A Rate-based Drone Control with Adaptive Origin Update in Telexistence

Di Zhang
Nanjing Univ. of Posts and Tele.

Chi-Man Pun
University of Macau

Yang Yang
Nanjing Univ. of Posts and Tele.

Hao Gao*
Nanjing Univ. of Posts and Tele.

Feng Xu†
Tsinghua University

Figure 1: A telexistence drone is controlled by a user with our improved rate-based control mechanism. The system provides immersive viewing experience to the user in real time. As the user moves his head, the drone moves accordingly and transmits the recorded frames back to the HMD for the user to view.

ABSTRACT

A new form of telexistence is achieved by recording videos with a camera on an Uncrewed aerial vehicle (UAV) and playing the videos to a user via a head-mounted display (HMD). One key problem here is how to let the user freely and naturally control the UAV and thus the viewpoint. In this paper, we develop an HMD-based telexistence technique that achieves full 6-DOF control of the viewpoint. The core of our technique is an improved rate-based control technique with our adaptive origin update (AOU), in which the origin of the coordinate system of the user changes adaptively. This makes the user naturally perceive the origin and thus easily perform the control motion to get his/her desired viewpoint changing. As a consequence, without the aid of any auxiliary equipment, the AOU scheme handles the well known self-centering problem in the rate-based control methods. A real prototype is also built to evaluate this feature of our technique. To explore the advantage of our telexistence technique, we further use it as an interactive tool to perform the task of 3D scene reconstruction. User studies demonstrate that comparing with other telexistence solutions and the widely used joystick-based solutions, our solution largely reduces the workload and saves time and moving distance for the user.

1 INTRODUCTION

Telexistence allows users to view and interact in remote environments rather than their current locations [39, 40]. As it offers real-time and real-world sense of presence, telexistence can be applied to various applications, including social interaction, virtual traveling, remote control and so on [15, 18]. One key requirement in telexistence is to freely navigate and perceive the remote scene. Uncrewed aerial vehicles (UAVs) could facilitate this as they have sufficient motion freedom and the capacity of monitoring extensive spaces. As a consequence, researchers are interested in developing UAV-based telexistence systems. Another key issue here is to allow users naturally control the navigation, which is still a difficult problem. At present, joystick-based user interfaces are commonly used in UAV-based telexistence systems [23]. Compared with joystick-based user interfaces, an HMD-based telexistence system not only frees users' hands but also provides natural control for users. Furthermore, compared with a tedious 2D display system, an HMD plays stereo videos [4, 37], which contributes to the sense of immersion for users.

In this paper, we manipulate a UAV to perform telexistence via an HMD (Fig. 1). As the moving space of a user is always limited, asynchronous (position-control) mapping strategy [30], which typically conveys the displacement of the manipulator to the remote displacement, confines the movement range in the remote scene. Translation-to-velocity (rate-control) strategy does not have this limitation. But self-centering mechanism [44], which aims to let the user perceive the origin of the control system, is difficult to be accurately and robustly achieved. In this paper, we propose an adaptive origin update (AOU) scheme to dynamically change the coordinate system of the user in the rate control, which makes it easier and more precise to control the flying states of the UAV. Besides evaluating our technique on the quality of user experiences [1], we also explore new applications enabled by our technique. To be specific, we apply our HMD-based telexistence system to 3D reconstruction, i.e., acquiring the 3D model of a scene in real-time during the flight of a UAV. Based on our knowledge, this is for the first time to use an HMD-based telexistence system as an interactive tool in the process of 3D scene reconstruction. With an onboard stereo camera, aerial images can be recorded constantly and a 3D map can be built online from the image sequences [11]. Based on this, we combine our telexistence system with depth sensing and fusion techniques [11] to achieve real-time 3D reconstruction with free and natural control.

To validate our rate-based control technique, we first build a simulation system that provides an ideal environment without delays, interferes, and errors. Based on this system, we compare our method with the existing HMD-based interfaces by sophisticated user studies. The comparison results reveal that our method provides a natural and efficient interactive mechanism for users to complete the designed tasks. Furthermore, another simulation system is designed to evaluate our telexistence in the application of real-time 3D reconstruction. The traditional joystick-based drone control is also compared here. The results demonstrate that on average, our method spends less time, moving distance, and workload in controlling the UAV for 3D reconstruction while a high-quality reconstruction is maintained. Finally, a real prototype is built to perform telexistence to users. A real 3D reconstruction of a statue is also performed to further evaluate the practicality of our technique.

*e-mail: tsgaohao@gmail.com
†e-mail: xufeng2003@gmail.com

2 Related Works

In this section, we first discuss the controlling and rate-based control techniques in the existing telexistence systems. Second, data collected interaction methods for 3D reconstruction have been described in detail.

2.1 Telexistence with UAV control

Body-machine interfaces (BoMIs) have been widely used to control UAVs. Guo et al. [12] suggested a gesture-based scheme with tangible user interfaces for human-robot interaction (HRI). Sanna et al. [34] employed a Kinect to estimate the body motion of a user and used the motion to control the flying of a UAV. Pfeil et al. [28] defined different controlling mechanisms based on the 3D gestures of the upper body of a user. The mechanisms were well compared with a sophisticated study. Park et al. [26] also utilized body motion for control. However, they made use of a motion capture system to acquire the body motion and a haptic glove for hand manipulation. Recently, Fern´andez et al. [10] integrated speech, body position, hand gesture, and visual marker interactions to obtain a natural and free control of UAVs. However, all the aforementioned techniques only perform the control but do not provide telexistence sense to users.

For telexistence, an HMD not only provides immersive first-person viewing experiences but also helps to control the UAVs. Some works integrate HMD with other equipment as a control tool. Cherpilled et al. [9] developed a birdly interaction system to enhance the immersive sense of users in VR. The user laid on a platform and took control of the flight of the UAV by changing their gestures. In [33], a soft exoskeleton teleoperation system was introduced as a human-drone interface. A simple FlyJacket equipment was armed with the user's arm for teleoperation with a UAV. Herrmann et al. [16] proposed a mix interface, which contained gesture and speech, was proposed. In the system, the gesture-based interface was the main interface. Besides the methods based on deterministic rules, a data-driven technique was also proposed to utilize the motion of the upper body to achieve telexistence [22]. These techniques may afford good experiences of telexistence, but they require complex equipment to build the system. For "Head Translation", proposed by Bowman Doug A. and Hodges Larry F. [5], although it allows the operator to control the aircraft in a limited space. However, due to the lack of AOU, the operator will move far away from the operating point until he is out of range of the sensors. Some previous works along with this work focus on only using the HMD to control a UAV in the telexistence scenario. K. Higuchi and J. Rekimoto [17] proposed a "Flying Head" control mechanism for a UAV. Their system utilized an HMD to synchronize the motion between the manipulator and the UAV. The movement of the UAV is aligned with the body motion of the manipulator. However, due to the direct mapping mechanism, the moving space of the UAV is limited.

