IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 8 December 2023, accepted 22 December 2023, date of publication 28 December 2023,
date of current version 22 January 2024.

Digital Object Identifier 10.1109/ACCESS.2023.3347781

RESEARCH ARTICLE

Integrating Heterogeneous VR Systems Into Physical Space for Collaborative Extended Reality

JAEPUNG AN[1], **JOO HO LEE**[1], **SANGHUN PARK**[2], **AND INSUNG IHM**[1]
[1]Department of Computer Science and Engineering, Sogang University, Seoul 04107, Republic of Korea
[2]Graduate School of Metaverse, Sogang University, Seoul 04107, Republic of Korea

Corresponding author: Insung Ihm (ihm@sogang.ac.kr)

This work was supported by the National Research Foundation of Korea (NRF) Grant funded by the Korea Government [Ministry of Science and ICT (MSIT)] under Grant NRF-2020R1A2C2011709.

ABSTRACT Virtual reality (VR) applications are typically developed within their own immersive digital worlds; therefore, virtual spaces are usually treated as discrete from the physical space where augmented and mixed reality users exist, which makes it difficult to combine these heterogeneous realities into an integrated extended reality (XR) environment. Along these lines, we propose a method that enables a user to geometrically register the virtual space within a VR application to a real space using a commodity camera in a workspace as an anchor point. We first investigate the mathematical aspect of the computational model for connecting the virtual space to the physical world. Then, we present a computational procedure that implements our proposed method with numerical accuracy and stability. As an application, we demonstrate that users of two VR systems from different vendors may collaborate within a shared real workspace while interacting with each other physically. The presented method provides a key mechanism for enabling XR users to leverage these immersive technologies by effectively using different realities within an integrated environment.

INDEX TERMS Human-centered computing, human–computer interaction (HCI), interaction paradigms, virtual reality.

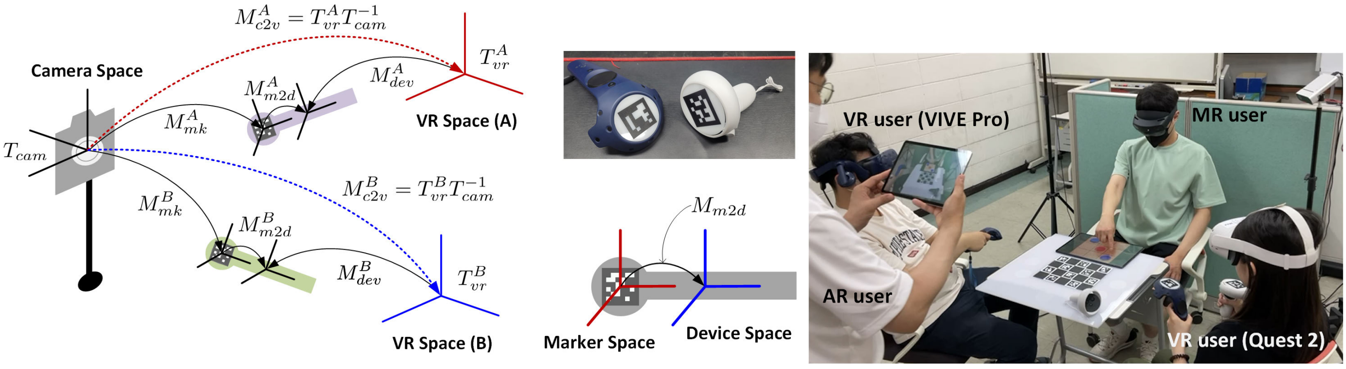
I. INTRODUCTION

A. BACKGROUND

Extended reality (XR) refers to immersive technologies that combine virtual reality (VR), augmented reality (AR), and mixed reality (MR) within a shared environment. In particular, XR aims to integrate the physical world with a virtual world so that users can effectively interact with each other by synergistically leveraging the strengths of each type of technology within a unified world. One of the critical requirements for building such an XR environment is to accurately match or align to each other the geometric space that is defined independently in each reality. Note that the AR and MR applications inherently involve physical spaces in the real world, while VR applications assume their own virtual space.

The world spaces used by the VR applications are usually regarded as separate from the physical space where users from heterogeneous realities collaborate with each other. Although some recent VR headsets provide a camera that enables to see a real-time view of surroundings, a user is often not allowed to access or manipulate the images or videos from the camera [1]. In addition, the use of camera in a VR application is often prohibited for a security reason. Therefore, it is of utmost importance to connect the virtual space employed by a VR application to the physical geometric space where AR and MR users already exist. While some recent attempts have considered this fundamental problem [2], [3], [4], [5], [6], [7], [8], an effective mechanism for physically integrating the VR space with the real space used by the AR and MR environments remains elusive. To this end, we investigate a mechanism for easily registering a VR system within physical spaces in this work.

The associate editor coordinating the review of this manuscript and approving it for publication was Giacinto Barresi.



(a) Interconnection of immersive VR spaces (b) Marker and device spaces (c) Tic-tac-toe game in an XR environment

FIGURE 1. Connecting immersive virtual worlds to physical space. (a) The immersive spaces of two heterogeneous virtual reality (VR) systems (i.e., *A* and *B*) are respectively connected geometrically to a physical space using a camera system as an anchor point. (b) To enable this connection, a marker is attached to an arbitrary position for each of the VR controllers, from which the marker-to-device transformation, M_{m2d} is defined for each VR system. Then, the unknown M_{m2d} is estimated by registering the controller with a marker in the camera system. As (a) implies, computing M_{m2d} is equivalent to computing the camera-to-VR transformation M_{c2v}. Once the estimates for M_{c2v} are obtained for the two VR systems, they become physically interconnected in a shared extended reality (XR) environment. The camera system easily enables both mixed reality (MR) and augmented reality (AR) users to connect to the camera space via marker tracking, which makes it possible for users from different realities to collaborate effectively within the integrated XR environment. (c) Schematic illustration of the presented idea where two heterogeneous VR users from the HTC VIVE Pro and the Meta Quest 2 systems, an MR user wearing a Microsoft HoloLens 2 headset, and an AR user holding a Samsung Galaxy Tab S6 tablet PC play a tic-tac-toe game in the physically integrated XR environment.

B. PROBLEM SPECIFICATION

Figure 1(a) illustrates an XR environment where two VR users with VR systems from different vendors (i.e., *A* and *B*) are supposed to interact with each other in the shared physical space. To connect the virtual worlds assumed by the respective VR systems to the physical space, we use a camera system whose six degrees of freedom (6-DoF) pose T_{cam} in the physical space defines the camera space (**camera space**). Note that the reference coordinate system for the physical space can be placed anywhere in the workspace. If we assume that the 6-DoF pose coincide with that of the camera system, it becomes $T_{cam} = I$.

On the other hand, the two VR systems independently set up their own virtual spaces (i.e., **VR spaces A** and **B**), which actually exist somewhere in the physical space. Let T^A_{vr} and T^B_{vr} be the (unknown) poses of the two VR spaces in the physical space. Then, the problem we aim to solve in this work is as follows: *Determine the unknown transformations $M^A_{c2v} = T^A_{vr}T^{-1}_{cam}$ and $M^B_{c2v} = T^B_{vr}T^{-1}_{cam}$, which respectively interconnect the corresponding VR spaces (i.e.,* A *and* B*) to the camera system in the physical space.* Once these *camera-to-VR transformations* are known, the two VR systems become geometrically coupled in the physically shared workspace via the following transformations $T_{A2B} = M^B_{c2v}(M^A_{c2v})^{-1}$ and $T_{B2A} = M^A_{c2v}(M^B_{c2v})^{-1}$.

C. OUR APPROACH AND CONTRIBUTION

We use a custom-designed marker with the system controller as a bridge between the camera and virtual spaces within a VR system. On the one hand, the geometry of the controller object is defined in its own local coordinate system, denoted by **device space**, whose 6-DoF pose M_{dev} is tracked by the VR system with respect to its virtual space (refer to Figure 1(a) again). On the other hand, the local coordinate system of the marker, denoted by **marker space**, is tracked by the camera system, which estimates its pose M_{mk} with respect to the camera space (see Figure 1(b) for the acrylic markers manufactured for the HTC VIVE Pro and Meta Quest 2 controllers).

In our framework, the user first fixes the marker to an arbitrary position of the VR controller. During this process, for easy use of the method, the user is allowed to attach the marker to the controller rather casually, leaving the transformation from the marker-to-device spaces unknown and yet to be determined (we denote this marker-to-device transformation by M_{m2d}). Note that the camera-to-VR transformation M_{c2v} can be expressed as $M_{c2v} = M^{-1}_{dev}M_{m2d}M_{mk}$. In fact, M_{dev} and M_{mk} can be easily estimated by *registering the controller with the marker to the camera*. In other words, we can obtain their measurements $\tilde{M}_{dev}$ and $\tilde{M}_{mk}$ by having the VR and camera systems track both the controller and marker simultaneously. This implies that if we can find M_{m2d}, it is possible to estimate M_{c2v} using $\tilde{M}_{dev}$ and $\tilde{M}_{mk}$, and eventually solve our problem.

In this work, we present a method for robustly estimating the marker-to-device transformation M_{m2d} using the measurements of the camera-to-marker transformation M_{mk} and the VR-to-device transformation M_{dev}. First, investigate the theoretical aspect of the estimation process systematically and show that exactly three carefully performed registrations are necessary and sufficient to uniquely decide M_{m2d}. Then, to cope with the inaccuracies and noises that inevitably exist in these measurements, we propose a computational procedure that estimates M_{m2d} from multiple registrations in a least-squares manner. Once the VR world is physically connected to the real world using the camera as an anchor point, it will be possible for colocated users from other VR, AR, and MR worlds to leverage the comparative advantages of technologies from different realities within an integrated XR environment.

