A Calibration Method for Optical See-through Head-mounted Displays with a Depth Camera

Hanseul Jun*　　Gunhee Kim†

Seoul National University

Abstract

We propose a fast and accurate calibration method for the optical see-through (OST) head-mounted displays (HMD), taking advantage of a low-cost time-of-flight depth-camera. Recently, affordable OST-HMDs and depth-cameras are widely appearing in the commercial market. In order to correctly reflect the user experience into the calibration process, our method demands a user wearing the HMD to repeatedly point at rendered virtual circles with their fingertips. From the repeated calibration data, we perform two stages of *full calibration* and *simplified calibration*, to compute key calibration parameters. The full calibration is required when the depth-camera is first installed to the HMD, and afterwards only the simplified calibration is performed whenever a user wears it again. Our experimental results show that the full and simplified calibration can be achieved with 10 and 5 user's repetitions (theoretically 3 and 2 at minimum), which are significantly less than about 20 of the stereo-SPAAM, one of the most popular existing calibration techniques. We also demonstrate that the 3D position errors of our calibration become much quickly smaller than those of the state-of-the-art method.

Keywords: Calibration, optical see-through head-mounted display, depth-camera.

1 Introduction

Optical see-through head-mounted displays (OST-HMDs) are devices that allow a user to see virtual 3D objects without blocking the outside view. In order to create an immersive user experience, the virtual objects and physical environment must be accurately positioned relative to each other, which requires a correct calibration of OST-HMDs. Contrary to the calibration of the normal cameras used in computer vision and photogrammetry, the calibration of OST-HMDs has an unique obstacle that it is impossible to directly observe retinal images of users. Therefore, it has been a challenge to design a cost-effective and accurate solution for general users to easily accomplish.

The objective of this paper is to propose a simple and fast calibration method for the low-cost OST-HMDs. Especially, our method comes through the wide availability of affordable time-of-flight depth-cameras and their integration with the OST-HMDs. Such integrated consumer devices have recently began to be released such as Meta 1. Or one can build their own system by simply installing a depth-camera to an OST-HMD. In the proposed calibration process, we demand a user wearing the OST-HMD to repeat pointing at virtual circles with their fingertips. Using the depth-camera, we can directly obtain the 3D coordinates of fingertips; from such repeated calibration data, our method estimates the key calibration parameters. We consider the two stages of *full calibration* and *simplified calibration*. We perform the full calibration only when the depth-camera is first installed to the HMD. Afterwards, unless the depth-camera moves on the HMD, only the simplified calibration is sufficient whenever a user takes off and wears it again.

In our experiments, we show that the full and simplified calibration can be successfully achieved with only 10 and 5 user's repetitions (theoretically 3 and 2 at minimum) respectively, which are significantly less than about 20 of the stereo-SPAAM [4], one of the most popular calibration techniques. Moreover, we also demonstrate that the 3D position errors of our calibration method are much smaller than those of the state-of-the-art method.

The calibration of OST-HMDs has been extensively studied for the development of augmented reality (AR) systems. Janin *et al.* develop one of the earliest methods [9], which use simple linear transformation and projection to align the coordinates of the workpiece, the virtual screen, the position sensor, and the user's eyes in the same coordinate system. Azuma and Bishop analyze static and dynamic errors for the calibration of OST-HMDs [1]. They use an optoelectronic tracker for accurate static registration across a wide range of viewing angles and positions. They also exploit inertial sensors mounted on the HMD for cancellation of dynamic errors caused by a user's head motion. Holloway studies a list of registration error sources (*e.g.* optical distortion and tracker measurement errors), and reveals their magnitudes [6]. McGarrity *et al.* introduce a calibration method that allows a user to interactively adjust the parameters by matching displayed virtual images with the real objects [11]. This method is *dynamic* in the sense that it does not require the user's head to be immobilized during calibration, using a 6-DOF magnetic tracker. Tuceryan and Navab propose one of the most successful and user-friendly methods called *Single Point Active Alignment Method* (SPAAM) [18]. Initially this method deals with only monocular OST-HMDs, and later is extended to stereo-SPAAM for binocular OST-HMDs in a following paper [4]. Using the magnetic tracker attached to the camera for measuring its 6-DOF position, this method requires the subject to repeatedly align a crosshair presented in the HMD to the one in a real space. The crosshair alignment is sequentially performed multiple times, for example 18 times (*i.e.* 9 for each of left and right eye) [17].

As the SPAAM has emerged as a practical solution to calibration, the following methods mainly focus on reducing calibration time, because the calibration involves users' intervention of several minutes, and ideally it should be re-done every time the HMD moves on users' head. Genc *et al.* propose two-staged SPAAM2 [5], in which an offline calibration is performed in advance to store extrinsic projection parameters that do not change according to the HMD moving, and then online calibration is performed for a specific user by collecting only a smaller number of data points. Owen *et al.* propose a similar two-phase calibration method named *Display-Relative Calibration* (DRC) [12]. This method shares the idea of the offline calibration, which is the first phase of measuring the parameters of the display system relative to the calibration coordinate system. Then it proposes several alternatives for the second phase that calibrates the eye positions relative to the display reference system. Kellner *et al.* propose another geometric calibration method with the two-phase concept [10], which can be done with less time than the SPAAM, by tracking a 6-DOF head-attached marker and a

*e-mail: hanseul@snu.ac.kr

†e-mail: gunhee@snu.ac.kr

IEEE Virtual Reality Conference 2016
19–23 March, Greenville, SC, USA
978-1-5090-0836-0/16/$31.00 ©2016 IEEE

3-DOF hand-attached marker.

The next advance of calibration methods has focused on improving the online calibration. Itoh and Klinker propose *Interaction-free Display Calibration* (INDICA), which performs online calibration automatically with an RGB eye tracking camera attached to the HMD [8]. It uses the SPAAM to obtain the parameters of offline calibration, and then makes online calibration easier and more reliable by utilizing dynamic 3D eye position measurements from an eye tracking camera. Plopski *et al*. develop a similar method named *Corneal-Imaging Calibration* (CIC) [13]. The difference of the CIC from the INDICA is that the INDICA estimates the eyeball position from the iris detection, whereas the CIC estimates the eye position using the corneal reflection. The common properties of the both methods are that they require an RGB camera attached to the HMD toward users' eyes, and assume the eye pinhole model known by exterior measurements. However, since the human vision system largely involves the visual process of the brain, the *perceptual* pinhole centers of eyes are hard to know through physical measurements. In other words, the anatomical pinhole centers of eyes may not coincide with the perceptual ones unless the visual process of the brain is completely understood [14, 15, 16]. Therefore, it can be more useful to include a certain amount of user interactions to make sure that users correctly see and feel.

To conclude the introduction, we highlight the main contributions of this paper as follows.

(1) To the best of our knowledge, this work is the first to utilize the depth-camera for the calibration of OST-HMDs. We consider two stages of *full* and *simplified* calibration. The full calibration is performed once when the depth-camera is first installed, while the simplified calibration is carried out whenever a user wears it again.

(2) Our calibration method is a time-efficient and user-friendly procedure in the two respects. First, it does not require any external devices, but exploit users' fingertips only. Second, as far as we know, it requires minimum users' feedback among existing methods. Our calibration is successful with about 10 and 5 times of user interactions for full and simplified calibration (theoretically 3 and 2 at minimum), respectively.

(3) In experiments, we show that our method is robust enough to work with an affordable OST-HMD and a low-cost depth-camera that do not require high-precision performance. We demonstrate the calibration accuracies of our method are higher than those of stereo-SPAAM [4], with fewer calibration data.

2 Overview

Our method calibrates the rendering process of OST-HMDs using an attached depth-camera. The scope of the OST-HMD system consists of three parts; a pair of displays, a depth-camera, and a user wearing it. For the displays, the correct intrinsic parameters (*e.g*. field of views) are usually provided by manufacturers. For the depth-camera, the parameters are categorized into intrinsic and extrinsic ones. We assume that the depth-camera is *intrinsically* calibrated and attached to the HMD, which is a reasonable assumption because in most cases the intrinsic parameters (*e.g*. field of views and offsets) are provided by manufacturers or, if not, they can be obtained by its own separate calibration [19]. For a user, we consider two types of parameters, which are interpupillary distance (IPD) and the position of user eyes. In summary, our calibration aims to calculate the extrinsic parameters of the depth-camera, and the two users parameters, while the intrinsic parameters of displays and the depth-camera are known a priori.