Pittman et al. [29]investigated robot navigation with HMD tracking. In this technique, when users go back to their original position, the UAV will hover. But according to our experiment, when the users are equipped with an HMD, it is very difficult for them to perceive their original position precisely. S. Wang et al. [42] used "Head Rotation" to control the motion of the UAV. It is easy to control the flight of the UAV but is difficult to control the viewing direction as it is natural to use head rotation to change the viewing direction. "Head Translation" proposed by D. A. Bowman et al. [6] is a rate-control method. As it is a distance-to-velocity mapping, the aircraft is able to move in an unlimited space in theory. However, due to the well-known self-sensing problem, users may wrongly sense the origin, leading to unexpected flying direction of the aircraft.

2.2 3D reconstruction

By employing a depth camera and modern technology, S. Orts-Escolano and C. Rhemann [25] propose an Holoportation system to interact with remote participants in 3D with high-quality, real-time 3D reconstruction. By using a color and depth camera to capture the 3D pose of a local user and projecting their virtual copy into remote space [27], the authors proposed a real size telepresence system using 3D reconstruction technology. But the reconstructed object in their system is surrounded by the depth camera. The process of 3D reconstruction in [25, 27] depends on fixed camera. Therefore, only small scene could be reconstructed. For large-scale outdoor reconstruction, fast reconstruction method with feedback on mobile device is proposed by T. Schöps and T. Sattler [35] . B. Reitinger et al. [32] proposed an interactive 3D reconstruction AR system. In their system, a person is equipped with a handheld AR device to capture images, and the 3D model is generated on-the-fly for inspection on the device. In the two papers, for utilizing a mobile device, the users only reconstruct the local area. Then they couldn't immerse in remote scene. Zhexiong Shang and Zhigang Shen proposed a real-time UAV reconstruction system [36]. The reconstruction results are shown on the screen of ground station. Then the drone is controlled by a joystick to fly to an unreconstructed area. But their observation angle of the reconstructed results is not same as the drone's first view. The operator does not get natural feedback to fill the un-reconstructed area by joystick. In view of this, we apply our HMD based telexistence system with 6-DOF mechanism into 3D reconstruction. Therefore, users could naturally participate into the process of remote 3D reconstruction which would enhance the quality of results.

3 Methods

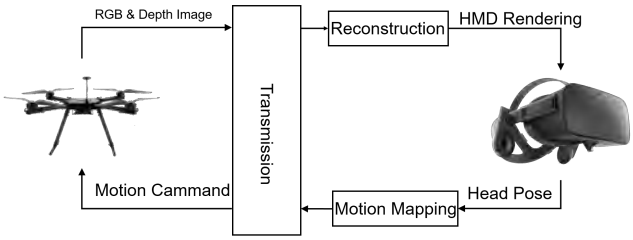

Figure 2: Our HMD-based UAV telexistence system.

3.1 System Overview

The architecture of our HMD-based telexistence drone system is shown in Fig. 2. First, we develop a customized quadcopter with a stereo depth camera and a transmission module. The gimbal used to carry the stereo camera gives extra 2 DOFs (pitch and roll), and the rest 4 DOFs (three translations and the yaw rotation) are provided by the UAV. So we have the full 6-DOF to achieve any desired viewpoints in telexistence. The ground station receives RGB and depth images transferred from the UAV via a WiFi connection, and reconstructs the 3D geometry of objects in the scene in real time. To visualize the scanning progress, RGB images blended with the reconstructed model surface are shown in the HMD. Then, the HMD transfers the motion of the user's head to the ground station for generating the control signals of the UAV by the improved rate-based control technique. Finally, the signals are transmitted to the UAV based on the MAVLink protocol via a digital video downlink. We illustrate the designed control method in Sect. 3.2 which includes a basic mapping technique mechanism, our AOU scheme, and our full viewpoint control. Discussions on the 3D reconstruction mechanism are presented in Sect. 3.3.

3.2 Control Mechanism

To realize an intuitive and natural control of a UAV by a user, we adopt the rate-control mechanism to perform omni-directional flight

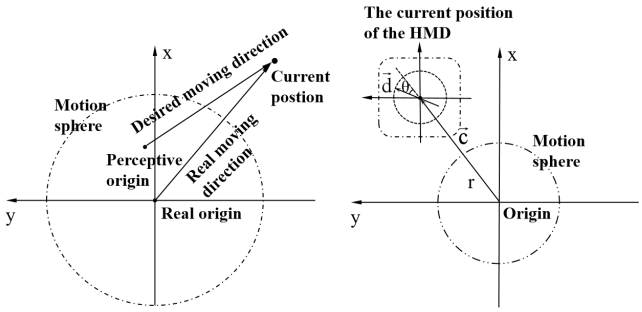

Figure 3: Illustration of our rate-based control technique. In the left image, the origin is the initial position of the user. $\vec{c}$ is the vector from the origin to the user's current position and $\vec{d}$ indicates the viewing direction of the user. Notice that this is the top view of the real 3D scene. The right illustrates the motion mapping errors caused by using a fixed origin. In some cases, the user may move to the *perceptive origin* and treat it as the *real origin*. Then, when the user moves to the *current position*, the *real moving direction* will be different from the *desired moving direction*.

with free movement space. An AOU mechanism is proposed to overcome the lack of self-centering [44] in the rate-control mechanism. Besides, our technique realizes 6-DOF viewpoint control with a rotation mapping strategy by utilizing a gimbal.

3.2.1 Basic Control Mechanism

As the volume of the user's movement and the tracking range of the sensor (Oculus Rift) are both restricted, a UAV in a telexistence system could not realize an unlimited flight by position-based control, even with a non-linear mapping function [5, 30]. It seems that the rate-based control mechanism, which maps the displacement of the input device to the velocity of a UAV, is applicable for freely flight in space. Thus we employ this mechanism following [43].

Hovering Mechanism In a traditional rate-based control system, when a user stays still at the origin and looks around, a slight position offset of the head is inevitable, which causes the motion of the UAV.

