment operation. The process involves realigning the cubic marker by moving the rotation and translation platform, aligning it with the post-calibration projected virtual model, and recording the coordinate transformation of the end point of the rotation and translation platform

where the displacement error (O_{3D}^{err}) is expressed by the Euclidean distance in three-dimensional space, and $[\triangle x, \triangle y, \triangle z]$ is the offset of the cube marker before and after realignment; Rotation error (Lg) is expressed by geodesic distance, and its physical meaning is the shortest path between two three-dimensional rotations. R_g and R_2 are the rotation matrices of the cube markers before and after realignment, respectively. The function $tr()$ is used to calculate the trace of a matrix, which numerically equals the sum of the elements on the main diagonal of the square matrix.

5.2 Result and discussion

The calibration method was evaluated using the aforementioned evaluation methods and performance metrics, through three different experiments. These experiments aimed to validate the superiority of the method in terms of calibration accuracy, system robustness, and data requirements.

5.2.1 Standard experiment

Calibration and evaluation tests were conducted on the same homemade OST-HMD device to ensure fairness. In each test, seven alignment operations were performed for calibration, and five reprojection operations were performed for evaluation. For ease of description, the traditional method of directly utilizing the pose of the cube marker is called ”Method A”, and the method we proposed is called ”Method L/M/R”, corresponding to the left/middle/right trajectory of the field of vision.

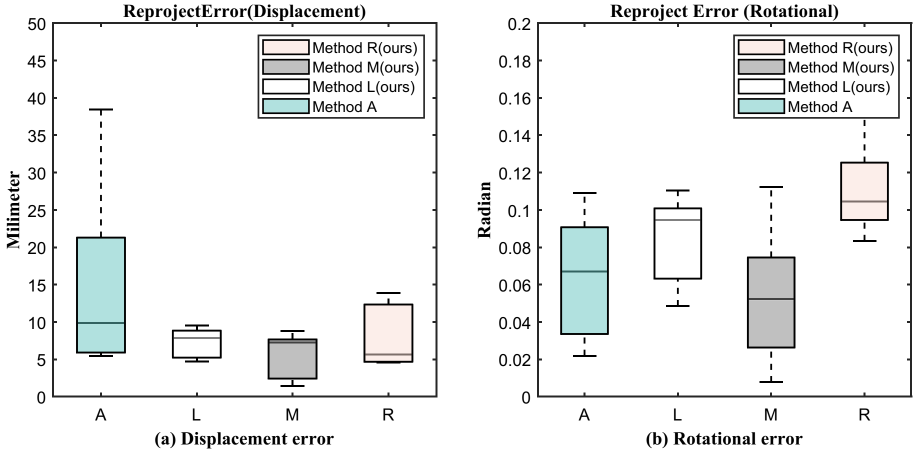
As shown in Table 1, the mean and standard deviation of displacement errors for each calibration method are Method A (15.1 ± 13.9 mm), Method L (7.2 ± 2.1 mm), Method M (5.5 ± 3.2 mm), and Method R (8.1 ± 4.4 mm). The mean and standard deviation of rotation errors for each method are Method A (0.06 ± 0.04 rad), Method L (0.08 ± 0.02 rad), Method M (0.05 ± 0.03 rad), and Method R (0.1 ± 0.02 rad).

In Fig. 15, the 3D reprojected analysis box plot is presented. The box’s centerline represents the median of the data, signifying the average level of the sample data. The upper and lower limits of the box correspond to the upper quartile and lower quartile of the data, reflecting the variability of the data. The lines above and below the box represent the maximum and minimum values of the data. The plot includes two error analyses: (a) Displacement error O_{3D}^{err}, where the y-axis represents the Euclidean distance between the reprojected model position and the true position. Regardless of which trajectory is selected for calibration in

Table 1 Reprojection errors of various calibration methods

Method	Displacement error(mm)		Rotation error(rad)	
	mean	std	mean	std
A	15.1	13.9	0.06	0.04
L	7.2	2.1	0.08	0.02
M	5.5	3.2	0.05	0.03
R	8.1	4.4	0.1	0.02