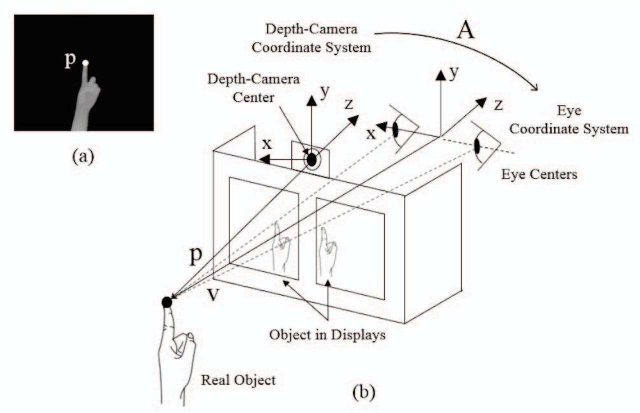


Figure 1: (a) A depth map measured from the depth-camera. (b) The coordinate systems and transformation used in our calibration model.

We demand users to repeat pointing at rendered virtual circles with their fingertips. From the repetition data, our method calculates key calibration parameters. We consider two types of calibration; full and simplified calibration. The full calibration calculates both the camera extrinsic parameters and the user parameters, while the simplified calibration obtains only the user parameters with less user repetitions. The details of both types will be presented in section 4.

It is worth noting that we use fingertips to obtain calibration data only because we want to perform calibration without using any extra device unlike other existing approaches, including stereo-SPAAM. Our calibration method described below is orthogonal to this setting; as long as a user can assign correspondences between actual and virtual targets as calibration data, our method can correctly calibrate, no matter where the calibration data exist in the space.

3 Mathematical Model

Our OST-HMD model consists of a pair of displays, a depth-camera, and a pair of eyes (See Figure 1). We assume that the depth-camera and the eye pair are based on the pinhole model, which abstracts a visual system with perspective projection to a pinhole-like center. Although we use the pinhole model for the eyes, we do not try to find out the *physical* pinhole centers of the eyes for the two reasons. First, it is hard to anatomically measure the accurate centers of the users' eyeballs. Second, since the perception actually occurs in the brain, the perceptual eye projections are likely to be different from the physical ones. Instead, our method aims at obtaining the *perceptual* pinhole centers of the users, based on the user interactions. We assume that the displays have infinite focal lengths without parallax, which is a well-known technique that mitigates eye fatigue in the implementation of head-up displays (HUDs). Similarly, in almost all OST-HMDs, such infinite focal lengths are also realized by an optical collimator that emits parallel lights; eventually it can prevent users from repeatedly refocusing from the real world to the virtual world or vice versa, which can bother the users. Finally, we assume that the displays are symmetric to each other, and thus have the same projection matrices except their flipped horizontal offsets.

Figure 1 illustrates the depth-camera coordinate system and the eye coordinate system used in our model. The depth-camera coordinate system has its origin at the camera's center and its *x*/*y* axis corresponds to the right/upward direction of the camera's images, using a right-handed coordinate system. The eye coordinate system has its origin at the center of two eyes; it defines the *x* axis toward the right eye center and the *z* axis in the direction opposite to the OST-HMD. We assume that the OST-HMD is worn in the way that the *z* axis is perpendicular to the surface containing OST-HMD displays. This is based on the nature of OST-HMDs, whose eye boxes of the displays, where the wearers' eyes exist, are very tight to disallow the eyes tilted, from which they can deliver the virtual view as large as possible.

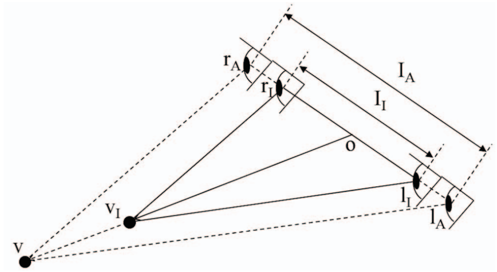


Figure 2: The triangular similarity for an initialized interpupillary distance (IPD) I_I $(=|\overline{l_Ir_I}|)$ and an actual IPD I_A $(=|\overline{l_Ar_A}|)$.

We represent a point in a 3D space by p from the depth-camera coordinate system and by v from the eye coordinate system. If the OST-HMD system is perfectly calibrated, the virtual objects in the display are shown in coincidence with the real object. With an example of Figure 1, the calibration makes the rendered hands perfectly overlap with the real hand. Therefore, the point v can be interpreted as a 3D position of a virtual point that corresponds to p. We use A to denote a linear transformation from p to v in a homogeneous coordinate system:

$$v = Ap, \tag{1}$$

where $A \in \mathbb{R}^{4\times4}$ and $v, p \in \mathbb{R}^{4\times1}$. The calibration is identical to finding the linear transformation A. We decompose A into a combination of three matrices for rotation R_A, translation t_A, and isotropic scaling s_A.

$$A = \begin{bmatrix} s_AI_3 & 0_{3,1} \\ 0_{1,3} & 1 \end{bmatrix} \begin{bmatrix} I_3 & t_A \\ 0_{1,3} & 1 \end{bmatrix} \begin{bmatrix} R_A & 0_{3,1} \\ 0_{1,3} & 1 \end{bmatrix}. \tag{2}$$

In summary, we have three parameters associated with the OST-HMD calibration: (i) camera rotation ($R_A \in \mathbb{R}^{3\times3}$), (ii) camera translation ($t_A \in \mathbb{R}^{3\times1}$), and (iii) scaling by a user's IPD ($s_A \in \mathbb{R}$). First, a point p obtained by the depth-camera is rotated by R_A and translated by t_A so that it is aligned with the eye coordinate system. Then, it is scaled isotropically by s_A so that the interpupillary distance is calibrated to the user, which will be discussed in section 3.1.

Historically, our formulation is based on [7], which first proposed a closed-form solution for such decomposition of a linear transformation to a combination of translation, rotation, and isotropic scaling.

3.1 Interpupillary Distance

The interpupillary distance (IPD) is the distance between the centers of a user's two eyes. As discussed, we use the fact that OST-HMDs are usually manufactured so that its displays have infinite focal lengths without parallax, which makes the ray direction of each pixel not change according to the eye translation. Using that the pixels have fixed ray directions, we can show the relation of the isotropic scaling between the IPD and the rendered point v in a 3D space as follows.

In Figure 2, suppose that l_I and r_I are the *initialized* (*i.e.* uncalibrated) left and right eye centers, whereas l_A and r_A are the *actual* (*i.e.* calibrated) eye centers. Obviously, the initialized IPD is $I_I = |\overline{l_Ir_I}|$, and the actual IPD is $I_A = |\overline{l_Ar_A}|$. For the initialized IPD, we set $I_I = 63$ mm based on a statistical study of [3]. v_I and v are the rendered points in 3D when we use the initialized and actual IPDs, respectively. o is the origin of the eye coordinate system. Considering $\overline{l_Ir_I}$ and $\overline{l_Ar_A}$ are on the x axis of the eye coordinate system and the direction of each pixel is fixed, we obtain

$$\overline{l_Ar_A} \parallel \overline{l_Ir_I}, \quad \overline{l_Av} \parallel \overline{l_Iv_I}, \quad \overline{r_Av} \parallel \overline{r_Iv_I}.$$

We then have the similarity between triangles of $\triangle vl_Ar_A \sim \triangle v_Il_Ir_I$, from which it is easy to obtain the ratio between $\overrightarrow{ov}$ and $\overrightarrow{ov_I}$ as

$$\overrightarrow{ov} = \overrightarrow{or_A} + \overrightarrow{r_Av} = \frac{I_A}{I_I}\overrightarrow{ov_I}. \tag{3}$$