To avoid this, within a predefined distance r from the origin, we only map the rotation of the HMD to that of the camera and keep the UAV hovering. If the distance between the HMD and the center is larger than r, the UAV will move with the velocity determined by the displacement. When the user moves back to the circle from outside, the UAV will stop moving and keep hovering. In all our experiments, we set r to be 20cm, which has been tuned by the trials of users.

A top view to illustrate our rate-based control mechanism is shown in Fig. 3. We set the initial position of the HMD as the origin of the coordinate system. The physical circle with the predefined radius r is also centered at the origin. The rotation on the yaw direction of the HMD is directly mapped to the yaw of the UAV, which is denoted as a unit vector $\vec{d}$. Notice that the rotation on the other two directions are not considered by the existing rate-base control techniques while we take it into consideration and will introduce it in Sect. 3.2.3. The velocity of UAV is determined by the position vector $\vec{c}$ which starts from the origin and ends at the HMD's current position. To be specific, if $|\vec{c}| > r$, the velocity of the UAV in pitch (forward) and roll (left) directions are calculated as

$$\vec{v_p} = k * (|\vec{c}| - r) * \cos\theta; \quad \vec{v_r} = k * (|\vec{c}| - r) * \sin\theta. \qquad (1)$$

This achieves that the UAV faces the direction of $\vec{d}$ while flying along the direction of $\vec{c}$ with the speed $k * (|\vec{c}| - r)$. The factor of the gain control is decided by k. k is set to 1 to balance the usability and the accuracy of the control. The motion of the UAV on the vertical direction is controlled by whether the user is standing on his/her toes or squatting down. We also employ the rate-based control mechanism with a trigger threshold here so that the user can freely adjust the vertical speed.

3.2.2 Adaptive Origin Update (AOU)

According to the description of the basic control mechanism, the origin is very important for controlling the flying direction of the UAV. Some previous works of 3D interaction have indicated that the rate-based control is inappropriate for devices without a self-centering mechanism [44]. For example, when stepping back into the circle from outside, it is difficult for the user to perceive the origin precisely, especially when he/she has stayed outside the circle for a long time. Fig. 3 gives an illustration of the control error. This drawback could be overcome by a clutching mechanism, haptic aids or visual hints but all requiring auxiliary setups.

Even though it is difficult to perceive the origin, the user could always perceive a rough direction of the physical circle area from the displayed images in the HMD according to the sense of the flying direction of the UAV. Based on this observation, we propose an adaptive origin updating (simplified as AOU) scheme. If the user steps back inside the circle and stays still for more than 2 seconds, we set the current position of the HMD as the origin. Additionally, we extend this scheme to a braking mechanism which means if the user moves far away from the physical circle, the origin is allowed to be updated as soon as the user quickly stepping towards the inverse flying direction of the UAV, even though the user has not come back into the circle. This could handle some emergencies once the user loses control in the case of high speed flying.

Our AOU scheme brings a problem that, in extreme cases, the user may step out of the sensing range if the origin is gradually updated to the boundary of the sensing range. Currently, we involve a restart mechanism that allows the user to set the origin of the motion sphere without changing the position of the UAV. When the user is approaching the sensing boundary, the system will make an alarm and the restart mechanism will be activated to let the user set a new origin.

3.2.3 Full Viewpoint Control

To achieve 6-DOF, in addition to the 4-DOF motion provided by the drone itself, we place a gimbal on UAV to map the pitch and roll of the user's HMD to that of the camera on the gimbal. Notice that for controlling rotations, we do not use the rate-based control mechanism as mapping translations because rotations do not suffer the space limitation as translations, and a direct mapping is more natural for users.

Joystick-based telexistence systems could also realize 6-DOF by adding a gimbal. However, users need to coordinate the direction of the UAV and the gimbal to make the cameras focusing on targets as shown in Fig. 4. And uses have to use different fingers to control the 2-DOF of the gimbal and the 4-DOF of the UAV to focus on a target during the fast movement of UAV. This is shown in our accompanying video. Since we directly use body and head motion to control the UAV and gimbal by the improved rate-based control technique, the task of focusing on targets could be completed more naturally.

3.3 3D Reconstruction with Our System

We follow the popular volumetric fusion method for dense and real-time 3D scanning. It consists of three steps: depth estimation, camera pose estimation, and depth fusion. In our system, the online reconstructed 3D geometry will be visualized as the guidance in the HMD to help the user perform the scanning. As utilizing a depth sensor, depth maps are directly captured. To fuse the depth data into a single global surface model, the sensor pose is estimated based on the photometric and geometric constraints. We develop a

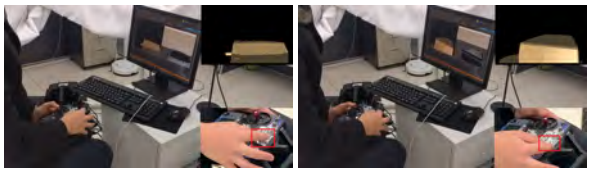

Figure 4: An illustration of the difficulty in the joystick control mechanism in telexistence mechanism.

simulation system and a real prototype to perform UAV-based 3D reconstruction.

3.3.1 Camera Pose Estimation

Given a 3D model reconstructed by fusing previous depth frames, we estimate the camera pose by the new depth frame. First, we project the 3d model to the camera pose of the previous frame and get a 3D surface. The new depth data is aligned with the extracted surface using an iterative closest point (ICP) algorithm with projective data association [8], which outputs the 6-DOF relative camera pose. We use this method in our simulation system. But for our real prototype, since the magnitude of the noise associated with the recorded depth data is too large, the depth-based tracker in [8] cannot work well. VINS-Mono [31], a real-time simultaneous localization and mapping (SLAM) framework with the ability of IMU pre-integration, is used to estimate the camera motion.

3.3.2 Depth Fusion

KinectFusion-based approaches could fuse depth into a single volumetric 3D surface model, but the size of the reconstructed scene is limited. For mixed reality and robot navigation scenarios, reconstruction scope should not be limited in a fixed space. To this end, works such as hash-based method [24] and octree technique [7] which compress space by using more efficient data structures have emerged. We adopt voxel hashing method [24] for depth fusion.

3.3.3 Reconstruction Visualization

In our system, the HMD is used not only to control drones naturally, but also to observe 3D reconstructed scenes in real time. So we want to render the online reconstructed geometry with the recorded images. Then the users could control the UAV to move to record the unreconstructed parts though an ideal path. In this way, we could naturally perform a complete 3D reconstruction.