Before we describe our method, we estimate the marker-to-device transformation M_{m2d} to assess the camera-to-VR transformation M_{c2v}. Basically, the same mathematical framework for M_{m2d} can be applied to directly estimate M_{c2v} (i.e., they are fundamentally the same problem). In this work, we describe our results in the context of obtaining M_{m2d} and discuss how they can be modified easily to obtain M_{c2v} directly.

II. PREVIOUS WORK

There is a vast body of literature on synergistic combinations of virtual, augmented, and mixed realities for producing effective extended reality applications. In this section, however, we only focus on the previous approaches that are directly related to the alignment of geometric spaces used in the respective realities. For the other works on the general asymmetric virtual environments, refer to the recent works, for instance, Cho et al. [9].

Alignment of world spaces unique to heterogeneous XR devices involves the estimation of the three-dimensional (3D) transformations between them. This section discusses previous work related to estimating such 3D transformations.

A. ALIGNMENT BY MANUAL PLACEMENT

To enable the co-experience for heterogeneous XR devices, the coordinate system of each device must be registered to the same canonical world space. A trivial way to align the world spaces unique to these devices is to position each device at positions with 6-DoF poses specified in the canonical space. By sampling pairs of world coordinates in the device and the canonical spaces, and computing the geometric relation between them, the world space of a device can be aligned to the canonical world. Roo and Hachet [6] aligned the world spaces of such devices as projectors, cameras and VR devices, to construct a hybrid mixed reality environment on a work table, where the VR space was registered by placing a VR controller at a pre-defined canonical position on the table. Grandi et al. [5] first aligned the AR space to the physical space via marker tracking, to which the VR space was additionally aligned by having a VR user stand at a specific location. Piumsomboon et al. [7] aligned the AR and VR spaces by fixing the coordinate system of the AR device at the center of the workspace.

On the other hand, Gottlieb [2] integrated the VR space to the AR space by manually aligning a real trackable device to its virtual 3D model while seeing both objects through the display. Roo and Hachet [10] estimated the geometric relation between the real and the virtual spaces by placing a head-mount display at the known position. In addition, Azimi et al. [11] utilized a cube textured with QR markers to estimate 3D transform between the camera and the display. These methods provided visual guidance to indicate where the tracked object should be located by rendering the 3D virtual model. However, they often suffered markedly from the human errors, which led to the space alignment errors easily ranging a few centimeters.

B. ALIGNMENT BY TRACKING

To minimize human intervention, some researchers used specially-designed tracking boards which can be recognized from both the canonical and the device world spaces through trackable devices. Bai et al. [12] built a target board that consisted of a 2D reference image and a VR tracker. The reference image produced the 6-DoF axes in the AR camera space while the paired VR tracker produced the 6-DoF axes in the VR device world space. Chun et al. [4] and An et el. [8] proposed to use a custom-made target board combined with VR trackers, in which the relation between the physical and the VR spaces had been pre-calibrated carefully on the board for precise space alignment. These tracking-based alignment methods increased the accuracy and stability as they minimized any human errors. However, the users needed to build complicated tracking objects. Besides, the relative 3D transform between tracking objects had to be pre-calibrated precisely to reduce the alignment bias. If this pre-calibration of the target board had an error, it was very difficult to correct the bias on the fly.

In addition, Weissker et al. [13] investigated the problem of inconsistent reference systems that occurred between colocated users using the HTC Vive systems. In particular, they proposed a computational method that allows to map the tracking data of each HTC Vive setup to the coordinate system of a reference user. While their method markedly improved the spatial consistency between collocated users, it was only applied to the homogeneous VR users who employed the HTC Vive system. Unlike the previous approaches based on either the fixed-point calibration or the marker-based calibration, Reimer et al. [14] exploited hand tracking data as spatial anchors of colocated users wearing SLAM-enabled VR headsets such as Meta Quest. For effective calibration between the colocated users, their method used an additional device capable of hand tracking. In contrast, our method aligns multiple heterogeneous AR and VR devices by attaching QR markers on tracking devices and registering other virtual spaces into one camera space.

C. SENSOR CALIBRATION WITH $AX = XB$

In computer vision and robotics, the equation of the form $AX = XB$ has frequently been formulated in various applications such as camera calibration, robot eye-to-hand calibration, Cartesian robot hand calibration, and image guided therapy sensor calibration. In this equation, A, B and X are each rigid-body motion in 3D space, where A and B are generated from sensor measurements while X is unknown. Since the numerical solutions for the equation were proposed by Shiu et al. [15] and Park et al. [16], it has frequently been solved in several variants, for instance, [17], [18], [19], [20].

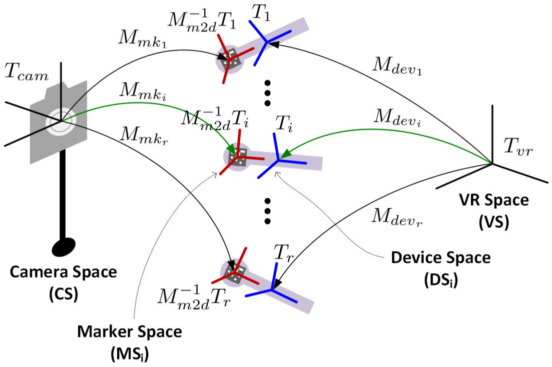


FIGURE 2. **Registrations of VR controller. To accurately estimate the relative transformation $M_{c2v} = T_{vr}T^{-1}_{cam}$ from the camera to the VR spaces, we perform a basic registration operation r times using a marker-attached VR controller with different 6-DoF poses (from top to bottom), resulting in r intermediate marker (in red) and device (in blue) spaces within the physical space, respectively. The relative poses M_{mk_i} (from the camera to ith marker spaces) and M_{dev_i} (from the VR to ith device spaces) are measured by tracking the marker in the camera space and the controller in the VR space, respectively. The common transformation M_{m2d} from the corresponding marker to the device spaces is still unknown but must be estimated.**

As considered in [15] and [16], the homogeneous transform equation of the form $AX = XB$ has two degrees of freedom, demanding two 'independent' equations for a unique solution. In real situations including ours, noises are often present in the measured values for the motions A and B. Therefore, the rigid transformation X was practically estimated through the least-squares approximation from a larger set of measurements $\{(A_1, B_1), \cdots, (A_n, B_n)\}$. The best-fit solution for X is usually obtained by treating the rotational and translational parts separately. In particular, the rotational part of X may be obtained by applying those numerical algorithms that were designed for finding a best-fit rotation matrix between two paired sets of noise-prone 3D points (Kabsch [21], Umeyama [22] and Arun et al. [23]). By contrast, our computational framework takes the approach presented in [16] and [24], where the rotation is calculated with the help of the Lie algebra.

III. OUR METHOD FOR ESTIMATING 3D TRANSFORMATIONS FROM CAMERA-TO-VR SPACES

A. REGISTRATIONS OF MARKER-ATTACHED VR CONTROLLERS

Figure 2 shows the physical space where a camera system and the virtual immersive space of a VR system exist. Let T_{cam} and T_{vr} denote the 6-DoF poses of the coordinate frames for the camera and VR systems with respect to the reference coordinate system for the physical space. Then, the problem to be solved is to *infer the relative pose* $M_{c2v} = T_{vr}T^{-1}_{cam}$, which enables us to geometrically connect the VR space to the camera space in the real-world physical workspace. In this setup, if the reference coordinate frame is naturally set to that of the camera system, it becomes that $M_{c2v} = T_{vr}$.

In the proposed method, the key operation for estimating M_{c2v} is to register the marker-attached VR controller in the physical space where the local coordinate frames of the marker and the controller object can be simultaneously tracked with respect to the camera and VR system, respectively. As will be explained shortly, we conducted this basic registration operation multiple times (r times) using the controller by varying its 6-DoF pose in the real-world space. During this process, we identified two intermediate space groups (see Figure 2 again). The first group refers to the set of the device spaces of the controller determined by each registration (the blue coordinate frames), where we denote the absolute pose in the reference space at the time of the ith registration by $T_i \, (i = 1, 2, \cdots, r)$. The second group denotes the set of the corresponding marker spaces (the red coordinate frames), in which the absolute pose at the time of the ith registration can be expressed as $M^{-1}_{m2d}T_i$ (recall the definition of the rigid transformation M_{m2d}).

Then, the camera, marker, device, and VR spaces together form a graph whose nodes corresponding to their absolute poses are interrelated through the edges of the relative poses. Here, the edges are classified into three kinds. The first group represents the relative pose M_{mk_i} that relates the camera space T_{cam} to the ith marker space $M^{-1}_{m2d}T_i$. The second one corresponds to the relative pose M_{dev_i} that relates the VR space T_{vr} to the ith device space of the controller T_i. The third one represents the single (unknown) relative pose M_{m2d} which, as defined earlier, transforms from the marker space to the device space. In this article, although they both represent the 3D rigid transformation, we use the symbols T and M to indicate the absolute and relative poses, respectively.

Now, if at least a pair of M_{mk_i} and M_{dev_i} for any i, and M_{m2d} is known, the goal of finding the 3D transformation between the camera and VR spaces in the real space can be trivially achieved. Note that, during the registration operation, the estimates $\tilde{M}_{mk_i}$ and $\tilde{M}_{dev_i}$ of M_{mk_i} and M_{dev_i} can respectively be obtained easily by a marker-tracking software and the VR controller tracking system. However, their fidelity is greatly dependent on several factors, such as the accuracy and precision of the sensing devices, or the lighting condition, among others. For each pair of estimates $\tilde{M}_{mk_i}$ and $\tilde{M}_{dev_i}$ obtained in the registration session, the transformations $\tilde{M}^{-1}_{dev_i}M_{m2d}\tilde{M}_{mk_i}$, which should be equal to M_{c2v} for all i, might be fairly different to each other. Moreover, M_{m2d} is not known in advance because the marker is attached to an arbitrary surface within the controller.