We represent the transformation from $\overrightarrow{ov_I}$ to $\overrightarrow{ov}$ using s_A of Equation 2. From Equation 3, we have

$$v = \begin{bmatrix} s_AI_3 & 0_{3,1} \\ 0_{1,3} & 1 \end{bmatrix} v_I = \begin{bmatrix} \frac{I_A}{I_I}I_3 & 0_{3,1} \\ 0_{1,3} & 1 \end{bmatrix} v_I. \tag{4}$$

Consequently, it is straightforward to see

$$s_A = \frac{I_A}{I_I}. \tag{5}$$

4 Calibration Methods

The calibration of the OST-HMDs reduces to finding a correct A of Equation 1, which is a combination of translation, rotation, and isotropic scaling in Equation 2. We collect point pairs by asking users to point virtual circles with their fingertips. Here the positions of a virtual circle are $\{v_i\}$ and their corresponding positions pointed by a user are $\{p_i\}$. Our objective is to find A that minimizes the mean square error between $\{v_i\}$ and $\{p_i\}$, based on Equation 1:

$$\hat{A} = \operatorname*{argmin}_A \sum_i ||v_i - Ap_i||^2. \tag{6}$$

We consider two different calibration settings for practical reasons. One is the *full calibration* that computes all calibration parameters with no assumption, while the other is the *simplified calibration* with a known R_A. In other words, the full calibration is required when the depth-camera is first installed to the HMD, and the simplified one is performed as long as the depth-camera is firmly fixed on the HMD. For example, the simplified calibration is sufficient whenever a user is changed or he/she wears the HMD again. Obviously, the simplified calibration needs less user repetitions thanks to a lower degrees of freedom. We describe the full and simplified calibration in section 4.1 and 4.2, respectively.

4.1 Full Calibration

For full calibration, we solve A of Equation 6 without additional assumption. We here do not use the homogeneous coordinate, and represent A with t_A, R_A, and s_A (*i.e.* parameters in Equation 2):

$$v = s_A(R_Ap + t_A). \tag{7}$$

where $v, p, t_A \in \mathbb{R}^{3\times1}$, $R_A \in \mathbb{R}^{3\times3}$, and $s_A \in \mathbb{R}^{1\times1}$. Then Equation 6 becomes

$$\hat{t}_A, \hat{R}_A, \hat{s}_A = \operatorname*{argmin}_{t_A,R_A,s_A} \sum_i ||v_i - s_A(R_Ap_i + t_A)||^2. \tag{8}$$

For convenience, we define the centroids as

$$\bar{v} = \frac{1}{n}\sum_{i=1}^{n} v_i \quad \bar{p} = \frac{1}{n}\sum_{i=1}^{n} p_i, \tag{9}$$

where n is the number of collected calibration point pairs. We then standardize the data using centroids

$$v'_i = v_i - \bar{v} \quad p'_i = p_i - \bar{p}. \tag{10}$$

The objective of Equation 8 becomes

$$\sum_{i=1}^{n} ||v'_i - s_AR_Ap'_i||^2 - 2t'_A \cdot \sum_{i=1}^{n}(v'_i - s_AR_Ap'_i) + n||t'_A||^2$$
$$= \sum_{i=1}^{n} ||v'_i - s_AR_Ap'_i||^2 + n||t'_A||^2, \ (\because \sum_{i=1}^{n} v'_i = 0, \ \sum_{i=1}^{n} p'_i = 0) \tag{11}$$

where $t'_A = s_A t_A - (\bar{v} - s_A R_A \bar{p})$. In order to minimize Equation 11, the second term $n||t'_A||^2 \geq 0$ that is always nonnegative has to be zero. That is, $||t'_A|| = 0$, from which

$$s_A t_A - (\bar{v} - s_A R_A \bar{p}) = 0 \;\Rightarrow\; \hat{t}_A = \frac{\bar{v}}{\hat{s}_A} - \hat{R}_A \bar{p}. \qquad (12)$$

Once we obtain the solution of $\hat{t}_A$, we optimize $\hat{R}_A$ and $\hat{s}_A$ by

$$\begin{aligned}\hat{R}_A, \hat{s}_A &= \underset{R_A, s_A}{\operatorname{argmin}} \sum_{i=1}^{n} ||v'_i - s_A R_A p'_i||^2 \\ &= \underset{R_A, s_A}{\operatorname{argmin}} \sum_{i=1}^{n} ||v'_i||^2 - 2 s_A \sum_{i=1}^{n} (R_A p'_i \cdot v'_i) + s_A^2 \sum_{i=1}^{n} ||p'_i||^2 \\ &= \underset{R_A, s_A}{\operatorname{argmin}} \alpha s_A^2 - 2\beta s_A + \gamma, \qquad (13)\end{aligned}$$

$$\text{where } \alpha = \sum_{i=1}^{n} ||p'_i||^2, \quad \beta = \sum_{i=1}^{n} (R_A p'_i \cdot v'_i), \quad \gamma = \sum_{i=1}^{n} ||v'_i||^2. \quad (14)$$

In Equation 13, we use the notation $(\cdot)$ for the inner product, and α, β, and γ are scalar values. In practice, $\alpha > 0$ (*i.e.* for some data i, $||p'_i|| > 0$), because $\alpha = 0$ happens only when the user perfectly points the same location in all repetitions, which is nearly impossible. The optimal $\hat{s}_A$ is obtained when we set the partial derivative of Equation 13 with respect to s_A to be zero:

$$\hat{s}_A = \frac{\beta}{\alpha} = \frac{\sum_{i=1}^{n} (\hat{R}_A p'_i \cdot v'_i)}{\sum_{i=1}^{n} ||p'_i||^2}. \qquad (15)$$

Although $\hat{s}_A$ of Equation 15 is the exact solution of s_A for Equation 8, as Horn [7] suggests, we use an approximation of Equation 15 that is symmetric between p_i and v_i as follows:

$$\hat{s}_A \approx \sqrt{\frac{\sum_{i=1}^{n} ||v'_i||^2}{\sum_{i=1}^{n} ||p'_i||^2}}. \qquad (16)$$

This approximation is applied to alleviate the instability of the solution by Equation 15, which often generates too small $\hat{s}_A$. Substituting Equation 16 into Equation 13 reduces to

$$\hat{R}_A = \underset{R_A}{\operatorname{argmin}} \alpha \hat{s}_A^2 - 2\beta \hat{s}_A + \gamma = \underset{R_A}{\operatorname{argmax}} \beta, \qquad (17)$$

because $\alpha \geq 0, \gamma \geq 0, \hat{s}_A \geq 0$ and $\hat{s}_A$ is independent from R_A. To solve Equation 17, we represent the rotation R_A with a quaternion q_A that satisfies $R_A p'_i = q_A p'_i q_A^*$. Therefore, from Equation 14

$$\beta = \sum_{i=1}^{n} (q_A p'_i q_A^*) \cdot v'_i. \qquad (18)$$

We introduce three key properties of quaternions necessary for solving the new form of β [7]. The proofs can be found in Appendix A. For a quaternion $p = p_w + ip_x + jp_y + kp_z$, and its vector representation $[p_w \; p_x \; p_y \; p_z]^T$, we define the left and right matrices of the quaternion, P and $\bar{P}$:

$$P = \begin{bmatrix} p_w & -p_x & -p_y & -p_z \\ p_x & p_w & -p_z & p_y \\ p_y & p_z & p_w & -p_x \\ p_z & -p_y & p_x & p_w \end{bmatrix}, \bar{P} = \begin{bmatrix} p_w & -p_x & -p_y & -p_z \\ p_x & p_w & p_z & -p_y \\ p_y & -p_z & p_w & p_x \\ p_z & p_y & -p_x & p_w \end{bmatrix}.$$

We use the notation $(*)$ for conjugations of quaternions (*e.g.* $p^* = p_w - ip_x - jp_y - kp_z$), and $(\cdot)$ for the dot product between quaternions (*e.g.* $p \cdot q = p_w q_w + p_x q_x + p_y q_y + p_z q_z$).