An example of what users see in our HMD is shown in Fig. 5. To make the reconstructed surface noticeable, we color-code the reconstructed surface depth and overlap the color to the images. Specifically, given the current camera pose, we get a depth map of the visible surface by raycasting the fused model and convert the depth map to HSV (Hue, Saturation, Value) color space following [38]. The depth map in the form of HSV is then converted to RGB and blended with the current RGB image. In this way, the reconstructed regions will show in different colors and thus can be easily distinguished from other unreconstructed regions.

4 Experiments

In the literature, there are five representative gestural navigation techniques and we will compare them with our solution. We first give an overview of them in Sect. 4.1 and compare our method with them by the simulated telexistence system in Sect. 4.2. To further evaluate our technique in the application of 3D reconstruction, we conduct another user study in the simulated 3D reconstruction system in Sect. 4.3. Joystick is also added as a compared algorithm as it is widely used as an operating tool in 3D reconstruction. Our real prototype is introduced and demonstrated in Sect. 4.4. Finally, we discuss the limitations of our current technique in Sect. 4.5.

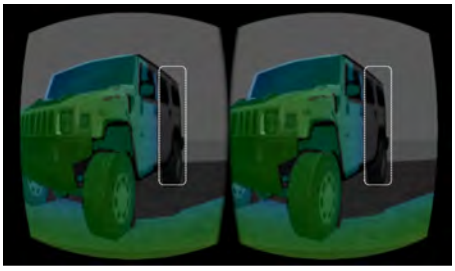

Figure 5: An example of images displayed in the HMD for the application of 3D reconstruction. The user can clearly distinguish the regions with the reconstructed geometry (the front part of the car) and the regions that have not been reconstructed (the back part of the car), and thus he/she could easily control the UAV to fill the missing regions.

4.1 Existing Interaction Techniques

We investigate five existing HMD-based interaction techniques including **Head Translation** (**HT**) [6], **Head Rotation** (**HR**) [42], **Flying Head** (**FH**) [17], **Modified Flying Head** (**MFH**) [29] and **Flying Plane** (**FP**) [29]. They are intuitively shown in Table 1. Notice that these techniques only provide the basic controlling solutions and they do not have our hovering mechanism or AOU to help users naturally control the UAV.

HT(Head Translation) maps the translation of the head to the speed of the UAV which is similar to ours. To turn the UAV, the user turn his/her head left or right and the UAV will keep rotating accordingly, which is still a rate-based control and different from ours in mapping the rotation motion.

HR(Head Rotation) maps the rotation of the head to the horizontal velocity of the UAV. If the user bows down his/her head, the UAV will move forward. If the user tilts his/her head backward and looks up, the UAV will move backward. The user can also roll his/her head to realize left and right translation. The UAV will keep turning if the user turns his/her head, which is the same as HT. The motion of the UAV on the vertical direction is also determined by whether the user is standing on his/her toes or squatting down.

FH(Flying Head) is a method that directly synchronizes the motion of the UAV to that of the user. A major flaw of it is that the flight range of the UAV is limited.

MFH(Modified Flying Head) again maps translation to speed as HT and our method. And its rotation control on the yaw direction is also a direct mapping mechanism as ours.

FP(Flying Plane) is inspired by the actions of the plane and the user can only move forward. When the user stands still, the UAV will move forward at a low speed. If the user moves backward, the UAV will lower down. If the user moves forward, the UAV will speed up. To turn the UAV, the user should tilt his/her head left or right. To move the UAV upward or downward, the user should tilt his/her head backward or forward.

4.2 Telexistence System

We build a simulation environment in Unity3D and set up a virtual 3D outdoor scene where the telexistence task is performed. Note that in this ideal environment, what the user controls is a virtual viewpoint rather than a virtual UAV, so there is no mapping error, delay and acceleration process. We conduct a user study in this simulation system to compare our method with the others.

4.2.1 Participants

Twenty-five participants are recruited to participant the user study and their average age is 24.24, ranging from 19 to 41. Twenty participants are male and five of them are female. Ten of them had

Table 1: Motion mapping mechanisms of different methods.

Mapping methods	Positioning mapping	Orientation mapping
Head Translation (HT)	HMD $(x,y,z) \rightarrow$ UAV $(\dot{x},\dot{y},\dot{z})$	HMD $yaw \rightarrow$ UAV $\dot{yaw}$
Head Rotation (HR)	HMD $(pitch,roll,z) \rightarrow$ UAV $(\dot{x},\dot{y},\dot{z})$	HMD $yaw \rightarrow$ UAV $\dot{yaw}$
Flying Head (FH)	HMD $(x,y,z) \rightarrow$ UAV (x,y,z)	HMD $yaw \rightarrow$ UAV yaw
Modified Flying Head (MFH)	HMD $(x,y,z) \rightarrow$ UAV $(\dot{x},\dot{y},\dot{z})$	HMD $yaw \rightarrow$ UAV yaw
2*Flying Plane (FP)	HMD $(x+/x-) \rightarrow$ UAV $(\dot{x}+/\dot{x}-)$ HMD $pitch \rightarrow$ UAV $\dot{z}$	2*HMD $roll \rightarrow$ UAV $\dot{yaw}$
2*Ours	2*HMD $(x,y,z) \rightarrow$ UAV $(\dot{x},\dot{y},\dot{z})$	HMD $yaw \rightarrow$ UAV yaw HMD $(pitch,roll) \rightarrow$ Camera $(pitch,roll)$

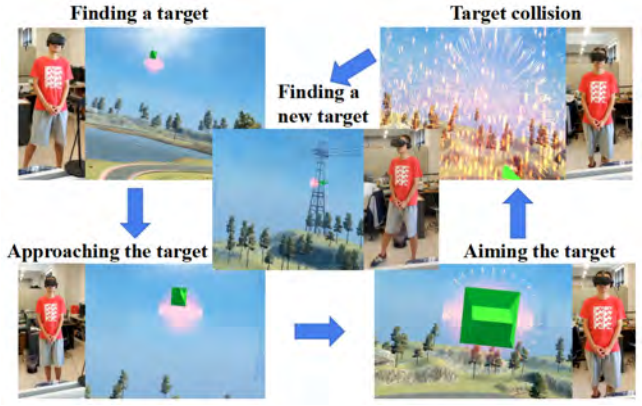

Figure 6: The task of our experiment in our simulation system. Step 1: identifying the target point in the scene; Step 2: approaching the point and identifying the required direction for passing through it; Step 3: adjusting the position and direction; Step 4: reaching the point; Step 5: identifying the next target point.