Method A represents the traditional 3D-3D calibration approach, directly utilizing the pose of the cubic marker. Methods L, M, and R, all developed by us, respectively employ the trajectories depicted in Figs. 10, 11 and 12. Evaluate the displacement error using O_{3D}^{err} in (11) and assess the rotation error using Lg



(a) Displacement error

(b) Rotational error

Fig. 15 3D reprojected analysis box plot with displacement and rotation errors. Method A represents the traditional 3D-3D calibration approach, directly utilizing the pose of the cubic marker. Methods L, M, and R, all developed by us, respectively employ the trajectories depicted in Figs. 10, 11 and 12. (a) Assess the displacement error using O_{3D}^{err} in (11), measured in millimeters. (b) Evaluate the rotation error using Lg in (11), measured in radians

our method (L/M/R), the error is smaller than method A and more stable. It is worth noting that when using the left and right side trajectories (L/R) of the field of view for calibration, the error is slightly larger than the middle track. We think this is due to slight edge distortion caused by the large field angle (135°) of the self-made OST-HMD device. (b) Rotation error Lg, where the y-axis represents the geodetic line distance between the reprojected model angle and the true angle. It can be seen that there is no significant difference between these methods, and they are all at a low error level, indicating that such methods using three-dimensional marker alignment perform well in calibrating the rotation angle.

The displacement error components for each coordinate axis are recorded in Table 2, while the rotation error components are recorded in Table 3. The YOZ plane is perpendicular to the user's sight line, and the X-axis is parallel to the user's sight line. Compared to other results, Method A has a significant error in depth (X-axis), which is the main factor causing the inaccuracy of this method. Analysis shows that this is due to a small amount of abnormal data in the pose of the cube marker estimated by the tracking camera during calibration, and traditional methods that directly utilize the pose of the cube marker cannot

Table 2 Displacement error along different axes(X/Y/Z) for calibration

Method	Axis X(mm)		Axis Y(mm)		Axis Z(mm)	
	mean	std	mean	std	mean	std
A	14.69	13.89	2.10	2.61	2.49	0.55
L	1.52	3.03	5.47	2.77	2.13	1.36
M	1.65	1.51	3.88	2.46	1.69	2.83
R	0.19	0.42	6.51	4.87	4.10	2.01

Method A represents the traditional 3D-3D calibration approach, directly utilizing the pose of the cubic marker. Methods L, M, and R, all developed by us, respectively employ the trajectories depicted in Figs. 10, 11 and 12

Table 3 Rotation error along different axes(X/Y/Z) for calibration

Method	Axis X(rad)		Axis Y(rad)		Axis Z(rad)	
	mean	std	mean	std	mean	std
A	0.015	0.015	0.041	0.02	0.028	0.015
L	0.029	0.015	0.026	0.015	0.061	0.016
M	0.012	0.021	0.011	0.015	0.045	0.025
R	0.015	0.019	0.081	0.011	0.104	0.016

effectively eliminate this interference information. The rotational errors for all the methods are significantly small and within the normal range.

5.2.2 Impact of abnormal data

To further validate the robustness of the calibration method based on marker motion trajectories, we artificially added some outliers to the cube marker trajectory data used in the previous experiment and re-aligned the trajectories. Figures 16, 17 and 18 illustrate the effects of adding outliers and the trajectory alignment in the three trajectories (L/M/R). It can be observed that our method does not overly fit individual outlier data points (highlighted in red circles) during the trajectory alignment process, but rather tends to prioritize the overall overlap of the motion trajectories.

Using the three trajectories with outlier data mentioned above, we performed calibration on the same OST-HMD device. The reprojected results after calibration are shown in Fig. 19, demonstrating good alignment. The mean displacement errors and standard deviations are as follows: 8.59±5.72mm, 6.81±3.73mm, and 9.22±4.95mm.