Property 1 *Let p, q, and r be quaternions. Then,*

$$(pq) \cdot r = p \cdot (rq^*).$$

Property 2 *Let p and q be quaternions. If we define the left and right matrices of the quaternion p by P and $\bar{P}$, then*

$$pq = Pq, \quad qp = \bar{P}q.$$

Property 3 *Let q be a vector corresponding to a unit quaternion and N be a 4×4 matrix. When λ is the largest eigenvalue of N and ν is the eigenvector corresponding to λ, then*

$$\forall q, \; q^T N q \leq \nu^T N \nu.$$

With Property 1 and Equation 18, we have

$$\beta = \sum_{i=1}^{n} (q_A p'_i) \cdot (v'_i q_A). \qquad (19)$$

Applying Property 2 to Equation 19, we derive another form of β:

$$\beta = \sum_{i=1}^{n} q_A^T \bar{P}'^T_i V'_i q_A = q_A^T (\sum_{i=1}^{n} \bar{P}'^T_i V'_i) q_A = q_A^T N q_A, \qquad (20)$$

where $\bar{P}'_i$ is the right matrix of p'_i, V'_i is the left matrix of v'_i, and $N = \sum_{i=1}^{n} \bar{P}'^T_i V'_i \in \mathbb{R}^{4\times4}$. The components of N is

$$\begin{bmatrix} m_{11}+m_{22}+m_{33} & m_{23}-m_{32} & m_{31}-m_{13} & m_{12}-m_{21} \\ m_{23}-m_{32} & m_{11}-m_{22}-m_{33} & m_{12}+m_{21} & m_{31}+m_{13} \\ m_{31}-m_{13} & m_{12}+m_{21} & -m_{11}+m_{22}-m_{33} & m_{23}+m_{32} \\ m_{12}-m_{21} & m_{31}+m_{13} & m_{23}+m_{32} & -m_{11}-m_{22}+m_{33} \end{bmatrix},$$

where m_{ab} is the (a,b)-th element of matrix $M = \sum_{i=1}^{n} p'_i v'^T_i$. Note that $M \in \mathbb{R}^{3\times3}$ because $p'_i, v'_i \in \mathbb{R}^{3\times1}$. From Equation 17 and 20,

$$\hat{R}_A = \underset{R_A}{\operatorname{argmax}} \, q_A^T N q_A.$$

By Property 3, $\hat{R}_A$ corresponds to the rotation composed by the elements of quaternion $\hat{q}_A = [\hat{q}_w \; \hat{q}_x \; \hat{q}_y \; \hat{q}_z]^T$, which is the eigenvector for the maximum eigenvalue of N. Using the conversion formula, $\hat{R}_A$ is obtained from $\hat{q}_A$:

$$\hat{R}_A = \begin{bmatrix} 1-2\hat{q}_y^2-2\hat{q}_z^2 & 2(\hat{q}_x\hat{q}_y-\hat{q}_w\hat{q}_z) & 2(\hat{q}_x\hat{q}_z+\hat{q}_w\hat{q}_y) \\ 2(\hat{q}_x\hat{q}_y+\hat{q}_w\hat{q}_z) & 1-2\hat{q}_x^2-2\hat{q}_z^2 & 2(\hat{q}_y\hat{q}_z-\hat{q}_w\hat{q}_x) \\ 2(\hat{q}_x\hat{q}_z-\hat{q}_w\hat{q}_y) & 2(\hat{q}_y\hat{q}_z+\hat{q}_w\hat{q}_x) & 1-2\hat{q}_x^2-2\hat{q}_y^2 \end{bmatrix}, \qquad (21)$$

where $\hat{q}_A = \hat{q}_w + i\hat{q}_x + j\hat{q}_y + k\hat{q}_z$. In summary, we estimate A by plugging $\hat{t}_A$ of Equation 12, $\hat{R}_A$ of Equation 21, and $\hat{s}_A$ of Equation 16 into

$$\hat{A} = \begin{bmatrix} \hat{s}_A I_3 & 0_{3,1} \\ 0_{1,3} & 1 \end{bmatrix} \begin{bmatrix} I_3 & \hat{t}_A \\ 0_{1,3} & 1 \end{bmatrix} \begin{bmatrix} \hat{R}_A & 0_{3,1} \\ 0_{1,3} & 1 \end{bmatrix}.$$

4.2 Simplified Calibration

The simplified calibration is different from the full calibration that it assumes R_A is known. Unless the depth-camera moves on the HMD, the simplified calibration is enough for a new user. Its main benefit is that we can reduce the degree of freedom, which decreases the user's input repetition. We let the known fixed R_A denoted by R_F, and then Equation 8 changes to

$$\hat{t}_A, \hat{s}_A = \underset{t_A, s_A}{\operatorname{argmin}} \sum_i ||v_i - s_A(R_F p_i + t_A)||^2. \qquad (22)$$

Method	Known	Unknown	DOF	Min # data
F	–	t_A, R_A, s_A	7 (= 3 + 3 + 1)	3
S	R_A	t_A, s_A	4 (= 3 + 1)	2

Table 1: Comparison between full (F) and simplified (S) calibration. From left to right, each column indicates known and unknown parameters, degrees of freedom, and minimum number of calibration data.

Algorithm 1: OST-HMD Calibration Algorithm

```
input  : (1) Number of repetition: $n$. (2) (For simplified
         calibration only) the rotation matrix $R_A$
output: Estimated transformation : $\hat{A}$
1: $\{p_i\} \leftarrow \{\}$; $\{v_i\} \leftarrow \{\}$;
while $|\{p_i\}| < n$ do
    2: Render a virtual circle on the displays and wait for the
    user to point the virtual circle;
    3: $\{p_i\} \leftarrow \{p_i\} +$ position of the fingertip by
    depth-camera;
    4: $\{v_i\} \leftarrow \{v_i\} +$ position of the virtual circle;
/* Full calibration */
5: Compute $\hat{t}_A$, $\hat{s}_A$, $\hat{R}_A$ by Equation 12, 16, 21, respectively;
6: Construct $\hat{A}$ by Equation 2;
/* Simplified calibration */
5: Compute $\hat{t}_A$, $\hat{s}_A$ by Equation 23 and 24, respectively;
6: Construct $\hat{A}$ by Equation 25;
```

The derivation of the key parameters is the same with that of full calibration in section 4.1; only difference is to substituting R_F into $\hat{R}_A$. As a result, we have the scaling and translation parameters from Equation 12 and 16, respectively.

$$\hat{t}_A = \frac{\bar{v}}{\hat{s}_A} - R_F \bar{p}, \tag{23}$$

$$\hat{s}_A \approx \sqrt{\frac{\sum_{i=1}^{n} ||v'_i||^2}{\sum_{i=1}^{n} ||p'_i||^2}}. \tag{24}$$

Finally, we estimate A and calibrate the OST-HMD:

$$\hat{A} = \begin{bmatrix} \hat{s}_A I_3 & 0_{3,1} \\ 0_{1,3} & 1 \end{bmatrix} \begin{bmatrix} I_3 & \hat{t}_A \\ 0_{1,3} & 1 \end{bmatrix} \begin{bmatrix} R_F & 0_{3,1} \\ 0_{1,3} & 1 \end{bmatrix}. \tag{25}$$

4.3 Analysis of the Calibration Method

Table 1 summarizes the comparative analysis between full and simplified calibration, including known and unknown parameters, degrees of freedom, and minimum number of calibration data. The degrees of freedom of translation (t_A), rotation (R_A), and scaling (s_A) are 3, 3, and 1, respectively. Since each user interaction produces 3 independent parameters, we need 3 calibration points for the full calibration at minimum, while 2 calibration points for the simplified calibration. However, for accurate estimation, our empirical results recommend about 10 and 5 calibration points for full and simplified calibration, respectively. Finally, Algorithm 1 summarizes the whole procedure of our OST-HMD calibration.

5 Experiments

We empirically show that the proposed calibration is not only more accurate but also easier and faster for a general user than existing methods such as stereo-SPAAM [4]. In section 5.2, we compare between the calibration methods in terms of accuracies and the number of users' input repetitions. In section 5.3, we report qualitative results.