B. DERIVATION OF CONSTRAINTS FROM MEASUREMENT DATA

Figure 3 illustrates the pose graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ built from the entire set of measurements $S_{poses} = \{(\tilde{M}_{mk_i}, \tilde{M}_{dev_i}) \,|\, i = 1, 2, \cdots, r\}$ (refer to Figure 2 again). The node (vertex) set $\mathcal{N}$ represents the $2r + 2$ absolute 6-DoF poses corresponding to the four types of the coordinate spaces defined in the physical space, where **CS**, MS$_i$, DS$_i$, and **VS** respectively represent the camera, the ith marker, the ith device, and the VR spaces, respectively. The edge set $\mathcal{E}$ is then partitioned into two groups of edges each corresponding to a different type of

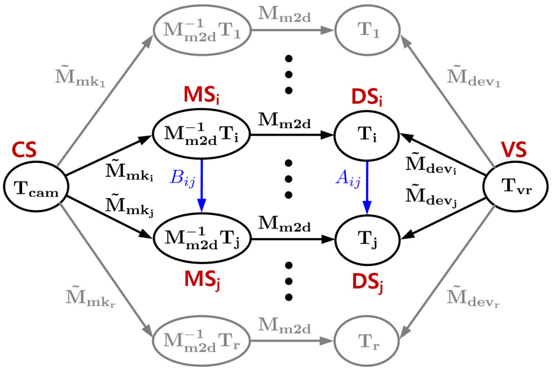

$\mathrm{M_{m2d}^{-1}T_1}$
$\mathrm{M_{m2d}}$
$\mathrm{T_1}$
$\tilde{\mathrm{M}}_{\mathrm{mk_1}}$
$\tilde{\mathrm{M}}_{\mathrm{dev_1}}$
MS$_\mathrm{i}$
DS$_\mathrm{i}$
CS
VS
$\mathrm{T_{cam}}$
$\mathrm{M_{m2d}^{-1}T_i}$
$\mathrm{M_{m2d}}$
$\mathrm{T_i}$
$\mathrm{T_{vr}}$
$\tilde{\mathrm{M}}_{\mathrm{mk_i}}$
B_{ij}
A_{ij}
$\tilde{\mathrm{M}}_{\mathrm{dev_i}}$
$\tilde{\mathrm{M}}_{\mathrm{mk_j}}$
$\tilde{\mathrm{M}}_{\mathrm{dev_j}}$
$\mathrm{M_{m2d}^{-1}T_j}$
$\mathrm{M_{m2d}}$
$\mathrm{T_j}$
MS$_\mathrm{j}$
DS$_\mathrm{j}$
$\tilde{\mathrm{M}}_{\mathrm{mk_r}}$
$\tilde{\mathrm{M}}_{\mathrm{dev_r}}$
$\mathrm{M_{m2d}^{-1}T_r}$
$\mathrm{M_{m2d}}$
$\mathrm{T_r}$

FIGURE 3. Derivation of $A_{ij}X = XB_{ij}$ from the pose graph. The loop closure condition derived from the ith and jth registrations naturally forms the equation $A_{ij}X = XB_{ij}$ where the unknown rigid transformation $X = M_{m2d}$ must be solved for $A_{ij} = \tilde{M}_{dev_j}\tilde{M}_{dev_i}^{-1}$ and $B_{ij} = \tilde{M}_{mk_j}\tilde{M}_{mk_i}^{-1}$.

constraint. The first partition represents the links formed by tracking either the marker $(\tilde{M}_{mk_1}, \tilde{M}_{mk_2}, \cdots, \tilde{M}_{mk_r})$ or the controller $(\tilde{M}_{dev_1}, \tilde{M}_{dev_2}, \cdots, \tilde{M}_{dev_r})$. The second one consists of the r links of the same kind M_{m2d} that indicates the (unknown) relative pose between the marker and the device spaces.

For every $i = 1, 2, \cdots, r$, the composite transformation $T_{vr}^{-1}\tilde{M}_{dev_i}^{-1}M_{m2d}\tilde{M}_{mk_i}T_{cam}$ should be the identity transformation I in $\mathbb{R}^3$. Then, the loop closure constraint

$$T_{vr}^{-1}\tilde{M}_{dev_i}^{-1}M_{m2d}\tilde{M}_{mk_i}T_{cam} = T_{vr}^{-1}\tilde{M}_{dev_j}^{-1}M_{m2d}\tilde{M}_{mk_j}T_{cam},$$

formed by pairing the ith and jth measurements $(i \neq j)$, can be represented in a more compact form as follows:

$$\tilde{M}_{dev_j}\tilde{M}_{dev_i}^{-1}M_{m2d} = M_{m2d}\tilde{M}_{mk_j}\tilde{M}_{mk_i}^{-1}. \tag{1}$$

If we let $A_{ij} \triangleq \tilde{M}_{dev_j}\tilde{M}_{dev_i}^{-1}$, $B_{ij} \triangleq \tilde{M}_{mk_j}\tilde{M}_{mk_i}^{-1}$, and $X \triangleq M_{m2d}$, Equation 1 is compactly expressed as $A_{ij}X = XB_{ij}$, for which the unknown rigid transformation X must be solved.

C. SOLVING THE SYSTEM OF EQUATIONS $A_{IJ}X = XB_{IJ}$

Solving the system of equations of $A_{ij}X = XB_{ij}\,(i \neq j$ and $i, j = 1, 2, \ldots, r)$ requires the computation of 3D rigid transformation $T = \begin{bmatrix} R_T & t_T \\ 0^\top & 1 \end{bmatrix} \in SE(3)$, which is composed of the rotational $(R_T \in SO(3))$ and translational $(t_T \in \mathbb{R}^3)$ components. The rotation is often represented by a vector $\omega \in \mathbb{R}^3$, where its unit vector $\bar{\omega} \triangleq \frac{\omega}{||\omega||}$ and length $||\omega||$ indicate the rotational axis and angle, respectively. Given the rotation matrix R_T of T, the corresponding 3-vector ω_T can be obtained by *the capitalized logarithmic map* Log : $SO(3) \rightarrow \mathbb{R}^3$ as $\omega_T = \mathrm{Log}(R_T)$. (Please refer to [25], [26] for a quick introduction to the Lie theory.)

Consider the single equation $A_{12}X = XB_{12}$ formed by the first and second registrations. From its matrix representation:

$$\begin{bmatrix} R_{A_{12}} & t_{A_{12}} \\ 0^\top & 1 \end{bmatrix}\begin{bmatrix} R_X & t_X \\ 0^\top & 1 \end{bmatrix} = \begin{bmatrix} R_X & t_X \\ 0^\top & 1 \end{bmatrix}\begin{bmatrix} R_{B_{12}} & t_{B_{12}} \\ 0^\top & 1 \end{bmatrix}, \tag{2}$$

we are led to the following constraints:

$$R_{A_{12}}R_X = R_XR_{B_{12}} \text{ and} \tag{3}$$

$$R_{A_{12}}t_X + t_{A_{12}} = R_Xt_{B_{12}} + t_X. \tag{4}$$

First, the rotational constraint in Equation 3 dictates an important fact between the rotations in $A_{12} = \tilde{M}_{dev_2}\tilde{M}_{dev_1}^{-1}$ and $B_{12} = \tilde{M}_{mk_2}\tilde{M}_{mk_1}^{-1}$.

Lemma 1. *Let $\omega_{A_{12}} = Log(R_{A_{12}})$ and $\omega_{B_{12}} = Log(R_{B_{12}})$ for given $A_{12}, B_{12} \in SE(3)$. If an equation $A_{12}X = XB_{12}$ is to hold for some $X \in SE(3)$, then it is necessary that $||\omega_{A_{12}}|| = ||\omega_{B_{12}}||$.*

(For the sake of readability, the proofs for the lemmas and theorem in this subsection are provided in the Appendix.)

This lemma implies that, for any equation $A_{ij}X = XB_{ij}$ to have a solution, the amounts of rotation involved in the coefficients A_{ij} and B_{ij} (i.e., $||\omega_{A_{ij}}||$ and $||\omega_{B_{ij}}||$) must be the same although the rotational axes $\bar{\omega}_{A_{ij}}$ and $\bar{\omega}_{B_{ij}}$ may differ. This is intuitively clear because, in the proposed frame work, a rigid marker is fixed on a rigid VR controller. Note that in previous works [15], [16], the assertion of this lemma was investigated in different context.

Now, in order to find the unknown rigid transformation X from the system of equations $A_{ij}X = XB_{ij}$, the first thing to know is how many registrations of VR controller are at least needed.

Lemma 2. *Given $A_{12}, B_{12} \in SE(3)$, let $\omega_{A_{12}} = Log(R_{A_{12}})$ and $\omega_{B_{12}} = Log(R_{B_{12}})$. Then, the equation $A_{12}X = XB_{12}$ has infinitely many solutions $X \in SE(3)$ if $||\omega_{A_{12}}|| = ||\omega_{B_{12}}||$.*

Therefore, the single equation does not allow us to uniquely determine the unknown rigid transformation $X = M_{m2d}$. Now, consider the system of equations $A_{12}X = XB_{12}$ and $A_{23}X = XB_{23}$ that are formulated by the first three registrations. The following constraints can be obtained similarly:

$$R_{A_{12}}R_X = R_XR_{B_{12}} \;\&\; R_{A_{23}}R_X = R_XR_{B_{23}} \text{ and} \tag{5}$$

$$\begin{aligned} R_{A_{12}}t_X + t_{A_{12}} &= R_Xt_{B_{12}} + t_X \\ R_{A_{23}}t_X + t_{A_{23}} &= R_Xt_{B_{23}} + t_X. \end{aligned} \tag{6}$$

The following theorem reveals that two "independent" equations are necessary and sufficient to guarantee the unique existence of the solution.