Furthermore, to further investigate the robustness of our method against interference and to ensure a more random and realistic distribution of errors in the data, we employed two distinct 2D identification codes (ArUco and TopoTag [40, 41]) to track the pose of the cubic markers during data collection. The appearance of the two 2D identification codes is illustrated in Fig. 20. Due to the fact that, during the pose estimation stage, the TopoTag method utilizes all feature points on its surface for calculations, whereas ArUco only employs the four corner

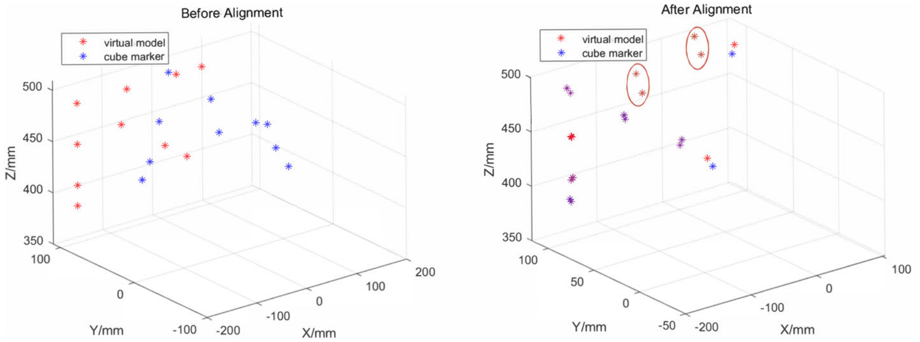


Fig. 16 Trajectory L with abnormal data. Red star * denotes a series of positions for the virtual model, while blue star * represents a series of positions for the cube marker. The left image illustrates the relative positional relationship between the two trajectories before the alignment operation, and the right image showcases the overlapping effect of the two trajectories after the alignment operation. The data within the red circles represents anomalies that have been manually introduced by us