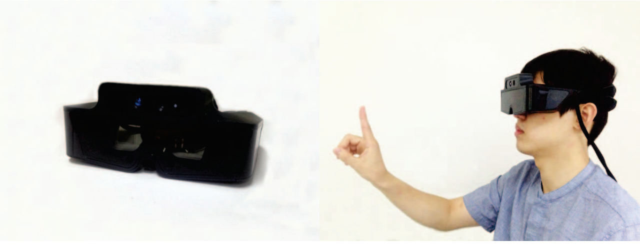


Figure 3: The Meta 1 system (left) and an actual calibration procedure by a user's repetitive pointing (right).

5.1 Experimental Setup

For evaluation, we use Meta 1[1] in Figure 3, which is an affordable OST-HMD equipped with 3D see through displays and a 3D time-of-flight depth-camera. The resolution of each display is 960 × 540. We use Unity3D[2], OpenCV [2], iisu[3], and ALGLIB[4] to implement the proposed calibration methods.

Since the quality of calibration can be solely measured by users who wear the HMD, we design our quantitative evaluation based on the user feedback. We recruit 20 people who are not familiar to OST-HMDs, and after careful instruction we ask each of them to point a randomly positioned virtual circle 50 times. We use $\{p_i\}$ to denote the positions of a single user's pointing fingertips from the camera coordinate system, and $\{v_i\}$ to denote the positions of the virtual circles from the eye coordinate system. In order to obtain a stable $\{p_i\}$, we use the average of the depth measurements of a fingertip from 20 consecutive frames. We measure the position errors $\{e_i\}$ as the difference from corresponding pairs of $\{p_i\}$ and $\{v_i\}$:

$$e_i = v_i - Ap_i. \tag{26}$$

The sources of errors in our measurement are two-fold: *calibration* errors and *non-calibration* errors. The calibration errors are originated from inaccurate estimation of A, whereas the non-calibration indicates the other remaining errors, which mainly come from humans' incorrect pointing to the virtual circles. That is, even with a perfectly calibrated HMD, a human may not be always able to correctly point the circles. Therefore, the calibration error is minimized and fixed after the calibration, and the non-calibration error can always occur whenever a user points a virtual circle, which is modeled to follow a Gaussian distribution.

For a position error e_i, we denote the calibration error by $\tilde{e}$ and the non-calibration error by e'_i. With the law of large numbers, if the size of $\{e_i\}$ denoted by n is large enough, the mean of the non-calibration errors approximately becomes zero. Therefore, the calibration error is

$$\frac{1}{n}\sum_{i=1}^{n} e'_i \approx 0 \Rightarrow \tilde{e} = \frac{1}{n}\sum_{i=1}^{n} \tilde{e}_i = \frac{1}{n}\sum_{i=1}^{n} (e_i - e'_i) \approx \frac{1}{n}\sum_{i=1}^{n} e_i. \tag{27}$$

5.2 Quantitative Results

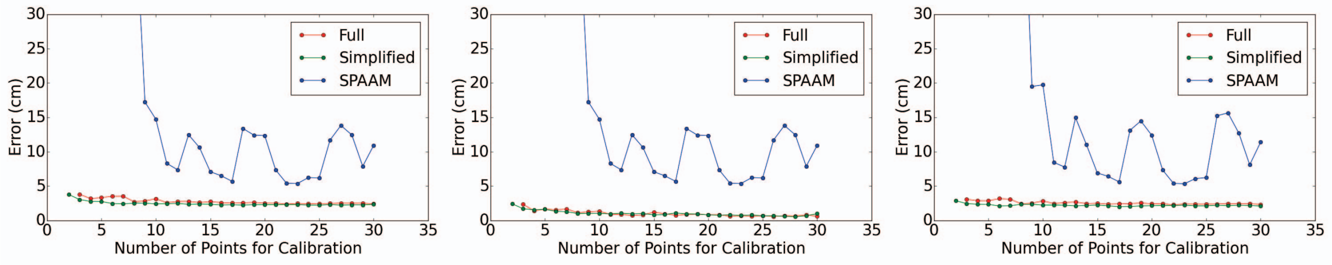
We compare our full and simplified calibration with the stereo-SPAAM, one of the most popular calibration techniques [4]. In our experiments, the original stereo-SPAAM often suffers from too large errors (*e.g.* even several meters), especially when novice users perform calibration. Hence, one main update for the stereo-SPAAM is that we fix the z value of the estimated depth-camera position

[1] https://www.getameta.com/

[2] https://unity3d.com/

[3] http://www.softkinetic.com/Products/iisuMiddleware

[4] http://www.alglib.net/



(a) Total position errors (MAE$_p$)

(b) Calibration errors (MAE$_c$)

(c) Non-calibration errors, (MAE$_n$)

Figure 4: Analysis of position errors of full calibration, simplified calibration and stereo-SPAAM. We show the results of total position errors in (a), calibration errors in (b), and non-calibration errors in (c).

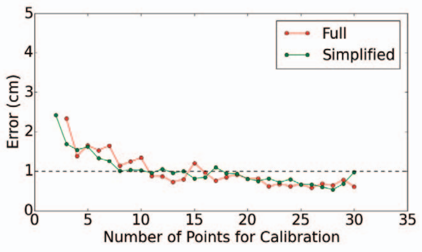


Figure 5: Variation of calibration errors (MAE$_c$) according to the number of calibration points.

to zero; otherwise the calibration accuracy of the stereo-SPAAM severely fluctuates. Note that fixing the z value is favorable for the stereo-SPAMM by excluding one dimension of errors (*i.e.* providing the answer for one dimension). In next section, we will show that our method does not take advantage of this fixation, but still significantly outperforms the stereo-SPAMM. Appendix B presents the details of the modification and how to obtain the position errors as a performance measure.

We use the 3D position errors as the evaluation metric, instead of 2D projection errors that are measured from the left and right displays. The main reason is that the results with a small 3D position error guarantee a more robust calibration. The OST-HMD with a small 2D projection error can be vulnerable to even a small amount of rotational disturbance, for example, if a user slightly rotates her viewing direction, a small 2D projection error can change to a very large one, whereas that that of 3D position error does not.

We let $\{p_i^j\}$ and $\{v_i^j\}$ to denote a set of positions of fingertips and virtual circles by user j. We also use n and m for the size of each point set and the number of users. We use $n = 50$ and $m = 20$ in our experiments. We measure the mean absolute errors (MAE) as the metric of the position errors e_i^j, calibration errors E^j, and non-calibration errors ε_i^j as follows:

$$\mathrm{MAE}_p = \frac{1}{m}\sum_j \frac{1}{n}\sum_i ||e_i^j||. \tag{28}$$

$$\mathrm{MAE}_c = \frac{1}{m}\sum_j ||\frac{1}{n}\sum_i e_i^j|| \approx \frac{1}{m}\sum_j ||E^j||. \tag{29}$$

$$\mathrm{MAE}_n = \frac{1}{m}\sum_j\sum_i (||e_i^j - \frac{1}{n}\sum_i e_i^j||) \approx \frac{1}{m}\sum_j\sum_i ||\varepsilon_i^j||. \tag{30}$$

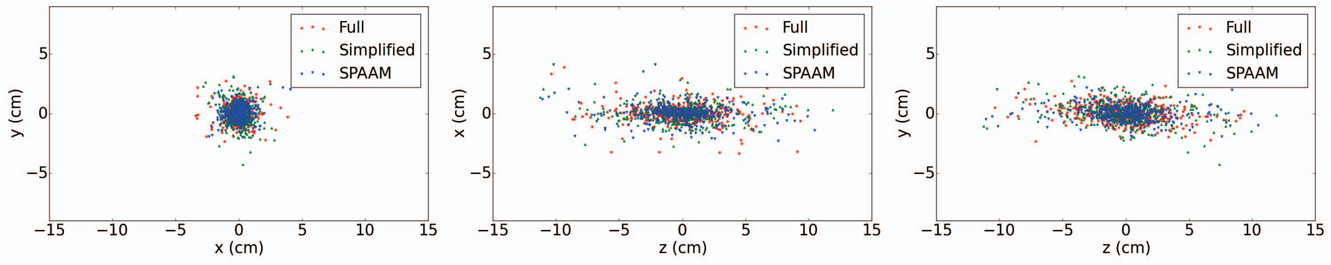
We show the MAEs of the position errors, calibration errors, and non-calibration errors in Figure 4(a)–(c), respectively. The MAE$_c$ decreases as the number of repetition increases, but the MAE$_p$ converges to 2 cm instead of the ideal zero. This bias is due to the non-calibration errors (*i.e.* MAE$_n$), which cannot be completely removed by calibration and depends on the users' skills. It is also supported by Figure 4(c) in which the non-calibration errors do not decrease with more number of points for calibration (*i.e.* as n increases, $\mathrm{MAE}_c \to 0 \Rightarrow \mathrm{MAE}_p = \mathrm{MAE}_n$). In the results of stereo-SPAAM, the magnitude of MAE$_c$ is not linear with the number of calibration points due to the lack of robustness. As shown in Figure 4, they include many spurious outliers, which are the points with large calibration errors.