Theorem. *Given $T = A_{12}, A_{23}, B_{12}, B_{23} \in SE(3)$, let $\omega_T = Log(R_T)$. Then, the system of equations $A_{12}X = XB_{12}$ and $A_{23}X = XB_{23}$ has the unique solution $X \in SE(3)$ if (i) $||\omega_{A_{12}}|| = ||\omega_{B_{12}}||$, (ii) $||\omega_{A_{23}}|| = ||\omega_{B_{23}}||$, and (iii) $\omega_{A_{12}} \times \omega_{A_{23}} \neq 0$ and $\omega_{B_{12}} \times \omega_{B_{23}} \neq 0$.*

Note that the first condition $\omega_{A_{12}} \times \omega_{A_{23}} \neq 0$ demands that the three consecutive registrations should be conducted carefully in such a way that (i) the coordinate frame of device space rotates nontrivially between the first and second registrations $(\omega_{A_{12}} \neq 0)$ and between the second and third registrations,$(\omega_{A_{23}} \neq 0)$ and (ii) the two consecutive rotations do not share the same rotational axis ($\omega_{A_{12}} \times \omega_{A_{23}} \neq 0$). This same condition is applied to the coordinate frame of marker space due to the second condition $\omega_{A_{12}} \times \omega_{A_{23}} \neq 0$.

Corollary. *Three careful registrations of VR controller are necessary to uniquely determine the unknown transformation M_{m2d}. In particular, the controller must nontrivially rotate between the registrations, and the rotational axes must differ between the two consecutive rotations.*

D. ESTIMATING $X = M_{M2D}$ IN A LEAST-SQUARES MANNER

According to the theorem in the previous subsection, it can be inferred that three carefully performed registrations are necessary and sufficient to uniquely determine the rigid transformation $X = M_{m2d}$. As mentioned earlier, the two necessary conditions (i) and (ii) in the theorem appear to trivially hold because a rigid marker is fixed on the rigid controller. However, in our framework, the coefficients of the equations $A_{ij} = \tilde{M}_{dev_j}\tilde{M}_{dev_i}^{-1}$ and $B_{ij} = \tilde{M}_{mk_j}\tilde{M}_{mk_i}^{-1}$ are generated by tracking both the marker via the camera ($\tilde{M}_{mk_i}$ and $\tilde{M}_{mk_j}$) and the controller via the VR sensors ($\tilde{M}_{dev_i}$ and $\tilde{M}_{dev_j}$). Because nontrivial tracking errors often occur by consumer-grade cameras and VR systems, it is not reasonable to assume that the two necessary conditions always hold mathematically.

Moreover, to obtain a robust estimation of M_{m2d}, instead of solving for the unique solution using the two simultaneous equations as in, e.g., [15] and [16], we use a least-squares approach over a larger set of registration data. Given the coefficient matrices $A_{(k)} \triangleq A_{k,k+1}$ and $B_{(k)} \triangleq B_{k,k+1}$ generated by the kth and $(k+1)$th registrations, consider the system of n equations made of $A_{(k)}X = XB_{(k)}$, $k = 1, 2, \ldots, n\,(n \geq 2)$, for which R_X and t_X of X is to be estimated. From the kth equation, we get the rotational ($R_{A_{(k)}}R_X = R_X R_{B_{(k)}}$) and translational ($R_{A_{(k)}}t_X + t_{A_{(k)}} = R_X t_{B_{(k)}} + t_X$) constraints as before. For effective computation of R_X, we use the equivalent constraint $\omega_{A_{(k)}} = R_X\omega_{B_{(k)}}$ which can be derived from the rotational constraint, as shown in the proof of **Lemma 1**. For the error metric $f_k^{total}(R_X, t_X) = f_k^{rot}(R_X) + f_k^{trans}(R_X, t_X)$ where

$$f_k^{rot}(R_X) \triangleq ||R_X\omega_{B_{(k)}} - \omega_{A_{(k)}}||^2 \text{ and} \tag{7}$$

$$f_k^{trans}(R_X, t_X) \triangleq ||(R_{A_{(k)}} - I)t_X - R_X t_{B_{(k)}} + \boldsymbol{t}_{A_{(k)}}||^2, \tag{8}$$

we obtain an optimal X^* of R_X^* and t_X^* by solving the following minimization problem:

$$X^* = \arg\min_X \Sigma_{k=1}^n f_k^{total}(R_X, t_X).$$

Following [16], we find a "best-fit" solution in two steps by separating the rotational and translational components. First, we obtain R_X^* that minimizes $\min_{R_X} \Sigma_{k=1}^n f_k^{rot}(R_X)$. In fact, the optimal rotation is $R_X^* = (M^\top M)^{-\frac{1}{2}}M^\top$ for $M = \Sigma_{k=1}^n \omega_{B_{(k)}}\omega_{A_{(k)}}^\top$, which was revealed in [24]. Once the optimal rotation matrix R_X^* is known, the optimal translation vector t_X^* can be computed by solving the standard least-squares problem $\min_{t_X} \Sigma_{k=1}^n f_k^{trans}(R_X^*, t_X)$. In this manner of least-squares approximation, we can estimate the rigid transformation $X = M_{m2d}$ effectively from the possibly error-prone measurements generated during the registration process.

E. MODIFICATION FOR UNKNOWN MARKER SIZE

So far, it is assumed that the exact size of the marker, which is fed to the marker tracking software, is known. Sometimes, it is convenient to allow a VR user who does not care for the marker size. This case can be handled by introducing a scale factor σ in the translational part of $B_{(k)} = B_{k,k+1} = \tilde{M}_{mk_{k+1}}\tilde{M}_{mk_k}^{-1}$. (e.g., for $k = 1$, $t_{B_{12}}$ in Equation 2 is replaced by $\sigma t_{B_{12}}$.) Then, the unknown scale factor σ can be estimated simultaneously with M_{m2d} using the presented least-squares method, in which the translational error term in Equation 8 is slightly modified as follows:

$$f_k^{trans}(R_X, t_X, \sigma) \triangleq ||(R_{A_{(k)}} - I)t_X - \sigma R_X t_{B_{(k)}} + \boldsymbol{t}_{A_{(k)}}||^2.$$

F. COMPUTATION OF CAMERA-TO-VR TRANSFORMATION M_{C2V}

As mentioned in the Introduction, the problem of estimating the camera-to-VR transformation M_{c2v} is fundamentally the same as that of estimating the marker-to-device transformation M_{m2d}. Recall the relation $M_{c2v} = M_{dev}^{-1}M_{m2d}M_{mk}$ between M_{c2v} and M_{m2d}, which can be rearranged as $I = M_{mk}M_{c2v}^{-1}M_{dev}^{-1}M_{m2d}$. Then, from the ith and jth registrations, we have the relation $\tilde{M}_{mk_i}M_{c2v}^{-1}\tilde{M}_{dev_i}^{-1}M_{m2d} = \tilde{M}_{mk_j}M_{c2v}^{-1}\tilde{M}_{dev_j}^{-1}M_{m2d}$, which results in the equation: $\tilde{M}_{dev_j}^{-1}\tilde{M}_{dev_i}M_{c2v} = M_{c2v}\tilde{M}_{mk_j}^{-1}\tilde{M}_{mk_i}$.

Again, by letting $A_{ij} \triangleq \tilde{M}_{dev_j}^{-1}\tilde{M}_{dev_i}$, $B_{ij} \triangleq \tilde{M}_{mk_j}^{-1}\tilde{M}_{mk_i}$, and $X \triangleq M_{c2v}$, we are led to the equation of the same form $A_{ij}X = XB_{ij}$, which can be solved by the basically the same method as M_{m2d} with slight differences in the coefficients A_{ij} and B_{ij}. Please note the similarity between the last equation and Equation 1.

IV. RESULTS

To demonstrate the effectiveness of the presented mathematical framework for estimating M_{m2d} clearly, we implemented a numerical algorithm and tested with two consumer-grade VR systems: HTC VIVE Pro and Meta Quest 2. In this experiment, we defined a workspace of size 3 m×3 m in a room, within which the 6-DoF poses of the HTC VIVE Pro's headset and controller were tracked using three lighthouse base stations. On the other hand, the standalone Meta Quest 2 system used simultaneous localization and mapping to track its headset, and constellation tracking to track its controller. For a camera system defining the camera space (or the reference space if $T_{cam} = I$) in the physical space, we positioned a Microsoft Azure Kinect sensor along the workspace boundary so that it provided a good view of the central working area. Finally, the square-shaped markers of sides 4 cm and 5 cm were attached on the controllers of the HTC VIVE Pro and Meta Quest 2 systems, respectively.

As dictated by the theorem in the previous section, three careful registrations of VR controllers are sufficient for uniquely determining M_{m2d}. However, to consider the inaccuracies and noises that occur in the measurement, 10 legitimate registrations were made per VR controller ($r = 10$), from which M_{m2d} was estimated in the least-squares manner as described previously. During the registration process, it had to be checked if the necessary conditions specified in the theorem were satisfied between the consecutive registrations. In particular, it was important to remove an $(i + 1)$th registration from consideration if $||\omega_{A_{i,i+1}}||$ and $||\omega_{B_{i,i+1}}||$ differ by more than a given threshold, which happened intermittently due to the tracking error of the commodity VR systems. Furthermore, in constructing the linear system for the least-squares approximation, we included the equations $A_{ij}X = XB_{ij}$ for all combinations of (i, j) where $i \in [1, 9]$, $j \in [2, 10]$, and $i < j$. Although this added the seemingly unnecessary redundant equations in the system (note the equations of the form $A_{i,i+1}X = XB_{i,i+1}$ are sufficient), the inaccuracy in the estimation due to any possible outliers that occurred while tracking the VR controllers in particular was minimized.

A. ACCURACY IN THE ESTIMATION OF M_{M2D}

To evaluate the accuracy of the estimation of the developed numerical solution, a ground truth marker-to-device transformation M_{m2d} must be created for our experiments. Fortunately, we were able to use the 3D computer-aided design (CAD) models of the controllers provided by their vendors and those of the markers designed by ourselves. For this, the marker model was placed on the virtual model of the respective controller using CAD software so that the marker is well aligned with the controller (see Figure 4). Then, the inverse of the 6-DoF pose of the marker's coordinate frame expressed with respect to the local space of the controller model was set to the ground truth transformation M^{gr}_{m2d}.