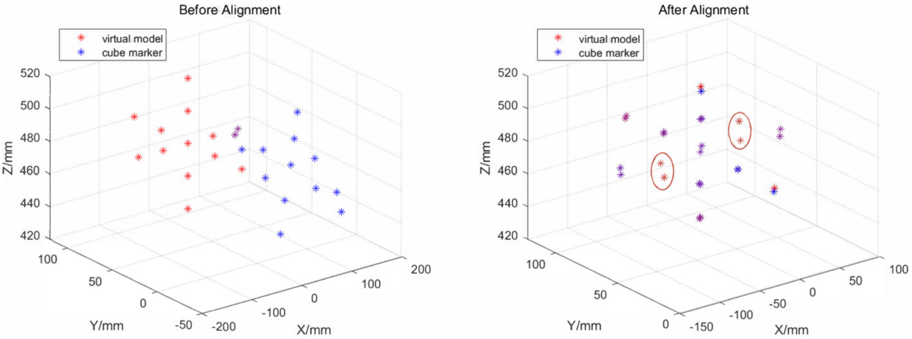


Fig. 17 Trajectory M with abnormal data

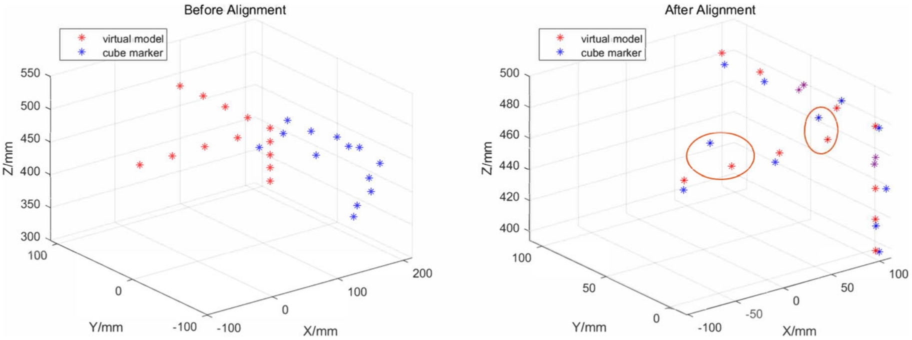


Fig. 18 Trajectory R with abnormal data

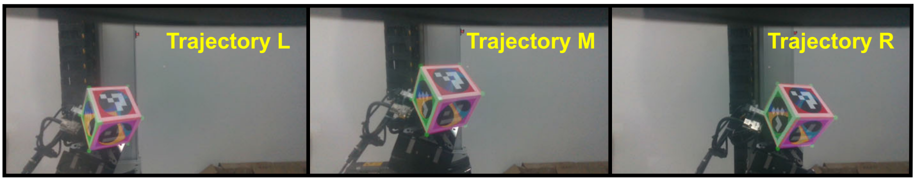
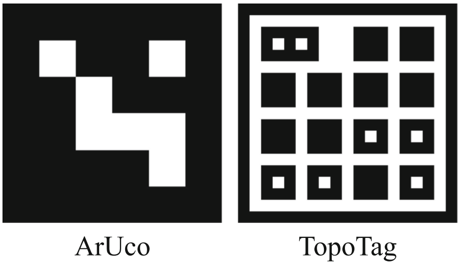


Fig. 19 Qualitative results of the trajectory with abnormal data

Fig. 20 The appearance of the two 2D identification codes (ArUco and TopoTag)



points on its surface, this results in a higher precision in tracking the cubic markers when using TopoTag. This precision difference is particularly evident in the depth dimension. In our experimental setup, the mean pose estimation error for the cubic markers tracked by TopoTag is consistently around 4mm lower than that of ArUco.

The data collected using both identification codes were applied to both traditional calibration methods and the calibration method we proposed, with the experimental results depicted in Fig. 21. Overall, whether utilizing traditional methods or our proposed calibration approach, the calibration results obtained using the TopoTag tracking method for data collection outperform those from the ArUco method. This superiority arises from the more precise marker tracking method effectively reducing errors present in the data used for calibration. Upon detailed analysis of the experimental outcomes, it becomes evident that the more accurate calibration data significantly enhances the traditional calibration method, which directly utilizes marker poses. The substantial disparity in results between the two different identification codes applied to the traditional calibration method also to some extent reflects the poor robustness of the traditional approach to interference. However, our proposed method based on marker motion trajectories shows no significant differences in calibration results when using two different precision tracking algorithms. It is less affected by errors introduced by the marker tracking algorithms, demonstrating higher robustness and adaptability.

Our proposed method exhibits minimal influence from individual alignment data and possesses good resilience against interference. It effectively addresses errors introduced by manual alignment operations and marker tracking algorithms, providing reliable and stable calibration in the process.

5.2.3 Impact of the number of alignment operations

To further validate the data requirements of our proposed method, we varied the number of alignment operations during the data collection process and compared the calibration

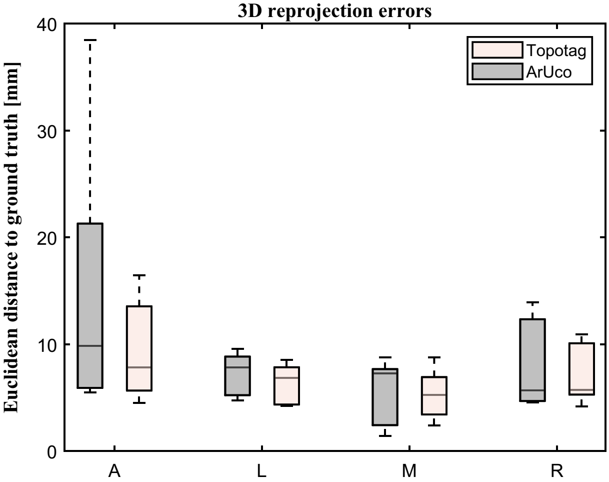


Fig. 21 Comparison of calibration results obtained through marker tracking using ArUco and TopoTag for different calibration methods during the data acquisition phase. The horizontal axis represents four calibration methods, while the vertical axis depicts the assessment of displacement error using O_{3D}^{err} in (11), measured in millimeters

Table 4 Displacement errors for each calibration method were assessed across various data volumes, gradually increasing manual alignment operation counts to 4, 7, and 10 times, respectively

Method	4		7		10	
	mean	std	mean	std	mean	std
A	18.8	13.9	15.1	13.9	10.2	8.7
L	8.6	2.8	7.2	2.1	6.9	1.7
M	6.3	3.4	5.5	3.2	5.3	3.1
R	8.9	4.7	8.1	4.4	8.0	4.3

results of Method A and Methods L/M/R. During the actual testing, the number of alignment operations was increased or decreased by 3 times (7±3 times). The displacement errors for each method, as the data volume increased, are shown in Table 4.