VR experience, and three of them had both VR experience and UAV operation experience.

4.2.2 Experiment Setup

We devise a task (Fig. 6) to evaluate different solutions. In the task, the user needs to control the UAV to move to four target locations one by one. The UAV is initialized in the center of the scene, and the first target location is rendered as a shining ball. The user should first saccade the surroundings to find the ball and then move towards it. A green arrow is pointing to the ball in a certain direction, which indicates that the user should move to the ball along the direction of the arrow. Therefore it is necessary for the user to tune the position and direction of the UAV carefully. If the user passes through the ball successfully, the next ball (indicating the next target location) will be generated in the scene and the user should follow the same rules to accomplish the task.

To ensure that the latter tested methods do not benefit from the experience of the user, the target locations are generated in a random manner. Note that the minimum flight distance to finish the task is fixed by properly setting the target locations. Moreover, we confirm that our task covers all the 6-DOF motions. Besides, we set two restrictions on the judgment of whether the user successfully passes through a target location. One requirement is that the distance between the user and the target point should be less than ρ (ρ= 0.5m, and the collision point is defined at the first the location that the user is less than ρ to the target.). The other is that the angle between the collision velocity and the direction of the arrow should be less than 5°. Notice that the two parameters are set empirically to make the task have a reasonable difficulty. We conduct our experiments, of course, in the form of blind test, which means the motion control mechanisms are tested in a random order.

4.2.3 Procedure and Task

Among the five methods, the FH method, which is the original version of position-based control methods, is not a viable solution to this task due to the limited motion space. As a consequence, we only show the user study results of the other four methods including MFH, HR, HT and FP as well as our method.

The performance of each method is evaluated in the aspects of **Time**, **Distance**, **Score** and **Workload**. **Time** indicates how much time the user spends on the task in seconds. **Distance** indicates the total length of the actual trajectory of UAV in the virtual space in meters. **Score** represents the score achieved in the task (rescaled to [0, 80]), which is calculated by:

$$Score = \sum_{i=1}^{4}[10*(1-\frac{\alpha_i}{180})+10*(1-\frac{l_i}{\rho})]. \quad (2)$$

Where i is the index of a target location, α_i is the angle between the collision velocity and the direction of the arrow, and l_i is the distance between the collision point and the target location. The score is also empirically designed to keep the angle and distance almost contribute equally to the final score. **Workload** indicates the overall rate of four subjective questions on the five control methods , which is calculated based on NASA-TLX [13,14] by the rating of the four subjective assessments (*mental demand*, *physical demand*, *effort* and *frustration*) of the users. Each one of them is divided into seven levels from positive evaluation to negative evaluation with identical intervals. *Mental demand* and *physical demand* are related to mental and physical consumption respectively. *Effort* measures how much effort users make to accomplish the task, whereas *frustration* reflects the extent of discouragement and anger expressed by the users.

4.2.4 Result Of Telexistence Experiment

We compare the performance of different methods and evaluate the statistical significance in Fig. 7 and Table 2, respectively. In Fig. 7, we calculate the population means using 95% confidence intervals (*CI*) for all the methods. In Table 2, we list the pairwise comparisons between our method and each of the compared methods obtained by one way ANOVA. So for each value of a particular method in Table 2, it stands for the comparison between this particular method and our method. To be more specific, we calculate F value, Partial η_p^2 , and Tukey adjusted p-values (referring to [21], [19], and [41] for details to calculate them respectively). As shown in Fig. 7, our method achieves better mean values on all the terms comparing with all the compared solutions. The values in Table 2 further indicate that the differences on **Time**, **Distance**, and **Workload** are statistically significant.

This means that compared with other HMD based control methods, our method not only provide a nature way to perceive the origin in rate-based control but also employ a full viewpoint control mechanism, which allows users to spend less time and distance to complete a task. As a consequence, we achieve the best mean values. The performance of FP is far behind our method on this task because FP only allows controlling one motion direction. The performance

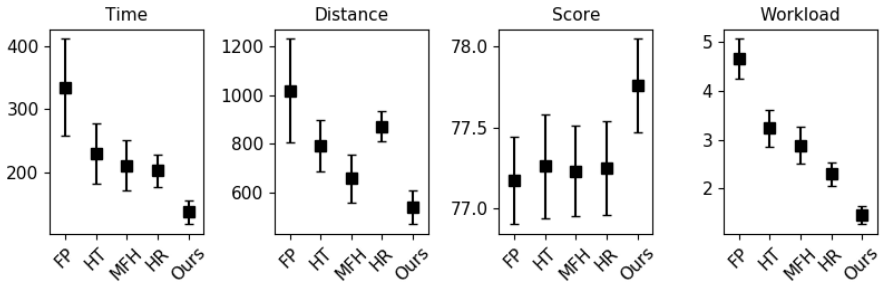

Figure 7: Statistics of different methods on Time, Distance, Score and Workload, which shows 95% CIs of the results of the 5 compared mapping mechanisms.

Table 2: Statistical analysis of Telexistence experiment with F-value, Partial η^2, and Tukey adjusted p-values.

Item	Time			Distance			Workload		
	F(1, 48)	η_p^2	p(F<f)	F(1, 48)	η_p^2	p(F<f)	F(1, 48)	η_p^2	p(F<f)
FP	70.598	0.595	0.000	77.769	0.618	0.000	247.496	0.837	0.000
HT	47.332	0.496	0.018	83.061	0.633	0.002	139.189	0.743	0.000
MFH	21.459	0.308	0.000	32.715	0.405	0.000	74.253	0.607	0.000
HR	13.076	0.214	0.020	26.001	0.351	0.003	30.248	0.386	0.001

of HR, HT and MFH is relatively better than FP. HR maps the rotation of the head to the horizontal velocity of the UAV, which is not natural enough and results in unwanted motions leading to bigger **Distance** values. FP, HR and HT employ rate-control on yaw angle and require users to remember their initial orientations, which is not an intuitive interactive way. Although this drawback is overcome by MFH, it does not achieve 6-DOF full viewpoint control with the absence of two DOFs, increasing the **Time** values.

4.2.5 Result of AOU Experiment

As indicated by some previous works in 3D interaction [3, 44], traditional rate-based control methods are not appropriate for devices without a self-centering mechanism. To achieve self-centering, we propose the AOU scheme to reconfigure the origin. In this section, we evaluate this contribution by a user study.