During the experiment, we attached the physical marker to the actual controller as closely as possible in the same way as we have done using the CAD models. Then, after obtaining the estimate $\tilde{M}_{m2d} \triangleq (R_{\tilde{M}_{m2d}}, t_{\tilde{M}_{m2d}})$, it was compared to the ground truth $M^{gr}_{m2d} \triangleq (R_{M^{gr}_{m2d}}, t_{M^{gr}_{m2d}})$. First, the rotational error metric was defined as $\epsilon^{rot} \triangleq ||\text{Log}(R^{\top}_{M^{gr}_{m2d}} R_{\tilde{M}_{m2d}})||$, which represents the angular difference between the two transformations. Then, we defined the translational error metric as $\epsilon^{trans} \triangleq ||t_{\tilde{M}_{m2d}} - t_{M^{gr}_{m2d}}||$, whose meaning is obvious.

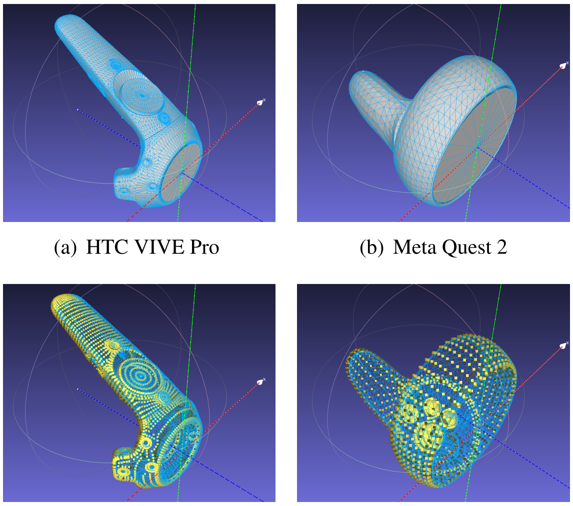


FIGURE 4. Generation of the ground truth transformation M^{gr}_{m2d}. (a) & (b) Using CAD software, we carefully positioned the 3D model for the marker on the top side of the controller so that the actual marker can be attached to the physical controller as closely as possible in the registration process. The inverse of the relative pose of the marker's coordinate frame (denoted by RGB lines) in the local space of the controller model was then set to the ground truth transformation M^{gr}_{m2d}. (c) & (d) When we made an estimate $\tilde{M}_{m2d}$ using the presented method, the 3D CAD model of the controller was transformed to the marker space using $(M^{gr}_{m2d})^{-1}$ (in sky blue) and $\tilde{M}^{-1}_{m2d}$ (in yellow), respectively, and rendered simultaneously to visually verify the estimation error.

Note that these errors actually include not only the errors introduced while estimating M_{m2d} with our method, but also those caused when attaching the marker to the controller as done in the virtual space. We tried out best to minimize the second type of error so that the resulting error estimates reflect the effectiveness of the presented numerical framework as precisely as possible. It should also be emphasized that the exact M_{m2d} is practically unknown because the proposed framework allows a user to conveniently fix an arbitrarily shaped marker at any controller location. In addition, the 3D CAD models for the marker are often not available.

Table 1 summarizes the estimation errors obtained by the developed numerical solution for the two heterogeneous VR systems. When the exact size of the marker was provided, which is usual, the VIVE Pro controller showed the rotational (ϵ^{rot}) and translational (ϵ^{trans}) errors of 1.602° and 2.122 mm on average. With respect to the ground truth of $||\text{Log}(R_{M^{gr}_{m2d}})|| = 180.0°$ and $||t_{M^{gr}_{m2d}}|| = 37.85$ mm, the relative errors amounted to 0.890% (rotation) and 5.61% (translation). These errors were quite low considering that the rotational and translational errors from the tracking system were often reported to easily increase up to 5° and 15 mm, respectively [8], [27], [28], [29]. While the tracking errors are known to be greater on the standalone Quest 2 system, the average estimation errors were shown to be 1.302° (rotation) and 3.644 mm (translation), respectively corresponding to the relative errors of 0.764% and 9.87% as $||\text{Log}(R_{M^{gr}_{m2d}})|| = 170.4°$ and $||t_{M^{gr}_{m2d}}|| = 36.91$ mm.

We also tested the case when the exact size of the marker is not available. Then, both M_{m2d} and the scale factor σ had to be estimated simultaneously, which caused an additional amount of translational error as revealed in the table. In summary, we believe that the presented numerical framework for estimating M_{m2d} entailed quite acceptable

TABLE 1. **Estimation errors for M_{m2d}. The rotational and translational errors are given in degrees and in millimeters, respectively. The pair (ϵ^{rot}, ϵ^{trans}) indicates the errors that occurred in the usual case when the exact size of the marker was provided. By contrast, if the size was assumed to be unknown, the scale factor σ had also to be estimated along with M_{m2d}, in which $\epsilon^{trans}_{\sigma}$ represents the corresponding translational error (note the exact value of σ^*, which is the scale factor adjusted against the exact size of the marker, should be 1.0). For each VR controller, we conducted five independent registration sessions, thus producing five datasets. Although we tried to conduct the same registration process in every data acquisition session, the introduced errors varied across the datasets. This was mainly because of the difficult-to-understand aspects of the rotational/translational variability and temporal noise experienced by the commodity tracking devices.**

(a) HTC VIVE Pro

	data 1	data 2	data 3	data 4	data 5
ϵ^{rot}	1.172	1.238	1.852	1.793	1.956
ϵ^{trans}	1.458	1.494	2.318	3.064	2.275
$\sigma*$	1.025	1.049	1.028	1.031	0.990
$\epsilon^{trans}_{\sigma}$	2.811	2.383	2.796	3.568	2.262

(b) Meta Quest 2

	data 6	data 7	data 8	data 9	data 10
ϵ^{rot}	1.733	1.225	1.074	1.291	1.185
ϵ^{trans}	5.012	1.948	2.773	4.724	3.763
$\sigma*$	1.023	1.009	1.031	1.004	1.016
$\epsilon^{trans}_{\sigma}$	5.186	2.118	2.889	4.735	3.708

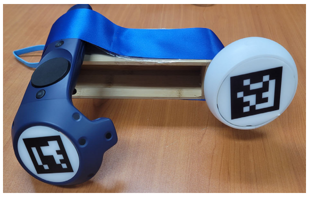


FIGURE 5. **Two VR controllers fixed together. To evaluate the stability observed when users from two heterogeneous VR systems interact with each other, we locked the HTC VIVE Pro and Meta Quest 2 controllers together so that they were roughly 220 mm apart. Then, we tested whether their measured relative 6-DoF pose remained the same while they moved within a workspace.**

accuracy on both VR systems despite several negative factors usually found in VR environments.

B. STABILITY IN COMBINING TWO VR SYSTEMS

In the next experiment, we investigated how effectively the combination of two consumer-grade VR systems of different vendors worked systematically in a shared physical space. Before the test began, we fixed a pair of HTC VIVE Pro and Meta Quest 2 controllers together (see Figure 5), and estimated the camera-to-VR transformations using the presented methods for the VIVE Pro controller ($\tilde{M}^V_{c2v}$) and the Quest 2 controller ($\tilde{M}^Q_{c2v}$), respectively. Once this setup was done, we repeatedly measured the 6-DoF poses $\tilde{M}^V_{dev_i}$ and $\tilde{M}^Q_{dev_i}$ of the two controllers simultaneously at selected sample locations ($i = 0, 1, \cdots, n_{samp} - 1$) while moving within a workspace.

TABLE 2. **Variations in the estimations of M^{V2Q}_{dev}. The rotational and translational quantities are given in degrees and in millimeters, respectively. In this experiment, we locked an HTC VIVE Pro controller and a Meta Quest 2 controller together so that they were about 220 mm apart. Thus, the relative transformation M^{V2Q}_{dev} between them should have remained the same even if they moved along a trajectory. To find out if this is true in real-world situations, we ran an experiment five times, in each of which 10 samples of M^{V2Q}_{dev} were taken while moving in a workspace ($n_{samp} = 10$). Before the last three experiment sessions began, we rotated the controllers slightly so that, compared to** `data 11` **and** `data 12`**, the rotational angle between them increased (**`data 13`**) and decreased (**`data 14` **and** `data 15`**) while trying to maintain their distance as much as possible. Then, the arithmetic means δ_{ave} and root mean squares (RMS) δ_{rms} of the rotational and translational deviations from their averages μ were evaluated. Note that the RMS value is always greater than or equal to the average because $\delta^2_{rms} = \delta^2_{ave} + \sigma^2_{\delta}$ for the variance σ^2_{δ} of the deviations.**

	data 11	data 12	data 13	data 14	data 15
μ^{rot}	22.71	22.35	29.14	15.47	17.80
δ^{rot}_{ave}	2.923	3.361	1.130	1.518	3.009
δ^{rot}_{rms}	3.250	3.618	1.229	1.607	3.260
μ^{trans}	209.3	216.9	217.7	249.9	202.5
δ^{trans}_{ave}	12.91	20.65	8.69	20.41	16.24
δ^{trans}_{rms}	13.54	26.20	10.70	22.20	17.71

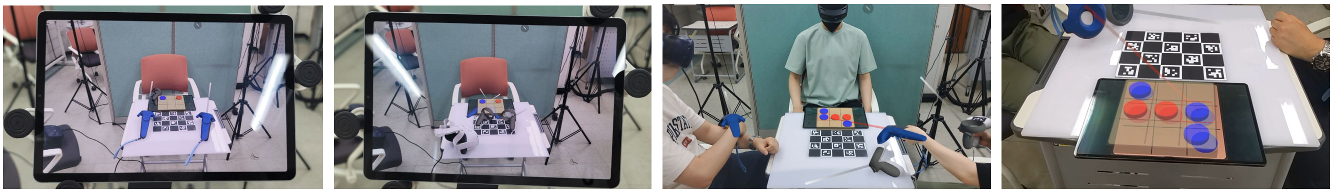
The relative 6-DoF pose from the HTC VIVE to Meta Quest 2 device space M^{V2Q}_{dev} should remain the same across the samples because they were locked together. More specifically, the following measurements of M^{V2Q}_{dev} must be identical for all i:

$$\begin{aligned}\tilde{M}^{V2Q}_{dev_i} &= \tilde{M}^Q_{dev_i}\tilde{M}^Q_{c2v}(\tilde{M}^V_{c2v})^{-1}(\tilde{M}^V_{dev_i})^{-1}\\ &= \tilde{M}^Q_{dev_i}\tilde{M}^Q_{c2v}(\tilde{M}^V_{dev_i}\tilde{M}^V_{c2v})^{-1}.\end{aligned}$$

See Figure 1(a) again to understand this relative pose, in which **(A)** and **(B)** correspond to the HTC VIVE Pro and Meta Quest 2 systems, respectively.