The specific experimental results were compiled into a box plot format (Fig. 22). In the figure, the y-axis represents the Euclidean distance between the reprojected model position and the true position, and the x-axis represents the number of alignment operations. The specific legend meanings are the same as in Fig. 15. Method A is heavily influenced by the amount of data and requires users to perform multiple alignment operations to gradually improve the accuracy of the calibration results, significantly increasing the workload of data collection. Methods L/M/R are less affected by the data volume and achieve good calibration accuracy with a small amount of data collected in just 4 alignment operations. When the number of alignment operations increases to 7, the reprojection error noticeably decreases. The calibration results become stable at 7 alignment operations, and further improvement becomes small after 7 operations. It is unnecessary to continue increasing the data volume beyond this point as it would only add unnecessary workload without significantly improving the calibration accuracy. Compared to traditional 3D-3D calibration methods, our method requires less data to achieve better calibration results.

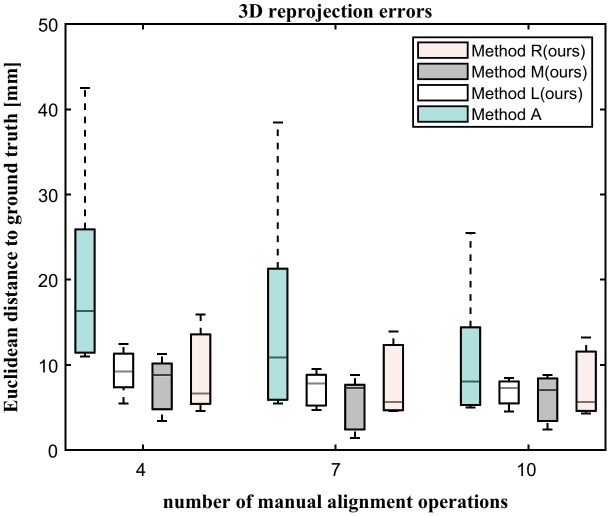


Fig. 22 Reprojection errors of various calibration methods under different data quantities. Conducting multi-times (4, 7, 10) manual alignment operations

Table 5 Comparison of displacement errors (O_{3D}^{err}) for calibration methods with different phases

Method	Manual alignment	Counts of alignments		
		4	7	10
Ours	1DoF	7.93mm	6.93mm	6.73mm
Pose-based [19]	6DoF	18.81mm	15.11mm	10.22mm

Our method performs only one degree of freedom(1DoF) changing during each manual alignment operation. The "Pose-based [19]" method requirs consideration of six degrees of freedom(6DoF) in each manual alignment operation. Both methods underwent iterative manual alignment operations, repeated 4, 7, and 10 times, to showcase the corresponding displacement errors

In order to further illustrate the simplicity and effectiveness of our proposed calibration method, we have divided it into smaller steps for a more intuitive comparison. Table 5 summarizes detailed information. In it, we conducted a comparative analysis based on two dimensions: the level of difficulty in a single alignment operation and the number of alignment operations that need to be repeated during the calibration process. From the results, it can be observed that our proposed method of using the trajectory of the cube markers for calibration requires simpler alignment operations during data collection. The alignment process only needs to consider a single degree of freedom change. Additionally, our method demonstrates superior calibration performance with lower errors, even when considering the same number or fewer manual alignment operations. Overall, our method offers a simpler data collection process, requiring less data while delivering improved performance.

6 Conclusions

In this study, we proposed a new method for calibrating OST-HMD devices using the motion trajectory of cube markers. Our method is based on the fundamental principles of 3D-3D calibration and does not require any prior knowledge of the physical parameters of the HMD. Additionally, it is applicable to a wide range of rendering principles used in OST-HMD devices. By leveraging cube marker motion trajectories for alignment instead of relying solely on corner information at a single moment, errors caused by outlier data can be eliminated, resulting in improved calibration accuracy and system robustness. This approach enhances the precision of calibration results and promotes a more reliable calibration process. Moreover, a new data acquisition method was designed, producing only a single degree of freedom change between two alignments, significantly reducing the alignment difficulty.