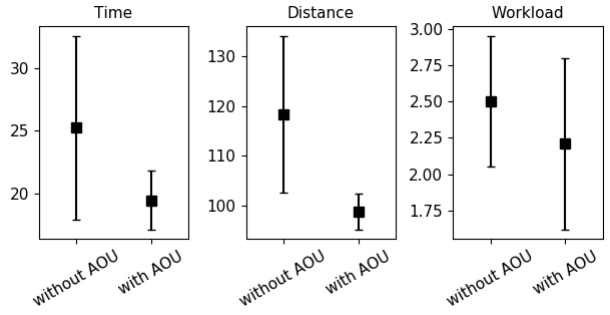

Figure 8: Evaluation of our AOU method with 95% CIs of the results of our scheme with and without AOU.

The task is the same to the former one, the same participants will do the experiment immediately after the previous experiment. But the user has to pass through only one target and the judgment of whether the user successfully passes through the target is much stricter. This is to verify that our AOU scheme gets rid of the mapping error and achieves a precise direction control. We compare our solution with and without AOU on **Time**, **Distance** and **Workload**. **Score** is no longer evaluated due to the strict judgment. The results are shown in Fig. 8. We can see that the performance of our method with AOU is generally better than without AOU on **Time** and **Distance** as well as **Workload**. The reason is that without AOU, sometimes the movement of users is not correctly mapped to the desired motion of the UAV's flight. Then more time and distance are required to complete the task, also leading more workload for users. For **Time** (F(1, 48) = 12.077, $\eta_p^2 = 0.201$, $p < 0.01$) and **Distance** (F(1, 48) = 18.230, $\eta_p^2 = 0.275$, $p < 0.01$), our improvement is statistically significant as AOU allows moving less distance to the origin. However, for **Workload** (F(1,48) = 2.899, $\eta_p^2 = 0.057$, $p = 0.095$), the difference is not statistically significant. This is because that the participants of this experiment have accomplished the previous experiment which give them more experience to master the rate-based control mechanism. Also, comparing with the previous four targets task, this one target task is relatively easy and the origin could be remembered with less errors.

4.3 Application in 3D Reconstruction

Another simulation system is developed to evaluate the performance of different techniques in the UAV-based 3D reconstruction. To validate the effectiveness of our 6-DOF mechanism, we further compare our original method with and without the gimbal. The traditional joystick-based control is also compared here, and a knob on the controller is employed to control the pitch angle.

4.3.1 Participants

For this study, fifteen participants are involved in the user study and their average age is 23.98, ranging from 21 to 42. Ten participants are male and five of them are female. Eight participants had 3D game operating experience, and three were familiar with 3D game and operating UAV by joystick.

4.3.2 Experiment Setup

This simulator is built on Gazebo [2], a robot simulation platform which is integrated with Robot Operating System (ROS) for easy robot communication. In this experiment, users are asked to reconstruct the model as complete as possible. Then we measure the flight distance, time and workload to compare different solutions. In this system, we simulate wind and gravity to make a relatively real environment. And a simulated UAV with noisy GPS signals is used to perform the experiment. With a virtual Kinect sensor set onboard, we record depth and RGB images of the virtual world. A virtual

Table 3: Quantitative reconstruction results of all compared methods with 10mm, 20mm and 30mm thresholds respectively.

Methods	Recall[%] (<10mm)	Recall[%] (<20mm)	Recall[%] (<30mm)
Ours	**60.88**	**69.42**	**73.62**
No gimbal	56.86	66.18	71.31
Joystick	57.02	66.69	71.52
MFH	55.26	69.10	72.86
HR	60.18	65.45	71.39
HT	53.79	65.45	71.39

gimbal is also implemented to achieve a full version of our proposed method. In this task, users are asked to control the UAV and move around a virtual car using different techniques. In all these methods, the recorded depth will be fused together to build 3D models. Fig. 9 is an illustration of the procedure and task.

4.3.3 Procedure and Task

Besides the assessments **Time**, **Distance** and **Workload** mentioned in Sect. 4.2. We use some new metrics to evaluate the reconstruction results. *Recall*, namely completeness, mentioned in [20], is defined to quantify to what extent the ground-truth points are completely reconstructed. The **No gimbal** method represents our scheme without gimbal. According to users' feedback, the **FP** method is not applicable for the task because the UAV can only move forward while 3D scanning often requires omnidirectional translations.

4.3.4 Result of Application in 3D Reconstruction

Figure 9: Our reconstruction task. The top row shows a user is controlling a simulated UAV using our proposed technique, and the bottom row is the illustration of joystick interface. For a comprehensive comparison, the right knob on the remote controller can be used to adjust pitch angle.

Table 3 shows the completeness of the reconstructed 3D models of different methods. Fig. 10 further compares the performance of different methods in the 3D reconstruction task based on **Time**, **Distance**, and **Workload**. In Table 4, we list two pairwise comparisons (our method vs. no gimbal, and our method vs. joystick) obtained by one-way ANOVA. To be specific, we again calculate F values, η_p^2, and Tukey adjusted p-values. It can be seen in Table 3 that our method helps the user to achieve more complete reconstruction than all the compared methods. From the results shown in Fig. 10, the joystick-based method has worse performance than our method on **Time** and **Distance**. And the differences are significant as indicated by the numbers in Table 4. This mainly because that users should coordinate the camera's angle of view, UAV's orientation, and the gimbal to focus on the target during the scanning. The coordinating operation may be difficult especially for less-skilled operators. For example, to control the pitch angle, the user has to spare one hand and turn the knob on the remote controller because the two joysticks can only provide 4-DOFs, which is quite troublesome compared with our HMD-based control. Fig. 4 illustrates this shortcoming. Note that it is not easy to control the camera always face towards the target during flying, especially when turning around a corner. The two hands need to cooperate very well to control the translation of the UAV and the orientation of the camera. This is also demonstrated

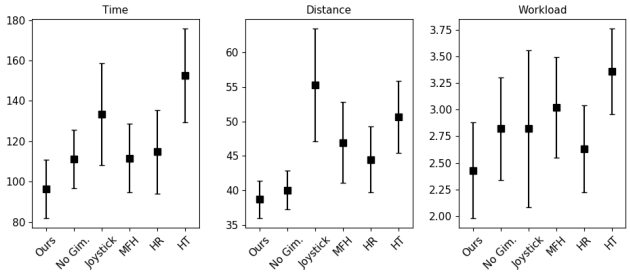

Figure 10: The results (95% CIs) of all control methods on Time, Distance and Workload in our simulated 3D reconstruction user study.

in our video. On the other hand, the joystick-based method uses fingers to control, which is less labor intensive than our method which uses body gestures. So our improvement on **Workload** is not statistically significant as shown in Table 4. The differences between our method and the no gimbal solution is not statistically significant on **Time**, **Distance** and **Workload** as shown in Table 4. This is because the no gimbal solution already has our AOU strategy and the user could finish the task by just moving around the target trunk from a horizontal plane without using the additional 2-DOF. Even though the task is completed, the roof region of the trunk is difficult to be fully reconstructed without the additional 2-DOF (helping to move vertically). As a consequence, the completeness of the reconstructed model of the no gimbal solution is not as good as ours as shown in Table 3. The comparisons between our method and MFH, HR, HT and FP on **Time**, **Distance** and **Workload** have already been performed in the telexistence experiment as shown in Fig. 7 and Table 2, so we have not compared them again in this section.