However, due to several reasons that will be explained further, they differed slightly between the sample measurements. To evaluate the stability of combining the two heterogeneous VR systems, we first computed the averages μ^{rot} and μ^{trans} of the rotational and translational parts of the estimations where $\mu^{rot} = \frac{1}{n_{samp}} \sum_i ||\mathrm{Log}(R_{\tilde{M}^{V2Q}_{dev_i}})||$ and $\mu^{trans} = \frac{1}{n_{samp}} \sum_i ||t_{\tilde{M}^{V2Q}_{dev_i}}||$. Then, it was investigated how widely the estimations $\tilde{M}^{V2Q}_{dev_i}$ varied across the samples by observing their deviations from the averages $d^{rot}_i = ||\mathrm{Log}(R_{\tilde{M}^{V2Q}_{dev_i}})|| - \mu^{rot}$ and $d^{trans}_i = ||t_{\tilde{M}^{V2Q}_{dev_i}}|| - \mu^{trans}$, and evaluating their arithmetic means δ_{ave} and root mean squares δ_{rms} of the absolute values of the respective deviations.

Table 2 shows the statistics on the deviations. Note that the ratio of the average deviation δ_{ave} to the estimated quantity μ (i.e., $\frac{\delta_{ave}}{\mu}$) is an indirect measure indicating the stability in tracking the 6-DoF poses of the two heterogeneous VR controllers in an integrated physical space. It can be observed that the ratios were 11.7% (rotation) and 7.17% (translation) on average where it varied from 3.88% to 16.9% and



(a) After the registration (VIVE Pro) (b) After the registration (Quest 2) (c) Seen from the AR user (d) Seen from the MR user

FIGURE 6. **A demonstration of an XR environment physically shared by heterogeneous reality users. (a) & (b) After the respective registration, the 3D model of the controller was rendered on the tablet PC display overlaying the captured image of the real controller. As seen in the two images, the visual differences were quite small on the display of the AR device. (c) & (d) Both the AR and MR users were able to visually see the laser pointers coming from the VR controllers. In (c), the Meta Quest 2 user on the right was using a HTC VIVE Pro controller to point at the tic-tac-toe board. In the snapshot image (d), captured by a camera on the headset, the real and virtual boards looked out of alignment with each other due to the disparity between the camera and the eyes. However, from the actual view of the MR user, they matched well on the holographic display.**

3.99% to 9.52%, respectively, across the datasets. These deviations (and their variations over the datasets) were mainly due to the inaccuracies and noises in $\tilde{M}^{V}_{dev_i}$ and $\tilde{M}^{Q}_{dev_i}$ that occurred while tracking the controllers at the sample locations.

In the consumer-grade VR systems, the tracking accuracy was markedly affected by the relative positions and orientation between the controllers and the sensing systems, making the rotational and translational errors increase unpredictably up to a few degrees and centimeters. Sometimes, the sensor on a VR controller was hidden by the other one from its tracking system, thus unpredictably deteriorating the measurement accuracy. These tracking errors and noises from the two VR systems were combined in complicated ways, often amplifying them and/or producing outliers in the measurements. We also conjectured that a signal interference between the two controllers took place, increasing the instability of the measurements. As widely agreed, these erroneous measurements are inherently unavoidable when developing VR applications. Considering all these negative factors, the deviations were found within quite acceptable margins of error, resulting in the accuracy sufficient for producing most XR applications.

C. APPLICATION

In addition to the quantitative analysis, we also implemented a proof-of-concept XR application, in which users from different realities play a game of tic-tac-toe in a physically integrated environment (see Figure 1(c) and Figure 6). In this demonstration, a VR user wearing an HTC VIVE Pro headset set up a workspace in a room by employing three lighthouse base stations. Another VR user participated in the demonstration using a standalone Meta Quest 2 system. To define the camera space, a Microsoft Azure Kinect camera was positioned in such a way that it had a good view of the central region of the workspace. Then, a touchscreen tablet PC (Samsung Galaxy Tab S8 Ultra) that displays a virtual tic-tac-toe board was placed on a table in the center of the workspace. Next to the tablet PC, a marker was also placed that can be recognized by the camera, an MR user wearing a Microsoft HoloLens 2 headset, and an AR user holding a table PC (Samsung Galaxy Tab S6). This marker, recognized by both the camera and the MR/AR users, connected the MR/AR users to the camera space. In addition, the 6-DoF pose of the virtual tic-tac-toe board was estimated with respect to the camera space by making the camera recognize another (temporary) marker on the touchscreen display before the game began.

After the four users from different realities geometrically registered themselves to the shared physical camera space through the presented framework (the two VR users) and marker tracking (the MR and AR users), they started to play the game. In this match, the two VR users were on the same side and the MR user was on the other side, taking turns playing the game. To place the tic-tac-toe pieces, the VR users used the laser pointers of the controllers, while the MR user touched the virtual board with a finger. On the other hand, the AR user worked as a spectator watching the game.

In this demonstration, the HTC VIVE Pro user employed an HTC VIVE Pro controller to point to the tic-tac-toe board. On the other hand, the Quest 2 user used both an HTC VIVE Pro controller and a Meta Quest 2 controller, which was possible because the heterogeneous VR systems geometrically shared the same physical workspace. While the two VR users were immersed in separate virtual worlds, they acted as if they played the game around the same physical board. On the other hand, the MR user could notice through the holographic glasses exactly what the VR users did in their immersive environments. For instance, when a VR user pointed at the tic-tac-toe board using the laser pointer, the MR users saw the same laser pointer as the VR user. Of course, when the MR user pointed to the board using a finger, the action was displayed on the screens of the VR headsets. All the activities of both VR and MR users were monitored by the AR user through the tablet screen that displayed both real and virtual images.

When the 3D models of the VR controllers were seen on the screen of the AR device after the respective registrations, overlaying the captured images of the real objects, the visual differences were quite small. However, during the game, we often observed some visual disparity in the rendering of the real and virtual objects. This was mainly because of the inaccuracies of the controller and marker tracking and presumably the interferences between the users and devices. While this phenomenon is inherent and unavoidable in developing XR applications, it remains a future work to reduce it.

V. CONCLUDING REMARKS

One of the essential requirements for enabling colocated VR, AR, and MR users to collaborate in an integrated XR environment while making the best use of immersive technologies from the other realities is to unify their heterogeneous workspaces, which are unique to each reality. Under this direction, we proposed a methodology for geometrically connecting the virtual world of a VR user to the physical space where AR and MR users reside. Then, we presented the computational procedure that implemented the proposed mathematical theory effectively. The proof-of-concept demonstration indicated that the developed method allowed a VR user to interact with not only AR and MR users but also a VR user from a different VR system in a geometrically unified physical space. Furthermore, the test results revealed that the numerical accuracy and stability are both sufficiently good for developing most consumer-grade XR applications.

As is noted, the geometrical errors in the estimation of the relative 6-DoF poses of XR users are produced by a complex combination of several factors such as, for instance, the spatiotemporal noises in the tracking signals or interferences between the XR devices. In the future, we intend to conduct an in-depth analysis of the complicated geometrical errors introduced while unifying the heterogeneous reality. This effort will help further enhance the numerical accuracy and stability of the space alignment procedure.

APPENDIX: PROOFS OF THE LEMMAS AND THE THEOREM IN SUBSECTION III.C

Let $\omega = (\omega_x, \omega_y, \omega_z)^\top$ be a 3-vector in $\mathbb{R}^3$ that represents the rotation by angle $||\omega||$ around the axis of direction $\bar{\omega} \triangleq \frac{\omega}{||\omega||}$. The 3×3 rotation matrix representation R_ω composes a group of 3D rotation matrices, called the *special orthogonal group* $SO(3)$. On the other hand, the skew-symmetric matrix representation $[\omega]_\times \triangleq \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}$ of rotation ω forms the Lie algebra of $SO(3)$, denoted by $\mathfrak{so}(3)$. The *exponential map* $\exp : \mathfrak{so}(3) \to SO(3)$ allows us to transfer from elements of the Lie algebra to those of the rotation matrix group as $R_\omega = \exp([\omega]_\times)$. Its inverse operation is the *logarithmic map* $\log : SO(3) \to \mathfrak{so}(3)$, in which $[\omega]_\times = \log(\exp([\omega]_\times))$. The *capitalized* versions of these two operations are shortcuts to map directly the vector space $\mathbb{R}^3$ to/from $SO(3)$, where $R = \text{Exp}(\omega)$ and $\omega = \text{Log}(R)$ for the same rotation $\omega \in \mathbb{R}^3$ and $R \in SO(3)$. In the below description, ω_T and R_T also casually denote the rotation component of T in the group of rigid transformations, called the *special Euclidean group* $SE(3)$.