Since more calibration points cost a user to spend more time for calibration, it is important to find an appropriate number of calibration points, which we denote by n^*. As shown in Figure 5, the magnitude of MAE$_c$ can be a criteria for finding the appropriate number of calibration points since it indicates how accurately the calibration is done. For an error threshold of 1 cm, 11 points are needed for the full calibration and 8 points are needed for the simplified calibration. This is due to the difference in the degree of freedom of full and simplified calibration, as in Table 1. However, since the error quickly drops with a few initial points, fewer points may be acceptable (*e.g.* 10 and 5 for full and simplified calibration).

Figure 6 shows the distribution of the non-calibration errors in the three planes. We plot them using the minimum number of calibration points for each calibration method, as recommended in the quantitative results of previous section 5.2: 11 (full), 8 (simplified), and 20 (stereo-SPAAM) points. We observe that the shapes of the distributions are similar between calibration methods. It indicates that the non-calibration errors of Equation 27 are independent to the calibration methods. In addition, Figure 6 also validates our assumption that the non-calibration errors follow a Gaussian distribution, and thus the law of large numbers is applicable to Equation 27. The variance along the z axis is wider than the others, while the variances of x and y axes are similar. The wider distribution of the z axis is natural since the depth perception of OST-HMDs originates from horizontal disparity, whose scale is related to the IPD (*e.g.* 63 mm), while the scale of z is related to the arm length (*e.g.* 50 cm). Such scale discrepancy makes the z-axis errors larger than x and y-axis errors.

5.3 Qualitative Results

We present qualitative results of how the calibration influences the rendering of virtual objects. Figure 7 shows a real hand and a cup and their rendered images in the displays after calibration. Since the results of full calibration and simplified calibration are almost the same, we compare only between the full calibration and the stereo-SPAAM. When each of the real objects in Figure 7(a) is in front of an OST-HMD, its virtual images that appear on the displays are Figure 7(b)–(c) for the full calibration and Figure 7(d)–(e) for the stereo-SPAAM. As shown in Figure 7(b)–(c), the full calibration leads that the shapes of both images are almost equal except that

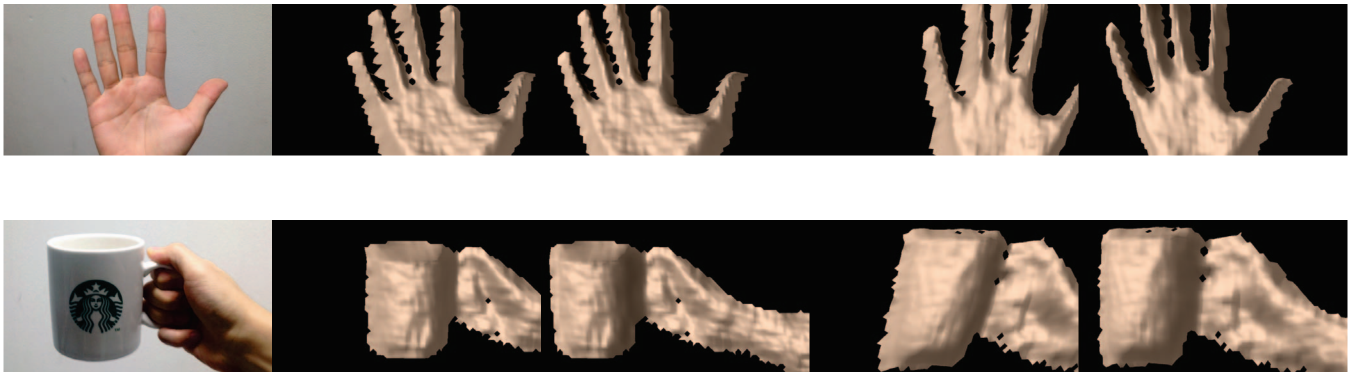


(a) $x-y$ plane

(b) $z-x$ plane

(c) $z-y$ plane

Figure 6: Non-calibration error distributions of full, simplified calibration, and stereo-SPAAM. The users' view directions are on the z-axis.



(a) Real

(b) Left - Full Calibration

(c) Right - Full Calibration

(d) Left - SPAAM

(e) Right - SPAAM

Figure 7: A real hand and a cup and their virtual images rendered on the left and right displays after calibration. We show the results of our calibration in (b)–(c) and the stereo-SPAAM in (d)–(e).

Method	Error	# repetition	Device	Aligned
Ours	< 1 cm	$5 \sim 10$ times	Depth camera	O
SPAAM	> 5 cm	≥ 20 times	6-DOF Tracker	X

Table 2: Comparison between our method and stereo-SPAAM. From left to right, each column indicates a range of calibration errors, the number of interaction repetitions, a device required for calibration, and whether left and right displays are well aligned to each other.

the hand in the left image is positioned rightward than the one on the right. This difference is originated from the horizontal disparity that causes the depth perception. Since the shapes in the images are equal, users can match clearly them to a real object. On the other hand, in the images of the stereo-SPAAM in Figure 7(d)–(e), the hand on the left is rotated compared to that on the right. Such discrepancy is likely to disturb the users from overlapping them onto a real object. This is due to an inherent limitation of stereo-SPAAM, which is originally developed for monocular OST-HMDs. The stereo-SPAAM, which is based on SPAAM, focuses on the correct projection to the 2D displays but lack a mechanism that keeps the shapes shown the same in both displays.

Finally, Table 2 summarizes the comparison between our method and the stereo-SPAAM.

6 Conclusion

We proposed an accurate, time-efficient, and user-friendly calibration method for OST-HMDs leveraging an affordable depth-camera. As many low-cost depth-cameras are recently available, even some OST-HMDs are equipped with built-in depth-cameras (*e.g.* Meta 1). We designed two stages of full and simplified calibration based on a practical HMD usage scenario. We empirically showed that the proposed method is more accurate than the current state-of-art methods while requiring less calibration points from users. Moreover, we solved some remaining issues discussed in the stereo-SPAMM [4], which include removing vertical disparity, and not requiring special physical targets for calibration. Novice users could quickly carry out calibration using their hands without any additional devices. One interesting immediate future direction is that we can directly collect the calibration data from AR user interactions. Since we proposed to use users' hands for obtaining calibration data, our method can be easily extended to perform calibration as soon as users start their desired AR tasks without demanding any extra calibration procedure.

References

[1] R. Azuma and G. Bishop. Improving Static and Dynamic Registration in an Optical See-through HMD. *Computer Graphics*, pages 194–204, 1994.

[2] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[3] N. A. Dodgson. Variation and Extrema of Human Interpupillary Distance. In *Proceedings of SPIE: Stereoscopic Displays and Virtual Reality Systems XI*, volume 5291, pages 36–46, May 2004.

[4] Y. Genc, F. Sauer, F. Wenzel, M. Tuceryan, and N. Navab. Optical see-through HMD calibration: A Stereo Method Validated with a Video See-Through System. In *Proceedings of the IEEE and ACM International Symposium on Augmented Reality*, pages 165–174, 2000.

[5] Y. Genc, M. Tuceryan, and N. Navab. Practical Solutions for Calibration of Optical See-through Devices. In *Proceedings of the IEEE In-*

ternational Symposium on Mixed and Augmented Reality, pages 169–175, 2002.