We applied our method to a self-made OST-HMD, observed the calibration effect through a built-in digital camera, and designed an evaluation experiment to report the numerical accuracy of the calibration results. The results show that our method outperformed the traditional 3D-3D calibration method in the same device, with a difference of approximately 9 mm. Additionally, our method achieves more accurate calibration results with less data and demonstrates excellent resistance to interference.

This paper successfully achieves millimeter-level accuracy in OST-HMD calibration. However, we recognize the importance of achieving sub-millimeter accuracy in many applications. In the future, we are committed to actively exploring and updating the method to further enhance its precision and simplify its operational steps. We already have some preliminary ideas that are currently awaiting further validation. In our view, two main factors limit the accuracy of our calibration method: the precision of the marker tracking algorithm and the potential for errors in manually observing and determining alignment states. These are the areas where we will focus our efforts in the future stages of this research project.

Funding This work was supported by grants from the Hebei Natural Science Foundation. under Grant No. F2021202021, and the S&T Program of Hebei, under Grant No. 22375001D.

Data availability statement The datasets generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

Declarations

Conflicts of interest The authors declare that they have no competing interests.

References

1. Cosentino F, John NW, Vaarkamp J (2014) An overview of augmented and virtual reality applications in radiotherapy and future developments enabled by modern tablet devices. J Radiother Pract 13(3):350–364. https://doi.org/10.1017/S1460396913000277
2. Hossain MS, Hardy S, Alamri A, Alelaiwi A, Hardy V, Wilhelm C (2016) Ar-based serious game framework for post-stroke rehabilitation. Multimed Syst 22(6):659–674. https://doi.org/10.1007/s00530-015-0481-6
3. Liu H, Auvinet E, Giles J, Rodriguez y Baena F (2018) Augmented reality based navigation for computer assisted hip resurfacing: a proof of concept study. Ann Biomed Eng 46:1595–1605. https://doi.org/10.1007/s10439-018-2055-1
4. Chen L, Day TW, Tang W, John NW (2017) Recent developments and future challenges in medical mixed reality. Paper presented at the 2017 IEEE international symposium on mixed and augmented reality (ISMAR), pp 12–135. https://doi.org/10.1109/ISMAR.2017.29
5. Qian L, Azimi E, Kazanzides P, Navab N (2017) Comprehensive tracker based display calibration for holographic optical see-through head-mounted display. arXiv:1703.05834
6. Cutolo F (2019) Letter to the editor on "augmented reality based navigation for computer assisted hip resurfacing: a proof of concept study". Ann Biomed Eng 47(11):2151–2153. https://doi.org/10.1007/s10439-019-02300-6
7. Li R, Han B, Li H, Ma L, Zhang X, Zhao Z, Liao H (2023) A comparative evaluation of optical see-through augmented reality in surgical guidance. IEEE Trans Visual Comput Graph. https://doi.org/10.1109/TVCG.2023.3260001
8. Fu Y, Schwaitzberg SD, Cavuoto L (2023) Effects of optical see-through head-mounted display use for simulated laparoscopic surgery. Int J Human-Comput Interact 1–16. https://doi.org/10.1080/10447318.2023.2219966
9. Ernst JM, Laudien T, Schmerwitz S (2023) Implementation of a mixed-reality flight simulator: blending real and virtual with a video-see-through head-mounted display. Artificial intelligence and machine learning for multi-domain operations applications V 12538:181–190. https://doi.org/10.1117/12.2664848
10. Wang C-H, Chang C-C, Hsiao C-Y, Ho M-C (2022) Effects of reading text on an optical see-through head-mounted display during treadmill walking in a virtual environment. Int J Human-Comput Interact 1–14. https://doi.org/10.1080/10447318.2022.2154891
11. Jun H, Kim G (2016) A calibration method for optical see-through head-mounted displays with a depth camera. Paper presented at the 2016 IEEE virtual reality (VR), pp 103–111. https://doi.org/10.1109/VR.2016.7504693