Lemma 1. *Let $\omega_{A_{12}} = Log(R_{A_{12}})$ and $\omega_{B_{12}} = Log(R_{B_{12}})$ for given $A_{12}, B_{12} \in SE(3)$. If an equation $A_{12}X = XB_{12}$ is to hold for some $X \in SE(3)$, then it is necessary that $||\omega_{A_{12}}|| = ||\omega_{B_{12}}||$.*

Proof: In this proof, we use the well-known properties without proofs [25], [26]: (i) $\exp(R[\omega]_\times R^\top) = R\exp([\omega]_\times)R^\top$ and (ii) $R[\omega]_\times R^\top = [R\omega]_\times$ for $R \in SO(3)$ and $\omega \in \mathbb{R}^3$. From the rotational constraint in Equation 3 and by the property (i), $R_{A_{12}} = R_X R_{B_{12}} R_X^\top = R_X \exp([\omega_{B_{12}}]_\times) R_X^\top = \exp(R_X[\omega_{B_{12}}]_\times R_X^\top)$. Then, if we take log on both sides, $[\omega_{A_{12}}]_\times = R_X[\omega_{B_{12}}]_\times R_X^\top = [R_X\omega_{B_{12}}]_\times$ by the property (ii). Since R_X is an orthogonal matrix, the condition $\omega_{A_{12}} = R_X\omega_{B_{12}}$ implies that $||\omega_{A_{12}}|| = ||\omega_{B_{12}}||$, demanding the rotational angles of A_{12} and B_{12} should be the same. ■

Lemma 2. *Given $A_{12}, B_{12} \in SE(3)$, let $\omega_{A_{12}} = Log(R_{A_{12}})$ and $\omega_{B_{12}} = Log(R_{B_{12}})$. Then, the equation $A_{12}X = XB_{12}$ has infinitely many solutions $X \in SE(3)$ if $||\omega_{A_{12}}|| = ||\omega_{B_{12}}||$.*

Proof: The necessary condition of $||\omega_{A_{12}}|| = ||\omega_{B_{12}}||$ is obvious from **Lemma 1**. Assume $X_1 \in SE(3)$ with (R_{X_1}, t_{X_1}) be a solution of the equation $A_{12}X = XB_{12}$, meaning $R_{A_{12}}R_{X_1} = R_{X_1}R_{B_{12}}$. Also, consider $X_2 \in SE(3)$ with $R_{X_2} \triangleq \exp([r\bar{\omega}_{A_{12}}]_\times)R_{X_1}\exp([s\bar{\omega}_{B_{12}}]_\times)$ for twe angles $s, r \in (0, 2\pi)$. Then, using the two properties and the fact $\omega_{A_{12}} = R_X\omega_{B_{12}}$, used and proven in **Lemma 1**, we find that

$$\begin{aligned} R_{X_2} &= R_{X_1}R_{X_1}^\top R_{X_2} \\ &= R_{X_1}\{R_{X_1}^\top \exp([r\bar{\omega}_{A_{12}}]_\times R_{X_1})\}\exp([s\bar{\omega}_{B_{12}}]_\times) \\ &= R_{X_1}\exp(R_{X_1}^\top[r\bar{\omega}_{A_{12}}]_\times R_{X_1})\exp([s\bar{\omega}_{B_{12}}]_\times) \\ &= R_{X_1}\exp([R_{X_1}^\top(r\bar{\omega}_{A_{12}})]_\times)\exp(s[\bar{\omega}_{B_{12}}]_\times) \\ &= R_{X_1}\exp([r\bar{\omega}_{B_{12}}]_\times)\exp(s[\bar{\omega}_{B_{12}}]_\times) \\ &= R_{X_1}\exp([(r+s)\bar{\omega}_{B_{12}}]_\times). \end{aligned}$$

Now, by substituting the last expression of R_{X_2} in $R_{X_2}R_{B_{12}}R_{X_2}^\top$ and using the simple fact that $[\omega]_\times$ is $\exp([t\bar{\omega}]_\times)[\omega]_\times\exp([t\bar{\omega}]_\times)^\top$ for any $t \in [0, 2\pi]$ and $\omega \in \mathbb{R}^3$, it is trivial to see that $R_{X_2}R_{B_{12}}R_{X_2}^\top = R_{X_1}R_{B_{12}}R_{X_1}^\top$, which is equal to $R_{A_{12}}$ because $R_{A_{12}}R_{X_1} = R_{X_1}R_{B_{12}}$. Thus, the fact that $R_{X_2}R_{B_{12}}R_{X_2}^\top = R_{A_{12}}$ shows that X_2 with (R_{X_2}, t_{X_2}) is also a solution of $A_{12}X = XB_{12}$ for any $s, r \in (0, 2\pi)$. □

Theorem. *Given $T = A_{12}, A_{23}, B_{12}, B_{23} \in SE(3)$, let $\omega_T = Log(R_T)$. Then, the system of equations $A_{12}X = XB_{12}$ and $A_{23}X = XB_{23}$ has the unique solution $X \in SE(3)$ if (i) $||\omega_{A_{12}}|| = ||\omega_{B_{12}}||$, (ii) $||\omega_{A_{23}}|| = ||\omega_{B_{23}}||$, and (iii) $\omega_{A_{12}} \times \omega_{A_{23}} \neq 0$ and $\omega_{B_{12}} \times \omega_{B_{23}} \neq 0$.*

Proof: The first two necessary conditions are obvious from **Lemma 1**. First, consider the rotational part R_X of X. As shown in the first part of the proof of **Lemma 2**, the constraints in Equation 5 imply $R_X\omega_{B_{12}} = \omega_{A_{12}}$ and

$R_X\omega_{B_{23}} = \omega_{A_{23}}$. Furthermore, using the property that R_X is a rotation matrix, we get $\omega_{A_{12}} \times \omega_{A_{23}} = (R_X\omega_{B_{12}}) \times (R_X\omega_{B_{23}}) = \det(R_X)(R_X^{-1})^\top(\omega_{B_{12}} \times \omega_{B_{23}}) = R_X(\omega_{B_{12}} \times \omega_{B_{23}})$. Then, since both $\omega_{A_{12}} \times \omega_{A_{23}}$ and $\omega_{B_{12}} \times \omega_{B_{23}}$ are nonzero, we can find a unique R_X from the following nonsingular system of equations:

$$R_X\left[\omega_{B_{12}}\omega_{B_{23}}(\omega_{B_{12}} \times \omega_{B_{23}})\right] = \left[\omega_{A_{12}}\omega_{A_{23}}(\omega_{A_{12}} \times \omega_{A_{23}})\right].$$

Second, for the translational part t_X of X, the constraints in Equation 6 are rephrased as the 6×3 linear system $Ct_X = b$ with unknown t_X where

$$C = \begin{bmatrix} R_{A_{12}} - I \\ R_{A_{23}} - I \end{bmatrix} \in \mathbb{R}^{6\times 3} \text{ and } b = \begin{bmatrix} R_X\boldsymbol{t}_{B_{12}} - t_{A_{12}} \\ R_X\boldsymbol{t}_{B_{23}} - t_{A_{23}} \end{bmatrix} \in \mathbb{R}^3.$$

To prove the theorem, it is sufficient to show that the rank of C is 3 because it guarantees the unique solution t_X, and thus the unique solution X of the theorem. To investigate the rank of C, let Q_{12} and Q_{23} be two rotation matrices that transform the unit vector $(0, 0, 1)^\top$ to the rotational axes $\bar{\omega}_{A_{12}}$ of $R_{A_{12}}$ and $\bar{\omega}_{A_{23}}$ of $R_{A_{23}}$, respectively. (Note that the third columns of Q_{12} and Q_{23} are $\bar{\omega}_{A_{12}}$ and $\bar{\omega}_{A_{23}}$, respectively.) Then, for the 3×3 rotation matrix $R_z(\phi)$ corresponding to the rotation around the z-axis by angle ϕ, we have $R_{A_{12}} = Q_{12}R_z(\phi_{12})Q_{12}^\top$ and $R_{A_{23}} = Q_{23}R_z(\phi_{23})Q_{23}^\top$, where $\phi_{12} \triangleq ||\omega_{A_{12}}||$ and $\phi_{23} \triangleq ||\omega_{A_{23}}||$ are the respective rotational angles of $R_{A_{12}}$ and $R_{A_{23}}$. Then, since $C^\top C = 4I - (R_{A_{12}}^\top + R_{A_{12}}) - (R_{A_{23}}^\top + R_{A_{23}})$, we have $Q_{12}^\top(C^\top C)Q_{12} = 4I - (R_z(\phi_{12})^\top + R_z(\phi_{12})^\top) - Q_{12}^\top Q_{23}\{R_z(\phi_{23})^\top + R_z(\phi_{23})\}Q_{23}^\top Q_{12}$. If we let $c_{12} \triangleq \cos(\phi_{12})$, $c_{23} \triangleq \cos(\phi_{23})$, and $n \triangleq (n_x, n_y, n_z)^\top$, where n is the third column of $Q_{12}^\top Q_{23}$, we have

$$\begin{aligned} Q_{12}^\top(C^\top C)Q_{12} &= 4I - 2\begin{bmatrix} c_{12} & 0 & 0 \\ 0 & c_{12} & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ &\quad - 2\,c_{23}I - 2(1 - c_{23})nn^\top \\ &= \begin{bmatrix} \alpha - \beta n_x^2 & -\beta n_x n_y & -\beta n_x n_z \\ -\beta n_x n_y & \alpha - \beta n_y^2 & -\beta n_y n_z \\ -\beta n_x n_z & -\beta n_y n_z & \beta(1 - n_z^2) \end{bmatrix}, \end{aligned}$$

where $\alpha = 4 - 2c_{12} - 2c_{23}$ and $\beta = 2 - 2c_{23}$. From the last matrix, a little calculation shows that $\det(Q_{12}^\top(C^\top C)Q_{12}) = \alpha\beta(\alpha - \beta)(1 - n_z^2)$. Note that the property of $\omega_{A_{12}} \times \omega_{A_{23}} \neq 0$ implies both $\omega_{A_{12}} \neq 0$ and $\omega_{A_{23}} \neq 0$. That is, $c_{12} < 1$ and $c_{23} < 1$ because $\phi_{12} > 0$ and $\phi_{23} > 0$, implying α, β, and $\alpha - \beta$ are all nonzero. Furthermore, n_z is the inner product of $\bar{\omega}_{A_{12}}$ and $\bar{\omega}_{A_{23}}$, which are the third columns of Q_{12} and Q_{23}, respectively, also implying $n_z^2 \neq 1$ since $\omega_{A_{12}} \times \omega_{A_{23}} \neq 0$. Therefore, $\det(Q_{12}^\top(C^\top C)Q_{12}) \neq 0$ and the 3×3 matrix $C^\top C$ has full rank because $\det(C^\top C) = \det(Q_{12}^\top)\det(C^\top C)\det(Q_{12}) = \det(Q_{12}^\top(C^\top C)Q_{12})$. Finally, from the fact that $3 = \operatorname{rank}(C^\top C) \leq \min\{\operatorname{rank}(C^\top), \operatorname{rank}(C)\} = \operatorname{rank}(C)$, we see that the rank of C is 3. This completes the proof of the theorem. □