4.4 Real prototype

Besides the simulation systems, we also build a real prototype, which is extremely challenging to deploy such a complex system with a real drone. We demonstrate our prototype in the supplementary video.

4.4.1 Hardware Setup

We adopt an Oculus Rift with a resolution of $1280*720$ as the HMD in our system. An IR sensor is used to track the 6-DOF motions of the HMD in real time, which is placed about one-and-a-half meters in front of the user with a height of about one meter. The ground station of our system is a PC running with Win10, equipped with a 16GB RAM, an AMD Ryzen 5 2600X Six-Core processor and one NVIDIA GeForce GTX Titan XP graphics card. We do not use an off-the-shelf UAV as it cannot fully satisfy our requirements. Instead, we DIY a UAV with a Pixhawk flight control unit, a GPS and a pan-tilt-zoom. The stereo camera onboard is MYNTAI D1000-50/Color, which is equipped with an IMU and outputs depth map ranging from 0.49m-10m at the rate of 30 fps, as well as stereo color streams. An NVIDIA Jetson TX2 on the UAV is utilized for image capture and data transmission. In our prototype, we transfer video streams from the UAV to the ground station via a WiFi connection. A 5.8GHz digital downlink is adopted to convey control signals, which are packed by MAVLink protocol to the UAV.

Fig. 11 shows the real equipment of our prototype. We can see the two transmission modules on the UAV and the HMD with its IR sensor as well. Please refer to our video to see how a user controls a UAV to navigate in an outdoor scene.

4.4.2 Real Results

Here we just perform the reconstruction task using our real prototype. Fig. 12 gives an example of it, and the obtained 3D models are shown

Table 4: Statistical analysis of reconstruction experiment with F-value, Partial η^2, Tukey adjusted p-values.

Item	Time			Distance			Workload		
	F(1, 28)	η_p^2	p(F<f)	F(1, 28)	η_p^2	p(F<f)	F(1, 28)	η_p^2	p(F<f)
No Gimbal	2.264	0.074	0.144	0.595	0.018	0.474	1.509	0.051	0.230
Joystick	6.991	0.199	0.013	15.911	0.362	0.000	0.818	0.028	0.374

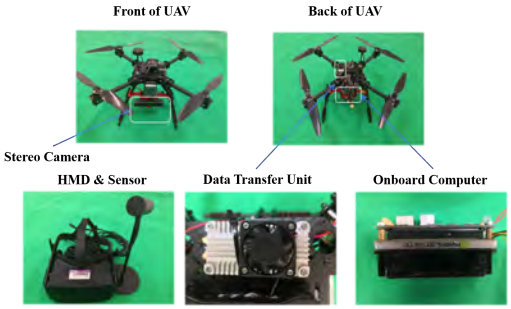

Figure 11: The hardware of our real prototype. Top: the frontal and backside of the UAV; Bottom: the HMD, the data transmission module and the NVIDIA Jetson TX2 for image capture and data transmission.

Figure 12: The real prototype with our proposed interaction interface. The top left part of each figure is the real-time fused model, the bottom left is the picture rendered in the HMD, the middle shows the flight of the UAV, and the right shows the motion of the user.

in Fig. 13. The reconstruction quality here is not so high as that in the simulation system due to the limitations of the depth image estimation and the low quality raw depth obtained by our physical stereo camera. The two issues should be conquered by the progress of the 3D algorithms and equipment. But we can still see that with our technique, a user could control a UAV to record a complete 3D model of an object instead of holding a sensor and moving around the target with a delicately designed path to cover a complete surface.

4.5 Discussions and Limitations

In general, our AOU strategy improves the rate-based control for the HMD-base telexistence. In the simulated navigation task, a comprehensive user study indicates that in average our method performs better than the alternative solutions on the four evaluation items of **Time**, **Distance**, **Score** and **Workload**. The AOU is further evaluate by another simple task. Its improvement on **Time** and **Distance** is statistically significant while the improvement on **Workload** is not significant due to the experience of the participants and the simplicity of the task. The evaluation on the application of 3D reconstruction also indicates the improvement of our method on **Time**, **Distance** and **Workload** over alternative solutions including the joystick-based solution. For the additional 2 DOFs provided by the gimbal, they do not make significant improvement on controlling the UAV, but they help in the reconstruction completeness, which is also important in this application.

Figure 13: Reconstruction results using our telexistence prototype. The left two columns are the RGB images for reference and the right two columns are the corresponding reconstructed 3D models.

Our prototype still suffers from the latency caused by all the related components of our system. It takes about 200ms to transmit images back to the ground station and 50ms to transmit the controlling signals to the UAV. Another 250ms is taken for the UAV to respond. As a consequence, the total latency is about 500ms. Owing to the latency, the user should move slowly to use the system and thus the telexistence experience is limited. Reducing this latency is the key issue to make this kind of technique be practical in our daily lives.

Our system employs some predefined parameters, which may influence users' experience and different users may have different experiences for the same set of parameters. In all our experiments, we first tune the parameters by 3-5 users and then fix them. But from the feedback of the users, the parameters are not perfect for all of them. For example, some users think the system is too sensitive. When they just want to saccade in the scene, the UAV moves with the head motions, indicating that we should set a larger motion sphere. On the other hand, some users think the sphere is too large that they feel difficult to move the UAV. To this end, we think it is better to recognize the semantic meanings of the user's actions, rather than a threshold on the user's position. In the future, we would like to label some data and train a machine learning model to distinguish the meanings of users' motions. Better performance is expected to be achieved.