12. Owen CB, Zhou J, Tang A, Xiao F (2004) Display-relative calibration for optical see-through head-mounted displays. Paper presented at the third IEEE and ACM international symposium on mixed and augmented reality, pp 70–78. https://doi.org/10.1109/ISMAR.2004.28
13. Tuceryan M, Genc Y, Navab N (2002) Single-point active alignment method (spaam) for optical see-through hmd calibration for augmented reality. Presence: Teleoper Virtual Environ 11(3):259–276. https://doi.org/10.1162/105474602317473213
14. Azimi E, Qian L, Kazanzides P, Navab N (2017) Robust optical see-through head-mounted display calibration: taking anisotropic nature of user interaction errors into account. Paper presented at the 2017 IEEE virtual reality (VR), pp 219–220. https://doi.org/10.1109/VR.2017.7892255
15. Moser KR, Swan JE (2016) Evaluation of hand and stylus based calibration for optical see-through head-mounted displays using leap motion. Paper presented at the 2017 IEEE virtual reality (VR), pp 233–234. https://doi.org/10.1109/VR.2016.7504739
16. Itoh Y, Klinker G (2014) Interaction-free calibration for optical see-through head-mounted displays based on 3d eye localization. Paper presented at the 2014 IEEE symposium on 3d user interfaces (3dui), pp 75–82. https://doi.org/10.1109/3DUI.2014.6798846
17. Plopski A, Itoh Y, Nitschke C, Kiyokawa K, Klinker G, Takemura H (2015) Corneal-imaging calibration for optical see-through head-mounted displays. IEEE Trans Visual Comput Graph 21(4):481–490. https://doi.org/10.1109/TVCG.2015.2391857
18. Plopski A, Orlosky J, Itoh Y, Nitschke C, Kiyokawa K, Klinker G (2016) Automated spatial calibration of hmd systems with unconstrained eye-cameras. Paper presented at the 2016 IEEE international symposium on mixed and augmented reality (ISMAR), pp 94–99. https://doi.org/10.1109/ISMAR.2016.16
19. Azimi E, Qian L, Navab N, Kazanzides P (2017) Alignment of the virtual scene to the tracking space of a mixed reality head-mounted display. arXiv:1703.05834
20. Moser KR, Swan JE (2016) Evaluation of user-centric optical see-through head-mounted display calibration using a leap motion controller. Paper presented at the 2016 IEEE symposium on 3D user interfaces (3DUI), pp 159–167. https://doi.org/10.1109/3DUI.2016.7460047
21. Shi Z, Shang Y, Zhang X, Wang G (2021) Dlt-lines based camera calibration with lens radial and tangential distortion. Exper Mechan 61(8):1237–1247. https://doi.org/10.1007/s11340-021-00726-5
22. Huang K, Ziauddin S, Zand M, Greenspan M (2020) One shot radial distortion correction by direct linear transformation. Paper presented at the 2020 IEEE international conference on image processing (ICIP), pp 473–477. https://doi.org/10.1109/ICIP40778.2020.9190749
23. Genc Y, Tuceryan M, Navab N (2002) Practical solutions for calibration of optical see-through devices. In: Proceedings of the international symposium on mixed and augmented reality, pp 169–175. https://doi.org/10.1109/ISMAR.2002.1115086
24. Qian L, Winkler A, Fuerst B, Kazanzides P, Navab N (2016) Reduction of interaction space in single point active alignment method for optical see-through head-mounted display calibration. Paper presented at the 2016 IEEE international symposium on mixed and augmented reality (ISMAR-Adjunct), pp 156–157. https://doi.org/10.1109/ISMAR-Adjunct.2016.0066
25. Makibuchi N, Kato H, Yoneyama A (2013) Vision-based robust calibration for optical see-through head-mounted displays. Paper presented at the 2013 IEEE international conference on image processing, pp 2177–2181. https://doi.org/10.1109/ICIP.2013.6738449
26. Fuhrmann A, Schmalstieg D, Purgathofer W (1999) Fast calibration for augmented reality. In: Proceedings of the ACM symposium on virtual reality software and technology, pp 166–167. https://doi.org/10.1145/323663.323692
27. Grubert J, Tuemle J, Mecke R, Schenk M (2010) Comparative user study of two see-through calibration methods. VR 10(269–270): 16. https://doi.org/10.13140/2.1.1504.2249
28. Grubert J, Itoh Y, Moser K, Swan JE (2017) A survey of calibration methods for optical see-through head-mounted displays. IEEE Trans Visual Comput Graph 24(9):2649–2662. https://doi.org/10.1109/TVCG.2017.2754257
29. Zhang Z, Weng D, Liu Y, Wang Y, Zhao X (2017) Ride: region-induced data enhancement method for dynamic calibration of optical see-through head-mounted displays. Paper presented at the 2017 IEEE virtual reality (VR), pp 245–246. https://doi.org/10.1109/VR.2017.7892268
30. Owen CB, Zhou J, Tang A, Xiao F (2004) Display-relative calibration for optical see-through head-mounted displays. Paper presented at the third IEEE and ACM international symposium on mixed and augmented reality, pp 70–78. https://doi.org/10.1109/ISMAR.2004.28
31. Genc Y, Sauer F, Wenzel F, Tuceryan M, Navab N (2000) Optical see-through hmd calibration: a stereo method validated with a video see-through system. In: Proceedings IEEE and ACM international symposium on augmented reality (ISAR 2000), pp 165–174. https://doi.org/10.1109/ISAR.2000.880940