REFERENCES

[1] Meta. (2023). *Use Passthrough on Meta Quest*. [Online]. Available: https://www.meta.com/help/quest/articles/in-vr-experiences/oculus-features/passthrough/

[2] D. Gottlieb, "Mixing reality with virtual reality," Jan. 2018. [Online]. Available: http://drewgottlieb.net/2017/01/31/mixing-reality-with-vr.htm

[3] J. Gugenheimer, E. Stemasov, J. Frommel, and E. Rukzio, "ShareVR: Enabling co-located experiences for virtual reality between HMD and non-HMD users," in *Proc. CHI Conf. Human Factors Comput. Syst.*, May 2017, pp. 4021–4033.

[4] W. Chun, G. Choi, J. An, W. Seo, S. Park, and I. Ihm, "On sharing physical geometric space between augmented and virtual reality environments," in *Proc. IEEE Conf. Virtual Reality 3D User Interfaces (VR)*, Mar. 2019, pp. 884–885.

[5] J. G. Grandi, H. G. Debarba, and A. Maciel, "Characterizing asymmetric collaborative interactions in virtual and augmented realities," in *Proc. IEEE Conf. Virtual Reality 3D User Interfaces (VR)*, Mar. 2019, pp. 127–135.

[6] J. S. Roo and M. Hachet, "One reality: Augmenting how the physical world is experienced by combining multiple mixed reality modalities," in *Proc. 30th Annu. ACM Symp. User Interface Softw. Technol.*, Oct. 2017, pp. 787–795.

[7] T. Piumsomboon, A. Day, B. Ens, Y. Lee, G. Lee, and M. Billinghurst, "Exploring enhancements for remote mixed reality collaboration," in *Proc. SIGGRAPH Asia Mobile Graph. Interact. Appl.*, Nov. 2017, p. 16.

[8] J. An, G. Choi, W. Chun, Y. Joo, S. Park, and I. Ihm, "Accurate and stable alignment of virtual and real spaces using consumer-grade trackers," *Virtual Reality*, vol. 26, no. 1, pp. 125–141, Mar. 2022.

[9] Y. Cho, M. Park, and J. Kim, "XAVE: Cross-platform based asymmetric virtual environment for immersive content," *IEEE Access*, vol. 11, pp. 71890–71904, 2023.

[10] J. S. Roo and M. Hachet, "Towards a hybrid space combining spatial augmented reality and virtual reality," in *Proc. IEEE Symp. 3D User Interfaces (3DUI)*, Mar. 2017, pp. 195–198.

[11] E. Azimi, L. Qian, N. Navab, and P. Kazanzides, "Alignment of the virtual scene to the tracking space of a mixed reality head-mounted display," 2017, *arXiv:1703.05834*.

[12] H. Bai, L. Gao, and M. Billinghurst, "6DoF input for hololens using vive controller," in *Proc. SIGGRAPH Asia Mobile Graph. Interact. Appl.*, Bangkok, Thailand. New York, NY, USA: Association for Computing Machinery, Nov. 2017, Art. no. 4:1. [Online]. Available: https://doi.org/10.1145/3132787.3132814

[13] T. Weissker, P. Tornow, and B. Froehlich, "Tracking multiple collocated HTC vive setups in a common coordinate system," in *Proc. IEEE Conf. Virtual Reality 3D User Interfaces Abstr. Workshops (VRW)*, Mar. 2020, pp. 592–593.

[14] D. Reimer, I. Podkosova, D. Scherzer, and H. Kaufmann, "Colocation for SLAM-tracked VR headsets with hand tracking," *Computers*, vol. 10, no. 5, p. 58, Apr. 2021.

[15] Y. C. Shiu and S. Ahmad, "Calibration of wrist-mounted robotic sensors by solving homogeneous transform equations of the form AX=XB," *IEEE Trans. Robot. Autom.*, vol. 5, no. 1, pp. 16–29, Feb. 1989.

[16] F. C. Park and B. J. Martin, "Robot sensor calibration: Solving AX=XB on the Euclidean group," *IEEE Trans. Robot. Autom.*, vol. 10, no. 5, pp. 717–721, Oct. 1994.

[17] A. Li, L. Wang, and D. Wu, "Simultaneous robot-world and hand-eye calibration using dual-quaternions and Kronecker product," *Int. J. Phys. Sci.*, vol. 5, no. 10, pp. 1530–1536, 2010.

[18] M. Shah, R. D. Eastman, and T. Hong, "An overview of robot-sensor calibration methods for evaluation of perception systems," in *Proc. Workshop Perform. Metrics Intell. Syst.*, Mar. 2012, pp. 15–20.

[19] A. Tabb and K. M. Ahmad Yousef, "Solving the robot-world hand-eye(s) calibration problem with iterative methods," *Mach. Vis. Appl.*, vol. 28, nos. 5–6, pp. 569–590, Aug. 2017.

[20] Z. Zhang, L. Zhang, and G.-Z. Yang, "A computationally efficient method for hand–eye calibration," *Int. J. Comput. Assist. Radiol. Surgery*, vol. 12, no. 10, pp. 1775–1787, Oct. 2017.

[21] W. Kabsch, "A solution for the best rotation to relate two sets of vectors," *Acta Crystallographica Sect. A*, vol. 32, no. 5, pp. 922–923, Sep. 1976.

[22] S. Umeyama, "Least-squares estimation of transformation parameters between two point patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 4, pp. 376–380, Apr. 1991.

[23] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-D point sets," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-9, no. 5, pp. 698–700, Sep. 1987.

[24] A. Nádas, "Least squares and maximum likelihood estimates of rigid motion," IBM Thomas J. Watson Res. Division, Yorktown Heights, NY, USA, Tech. Rep., RC6945, 1978.

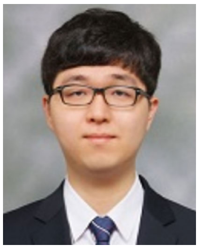
[25] J. Claraco, "A tutorial on SE(3) transformation parameterizations and on-manifold optimization," Dpto. de Ingeniería de Sistemas y Autom atica, Univ. de Malága, Malága, Spain, Tech. Rep., #012010, 2018.

[26] J. Sola, J. Deray, and D. Atchuthan, "A micro lie theory for state estimation in robotics," 2018, *arXiv:1812.01537*.

[27] D. C. Niehorster, L. Li, and M. Lappe, "The accuracy and precision of position and orientation tracking in the HTC vive virtual reality system for scientific research," *i-Perception*, vol. 8, no. 3, Jun. 2017, Art. no. 204166951770820.

[28] E. Luckett, T. Key, N. Newsome, and J. A. Jones, "Metrics for the evaluation of tracking systems for virtual environments," in *Proc. IEEE Conf. Virtual Reality 3D User Interfaces (VR)*, Mar. 2019, pp. 1711–1716.

[29] L. G. Sansone, R. Stanzani, M. Job, S. Battista, A. Signori, and M. Testa, "Robustness and static-positional accuracy of the SteamVR 1.0 virtual reality tracking system," *Virtual Reality*, vol. 26, no. 3, pp. 903–924, Sep. 2022.



JAEPUNG AN received the B.E. and M.E. degrees from Sogang University, Seoul, South Korea, in 2012 and 2014, respectively, where he is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering. His research interests include computational photography, real-time rendering, and GPU computing.

JOO HO LEE received the Ph.D. degree in computer science from KAIST, in 2020. He is an Assistant Professor with Sogang University and supervises the Visual Computing Laboratory. He was a Postdoctoral Researcher with the University of Tuebingen and the Max Planck Institute (before). His research interests include computer graphics, 3-D reconstruction, and computer vision. He served as a Reviewer for conference programs, such as SIGGRAPH and CVPR.

SANGHUN PARK received the B.S. degree in mathematics and the M.E. and Ph.D. degrees in computer science from Sogang University, Seoul, south Korea, in 1993,1995, and 2000, respectively. After the Ph.D. work, he was a Research Staff Member with the Computational Visualization Center, Institute for Computational Engineering and Sciences, The University of Texas at Austin, USA. He is currently a Professor with the Graduate School of Metaverse, Sogang University. Before joining Sogang University, he was a Professor with the Department of Multimedia, Graduate School of Digital Image and Contents, Dongguk University, Seoul. His research interests include computer graphics, extended reality, and high performance computing.

INSUNG IHM received the B.S. degree in computer science and statistics from Seoul National University, Seoul, South Korea, in 1985, the M.S. degree in computer science from Rutgers University, NJ, USA, in 1987, and the Ph.D. degree in computer science from Purdue University, IN, USA, in 1991. He is currently a Professor of computer science and engineering with Sogang University, Seoul. His research interests include real-time 3-D graphics, GPU computing, and extended reality.