[6] R. L. Holloway. Registration Error Analysis for Augmented Reality. *Presence: Teleoperators and Virtual Environments*, 6(4):413–432, 1997.

[7] B. K. P. Horn. Closed-form Solution of Absolute Orientation using Unit Quaternions. *Journal of the Optical Society of America A*, 4(4):629–642, 1987.

[8] Y. Itoh and G. Klinker. Interaction-Free Calibration for Optical See-Through Head-Mounted Displays based on 3D Eye Localization. In *Proceedings of the IEEE Symposium on 3D User Interfaces*, pages 75–82, 2014.

[9] A. L. Janin, D. W. Mizell, and T. P. Caudell. Calibration of Head-Mounted Displays for Augmented Reality Applications. In *Proceedings of the IEEE Virtual Reality Annual International Symposium*, pages 246–255, 1993.

[10] F. Kellner, B. Bolte, G. Bruder, U. Rautenberg, F. Steinicke, M. Lappe, and R. Koch. Geometric Calibration of Head-Mounted Displays and tis Effects on Distance Estimation. *IEEE Transactions on Visualization and Computer Graphics*, 18(4):589–596, Apr. 2012.

[11] E. McGarrity and M. Tuceryan. A Method for Calibrating See-through Head-mounted Displays for AR. In *Proceedings of the IEEE and ACM International Workshop on Augmented Reality*, pages 75–84, 1999.

[12] C. B. Owen, J. Zhou, A. Tang, and F. Xiao. Display-Relative Calibration for Optical See-Through Head-Mounted Displays. In *Proceedings of the IEEE International Symposium on Mixed and Augmented Reality*, pages 70–78, 2004.

[13] A. Plopski, Y. Itoh, C. Nitschke, K. Kiyokawa, G. Klinker, and H. Takemura. Corneal-Imaging Calibration for Optical See-Through Head-Mounted Displays. *IEEE Transactions on Visualization and Computer Graphics*, 21(4):481–490, 2015.

[14] K. Ponto, M. Gleicher, R. G. Radwin, and H. J. Shin. Perceptual Calibration for Immersive Display Environments. *IEEE Transactions on Visualization and Computer Graphics*, 19(4):691–700, 2013.

[15] N. Qian. Binocular Disparity and the Perception of Depth. *Neuron*, 18:359–368, 1997.

[16] R. S. Renner, B. M. Velichkovsky, J. R. Helmert, and R. H. Stelzer. Measuring interpupillary distance might not be enough. In *Proceedings of the ACM Symposium on Applied Perception*, page 130, 2013.

[17] A. Tang, J. Zhou, and C. Owen. Evaluation of Calibration Procedures for Optical See-through Head-Mounted Displays. In *Proceedings of the IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 161–168, 2003.

[18] M. Tuceryan and N. Navab. Single Point Active Alignment Method (SPAAM) for Optical See-through HMD Calibration for AR. In *Proceedings of the IEEE and ACM International Symposium on Augmented Reality*, pages 149–158, 2000.

[19] C. Zhang and Z. Zhang. Calibration between Depth and Color Sensors for Commodity Depth Cameras. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, pages 1–6, 2011.

A Proofs of Properties of Quaternions

A.1 Property 1

Property 1 can be proved by direct calculation. We let $p = p_w + ip_x + jp_y + kp_z$, $q = q_w + iq_x + jq_y + kq_z$, and $r = r_w + ir_x + jr_y + kr_z$. Through multiplications and inner products, the left and right terms become

$$\begin{aligned}(pq)\cdot r = (&p_wq_w - p_xq_x - p_yq_y - p_zq_z\\&+ i(p_wq_x + p_xq_w + p_yq_z - p_zq_y)\\&+ j(p_wq_y - p_xq_z + p_yq_w + p_zq_x)\\&+ k(p_wq_z + p_xq_y - p_yq_x + p_zq_w))\cdot r\\=\ &r_w(p_wq_w - p_xq_x - p_yq_y - p_zq_z)\\&+ r_x(p_wq_x + p_xq_w + p_yq_z - p_zq_y)\\&+ r_y(p_wq_y - p_xq_z + p_yq_w + p_zq_x)\\&+ r_z(p_wq_z + p_xq_y - p_yq_x + p_zq_w),\end{aligned} \tag{31}$$

$$\begin{aligned}p\cdot(rq^*) = p\cdot(&r_wq_w + r_xq_x + r_yq_y + r_zq_z\\&+ i(-r_wq_x + r_xq_w - r_yq_z + r_zq_y)\\&+ j(-r_wq_y + r_xq_z + r_yq_w - r_zq_x)\\&+ k(-r_wq_z - r_xq_y + r_yq_x + r_zq_w))\\=\ &p_w(r_wq_w + r_xq_x + r_yq_y + r_zq_z)\\&+ p_x(-r_wq_x + r_xq_w - r_yq_z + r_zq_y)\\&+ p_y(-r_wq_y + r_xq_z + r_yq_w - r_zq_x)\\&+ p_z(-r_wq_z - r_xq_y + r_yq_x + r_zq_w).\end{aligned} \tag{32}$$

It is straightforward to see that Equation 31 and 32 are identical.

A.2 Property 2

Property 2 is also proved by direct calculation. We represent the quaternion p and q in the vector form of $p = [p_w\ p_x\ p_y\ p_z]^T$ and $q = [q_w\ q_x\ q_y\ q_z]^T$. We then obtain the vector form of pq and qp:

$$pq = \begin{bmatrix}p_wq_w - p_xq_x - p_yq_y - p_zq_z\\p_wq_x + p_xq_w + p_yq_z - p_zq_y\\p_wq_y - p_xq_z + p_yq_w + p_zq_x\\p_wq_z + p_xq_y - p_yq_x + p_zq_w\end{bmatrix}, \tag{33}$$

$$qp = \begin{bmatrix}q_wp_w - q_xp_x - q_yp_y - q_zp_z\\q_wp_x + q_xp_w + q_yp_z - q_zp_y\\q_wp_y - q_xp_z + q_yp_w + q_zp_x\\q_wp_z + q_xp_y - q_yp_x + q_zp_w\end{bmatrix}. \tag{34}$$

We calculate Pq and $\bar{P}q$:

$$Pq = \begin{bmatrix}p_w & -p_x & -p_y & -p_z\\p_x & p_w & -p_z & p_y\\p_y & p_z & p_w & -p_x\\p_z & -p_y & p_x & p_w\end{bmatrix}\begin{bmatrix}q_w\\q_x\\q_y\\q_z\end{bmatrix} = \begin{bmatrix}p_wq_w - p_xq_x - p_yq_y - p_zq_z\\p_wq_x + p_xq_w + p_yq_z - p_zq_y\\p_wq_y - p_xq_z + p_yq_w + p_zq_x\\p_wq_z + p_xq_y - p_yq_x + p_zq_w\end{bmatrix}, \tag{35}$$

$$\bar{P}q = \begin{bmatrix}p_w & -p_x & -p_y & -p_z\\p_x & p_w & p_z & -p_y\\p_y & -p_z & p_w & p_x\\p_z & p_y & -p_x & p_w\end{bmatrix}\begin{bmatrix}q_w\\q_x\\q_y\\q_z\end{bmatrix} = \begin{bmatrix}q_wp_w - q_xp_x - q_yp_y - q_zp_z\\q_wp_x + q_xp_w + q_yp_z - q_zp_y\\q_wp_y - q_xp_z + q_yp_w + q_zp_x\\q_wp_z + q_xp_y - q_yp_x + q_zp_w\end{bmatrix}. \tag{36}$$

We can show $pq = Pq$ from the equality of Equation 33 and 35, and $qp = \bar{P}q$ from the equality of Equation 34 and 36.