32. Sun Q, Mai Y, Yang R, Ji T, Jiang X, Chen X (2020) Fast and accurate online calibration of optical see-through head-mounted display for ar-based surgical navigation using microsoft hololens. Int J Comput Assist Radiol Surg 15:1907–1919. https://doi.org/10.1007/s11548-020-02246-4
33. Hu X, y Baena FR, Cutolo F (2021) Rotation-constrained optical see-through headset calibration with bare-hand alignment. Paper presented at the 2021 IEEE international symposium on mixed and augmented reality (ISMAR), pp 256–264. https://doi.org/10.1109/ISMAR52148.2021.00041
34. Itoh Y, Klinker G (2014) Interaction-free calibration for optical see-through head-mounted displays based on 3d eye localization. Paper presented at the 2014 IEEE symposium on 3d user interfaces (3dui), pp 75–82. https://doi.org/10.1109/3DUI.2014.6798846
35. Plopski A, Itoh Y, Nitschke C, Kiyokawa K, Klinker G, Takemura H (2017) Corneal-imaging calibration for optical see-through head-mounted displays. IEEE Trans Visual Comput Graph 21(4):481–490. https://doi.org/10.1109/TVCG.2015.2391857
36. Itoh Y, Klinker G (2014) Performance and sensitivity analysis of indica: interaction-free display calibration for optical see-through head-mounted displays. Paper presented at the 2014 IEEE international symposium on mixed and augmented reality (ISMAR), pp 171–176. https://doi.org/10.1109/ISMAR.2014.6948424
37. Itoh Y, Amano T, Iwai D, Klinker G (2016) Gaussian light field: estimation of viewpoint-dependent blur for optical see-through head-mounted displays. IEEE Trans Visual Comput Graph 22(11):2368–2376. https://doi.org/10.1109/TVCG.2016.2593779
38. Garrido-Jurado S, Muoz-Salinas R, Madrid-Cuevas FJ (2014) Automatic generation and detection of highly reliable fiducial markers under occlusions. Pattern Recogn 47(6):2280–2292. https://doi.org/10.1016/j.patcog.2014.01.005
39. Li S, Xu C, Xie M (2012) A robust o (n) solution to the perspective-n-point problem. IEEE Trans Pattern Anal Mach Intell 34(7):1444–1450. https://doi.org/10.1109/TPAMI.2012.41
40. Yu G, Hu Y, Dai J (2020) Topotag: a robust and scalable topological fiducial marker system. IEEE Trans Visual Comput Graph 27(9):3769–3780. https://doi.org/10.1109/TVCG.2020.2988466
41. Zhang Z, Hu Y, Yu G, Dai J (2022) Deeptag: a general framework for fiducial marker design and detection. IEEE Trans Pattern Anal Mach Intell 45(3):2931–2944. https://doi.org/10.1109/TPAMI.2022.3174603

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.