A.3 Property 3

Let N to be a 4×4 matrix. By eigen-decomposition, we can find four unit eigenvectors (v_1, v_2, v_3, and v_4) and their corresponding eigenvalues (λ_1, λ_2, λ_3, and λ_4) of N satisfying

$$Nv_i = \lambda v_i \quad \text{for } i = 1,2,3,4, \quad \text{where } \lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \lambda_4.$$

For a unit vector $q \in \mathbb{R}^{4\times 1}$,

$$q = \sigma_1 v_1 + \sigma_2 v_2 + \sigma_3 v_3 + \sigma_4 v_4.$$

Since q is a unit vector, $||q||^2 = \sum_{i=1}^{4} \sigma_i^2 = 1$. Finally, we can derive

$$q^T Nq = \sigma_1^2\lambda_1 + \sigma_2^2\lambda_2 + \sigma_3^2\lambda_3 + \sigma_4^2\lambda_4 \leq \lambda_1.$$

When $\sigma_1 = 1$ ($q = v_1$), $q^T Nq = \lambda_1$. Thus, we prove that $q^T Nq$ is maximized when q is equal to the eigenvector that corresponds to the maximum eigenvalue of N.

B The Stereo-SPAAM

The SPAAM is an OST-HMD calibration method that is first invented for monocular OST-HMDs [18], then extended to the stereo-SPAAM for binocular OST-HMDs [4]. However, the stereo-SPAAM handles a pair of monocular OST-HMDs independently, and thus it lacks a mechanism that keeps the shapes shown consistently in both displays, as described in the results of Figure 7

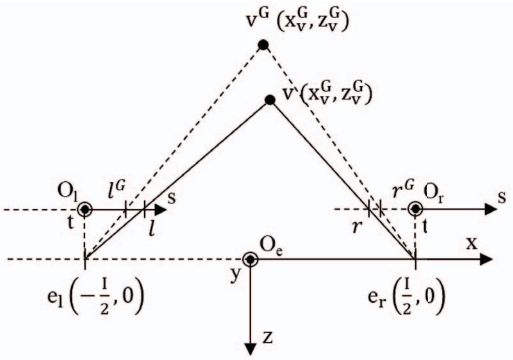


Figure 8: Illustration of left and right display coordinate system (O_l and O_r as centers), and the eye coordinate system (O_e) for the derivation of projection errors of the stereo-SPAAM. The left and right display points of v_i and v_i^G on the xz-plane are denoted by (l,r), and (l^G, r^G). e_l and e_r are the centers of left and right eyes with an IPD I.

(d)–(e). Since the shapes in the displays are not correspondent, it is difficult to obtain 3D coordinates from the two display images. Thus we project the 3D world onto the 2D xz-plane of the eye coordinate system to find the 3D positions from the results of the stereo-SPAAM.

We start from the relation between the points on the displays of the OST-HMDs (*i.e.* display points) and their corresponding 3D positions that originate from the model of our calibration method. We let $\{l_i\}$ and $\{r_i\}$ to denote the display points for $\{p_i\}$, and P_L and P_R to denote the projection matrices of the left and right displays[5]. After the full calibration with the correctly measured P_L, P_R, and A, we obtain the relation between $\{p_i\}$ and their display points:

$$l_i = P_L A p_i, \quad r_i = P_R A p_i. \tag{37}$$

We now formulate the positions of display points of the left and right displays $\{l_i^G\}$ and $\{r_i^G\}$ by the stereo-SPAAM. Note that they are different from the above $\{l_i\}$ and $\{r_i\}$, which are display points by our calibration method. We let G_L and G_R to denote the 3×4 camera matrices, which are the transformations from a point in the camera coordinate system to the left and right displays. By the definition of G_L and G_R, we have the display points for $\{p_i\}$:

$$l_i^G = G_L p_i, \quad r_i^G = G_R p_i. \tag{38}$$

Before further explanation, we introduce the element-wise notation of the points as follows:

$$\begin{aligned} l_i = [s_l\ t_l]^T, \quad r_i = [s_r\ t_r]^T, \quad v_i = [x_v\ y_v\ z_v]^T, \\ l_i^G = [s_l^G\ t_l^G]^T, \quad r_i^G = [s_r^G\ t_r^G]^T, \quad v_i^G = [x_v^G\ y_v^G\ z_v^G]^T. \end{aligned}$$

where s and t are indices in the display coordinate system and x, y, and z are those in the eye coordinate system. The s value is zero when the display point is exactly in front of the corresponding eye center. v_i and v_i^G are the points in the eye coordinate system that correspond to p_i and p_i^G in the depth-camera coordinate system as Equation 1. From now, we do not use the homogeneous coordinate system.

Figure 8 depicts the correspondence between the display points and the lines on the xz-plane of the eye coordinate system. Each of the display points of the OST-HMDs corresponds to a line penetrating the user's eye that is looking at the point. With an IPD I, the lines projected onto the xz-plane are:

$$\begin{aligned} \overline{ve_l} : z = ks_l\left(x + \frac{I}{2}\right), \quad \overline{v^G e_l} : z = ks_l^G\left(x + \frac{I}{2}\right), \\ \overline{ve_r} : z = ks_r\left(x - \frac{I}{2}\right), \quad \overline{v^G e_r} : z = ks_r^G\left(x - \frac{I}{2}\right), \end{aligned} \tag{39}$$

where e_l and e_r denote the left and right eye center positions and k is the conversion factor between the line slopes and the s values (See Figure 8 for better understanding).

In order to find v_i^G, we use the fact that $\overline{ve_l}$ and $\overline{ve_r}$ meet at v and $\overline{v^G e_l}$ and $\overline{v^G e_r}$ meet at v^G. Then we have

$$\frac{s_l^G}{s_l} = \frac{\frac{x_v^G + \frac{I}{2}}{z_v^G}}{\frac{x_v + \frac{I}{2}}{z_v}}, \quad \frac{s_r^G}{s_r} = \frac{\frac{x_v^G - \frac{I}{2}}{z_v^G}}{\frac{x_v - \frac{I}{2}}{z_v}}, \quad \frac{s_r}{s_l} = \frac{\frac{x_v - \frac{I}{2}}{z_v}}{\frac{x_v + \frac{I}{2}}{z_v}}. \tag{40}$$

From Equation 40, it is easy to see

$$x_v^G = \frac{(s_l - s_r)(s_l^G + s_r^G)}{(s_l^G - s_r^G)(s_l + s_r)} x_v, \quad z_v^G = \frac{s_l - s_r}{s_l^G - s_r^G} z_v. \tag{41}$$

By Equation 41, we can obtain the projected position error:

$$\underline{e}_i = \underline{v}_i^G - \underline{v}_i, \tag{42}$$

where $\underline{v}_i^G$ and $\underline{v}_i$ are v_i^G and v_i projected onto the xz-plane, respectively. Finally, we use $\underline{e}_i$ instead of e_i from Equation 26 to compute the errors of the stereo-SPAAM.

In our experiment, we do not use the original stereo-SPAAM [4], due to too large projected errors, which are sometimes larger than 10 m. This is because the error optimization of the stereo-SPAAM is performed in the homogeneous coordinate system, and thus wx and wy are used as the objectives of the optimization. The problem here is, $\frac{x}{w}$ and $\frac{y}{w}$ are the real targets of OST-HMD calibration, not wx and wy. The stereo-SPAAM works fine with small non-calibration errors and noises, but when the errors and noises are large, the stereo-SPAAM decreases w and increases $\frac{x}{w}$ and $\frac{y}{w}$, which leads to a failure in calibration. We avoid such calibration failures of the stereo-SPAAM by adding the following modification. When G_L and G_R are the camera matrices of the left and right displays, each of the matrices has a zero element in the 4th row and 4th column. It corresponds to the following statement of section 5.2: *we fix the z value of the estimated depth-camera position to zero*. This modification leads the stereo-SPAAM to become more robust to errors since it prevents w from being too small.

[5] P_L and P_R can be uniquely obtained from the field of views and center offsets of the displays. When the horizontal and vertical field of views are θ_u, θ_v, and the offsets of the left and right displays are (c_u, c_v), $(-c_u, c_v)$, we have

$$P_L = \begin{bmatrix} f_u & 0 & c_u & 0 \\ 0 & f_v & c_v & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad P_R = \begin{bmatrix} f_u & 0 & -c_u & 0 \\ 0 & f_v & c_v & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad \text{where } f_u = \frac{1}{\tan(\theta_u/2)}, \ f_v = \frac{1}{\tan(\theta_v/2)}.$$