5 Conclusions

In this paper, we have developed a 6-DOF telexistence drone to present natural viewpoint control and 3D immersive viewing experience to a user. We also demonstrate its power in 3D scene reconstruction. There are three key contributions in our technique. Firstly, we propose an AOU scheme that handles the lack of self-centering in the body-involved rate-based control mechanism. It doe not require any additional hardware comparing with the previous solutions. Secondly, we integrate a gimbal into our telexistence drone, which provides the full 6-DOF to users to control the drone, which helps in the 3D reconstruction application on the reconstruction completeness. Thirdly, we perform three user studies in two simulation systems and a real prototype to evaluate our technique by comparing alternative control method in telexistence as well as the joystick-based control for 3D reconstruction.

Acknowledgments

The authors acknowledge support by the NSFC (No.61822111, 61931012, 61727808) and Beijing Natural Science Foundation (JQ19015)

REFERENCES

[1] H. Abe, M. Miura, T. Abe, and T. Suganuma. A telexistence system using remote control robot and omnidirectional camera with qoe control. In *2018 IEEE 7th Global Conference on Consumer Electronics (GCCE)*, pp. 1–3, Oct 2018. doi: 10.1109/GCCE.2018.8574612

[2] C. E. Agüero, N. Koenig, I. Chen, H. Boyer, S. Peters, J. Hsu, B. Gerkey, S. Paepcke, J. L. Rivero, J. Manzo, E. Krotkov, and G. Pratt. Inside the virtual robotics challenge: Simulating real-time robotic disaster response. *IEEE Transactions on Automation Science and Engineering*, 12(2):494–506, April 2015. doi: 10.1109/TASE.2014.2368997

[3] L. Besançon, M. Ammi, and T. Isenberg. Pressure-based gain factor control for mobile 3d interaction using locally-coupled devices. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pp. 1831–1842. ACM, New York, NY, USA, 2017. doi: 10.1145/3025453.3025890

[4] E. Z. Borba, A. Montes, R. de Deus Lopes, M. K. Zuffo, and R. Kopper. Itapeva 3d: Being indiana jones in virtual reality. In *2017 IEEE Virtual Reality (VR)*, vol. 00, pp. 361–362, 2017. doi: 10.1109/VR.2017.7892326

[5] D. A. Bowman and L. F. Hodges. An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments. In *Proceedings of the 1997 Symposium on Interactive 3D Graphics*, I3D '97, pp. 35–ff. ACM, New York, NY, USA, 1997. doi: 10.1145/253284.253301

[6] D. A. Bowman, E. Kruijff, J. LaViola, and I. Poupyrev. *3D User Interfaces: Theory and Practice*. Addison-Wesley Professional, Boston, 2005.

[7] J. Chen, D. Bautembach, and S. Izadi. Scalable real-time volumetric surface reconstruction. *ACM Trans. Graph.*, 32(4):113:1–113:16, July 2013. doi: 10.1145/2461912.2461940

[8] Y. Chen and G. Medioni. Object modelling by registration of multiple range images. *Image and Vision Computing*, 10(3):145 – 155, 1992. Range Image Understanding. doi: 10.1016/0262-8856(92)90066-C

[9] A. Cherpillod, S. Mintchev, and D. Floreano. Embodied flight with a drone. *CoRR*, abs/1707.01788, 2017.

[10] R. A. S. Fernández, J. L. Sanchez-Lopez, C. Sampedro, H. Bavle, M. Molina, and P. Campoy. Natural user interfaces for human-drone multi-modal interaction. In *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 1013–1022, June 2016. doi: 10.1109/ICUAS.2016.7502665

[11] A. Geiger, J. Ziegler, and C. Stiller. Stereoscan: Dense 3d reconstruction in real-time. In *2011 IEEE Intelligent Vehicles Symposium (IV)*, pp. 963–968, June 2011. doi: 10.1109/IVS.2011.5940405

[12] C. Guo and E. Sharlin. Exploring the use of tangible user interfaces for human-robot interaction: A comparative study. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, p. 121–130. Association for Computing Machinery, New York, NY, USA, 2008. doi: 10.1145/1357054.1357076

[13] S. G. Hart. Nasa-task load index (nasa-tlx); 20 years later. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 50(9):904–908, 2006. doi: 10.1177/154193120605000909

[14] S. G. Hart and L. E. Staveland. Development of nasa-tlx (task load index): Results of empirical and theoretical research. In P. A. Hancock and N. Meshkati, eds., *Human Mental Workload*, vol. 52 of *Advances in Psychology*, pp. 139 – 183. North-Holland, 1988. doi: 10.1016/S0166-4115(08)62386-9

[15] H. Hayakawa, C. L. Fernando, M. Y. Saraiji, K. Minamizawa, and S. Tachi. Telexistence drone: Design of a flight telexistence system for immersive aerial sports experience. In *Proceedings of the 6th Augmented Human International Conference*, AH '15, pp. 171–172. ACM, New York, NY, USA, 2015. doi: 10.1145/2735711.2735816

[16] R. Herrmann and L. Schmidt. Design and evaluation of a natural user interface for piloting an unmanned aerial vehicle. *i-com*, 17(1):15–24, 2018.

[17] K. Higuchi and J. Rekimoto. Flying head: A head motion synchronization mechanism for unmanned aerial vehicle control. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '13, pp. 2029–2038. ACM, New York, NY, USA, 2013. doi: 10.1145/2468356.2468721

[18] S. Kasahara and J. Rekimoto. Jackin head: Immersive visual telepresence system with omnidirectional wearable camera for remote collaboration. In *Proceedings of the 21st ACM Symposium on Virtual Reality Software and Technology*, VRST '15, pp. 217–225. ACM, New York, NY, USA, 2015. doi: 10.1145/2821592.2821608

[19] F. N. Kerlinger. Foundations of behavioral research: educational and psicological inquiry. Technical report, 1964.

[20] A. Knapitsch, J. Park, Q.-Y. Zhou, and V. Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics*, 36(4), 2017.

[21] R. G. Lomax. *Statistical concepts: A second course*. Lawrence Erlbaum Associates Publishers, 2007.

[22] J. Miehlbradt, A. Cherpillod, S. Mintchev, M. Coscia, F. Artoni, D. Floreano, and S. Micera. Data-driven body–machine interface for the accurate control of drones. *Proceedings of the National Academy of Sciences*, 115(31):7913–7918, 2018. doi: 10.1073/pnas.1718648115

[23] B. Myers, S. E. Hudson, and R. Pausch. Past, present, and future of user interface software tools. *ACM Trans. Comput.-Hum. Interact.*, 7(1):3–28, Mar. 2000. doi: 10.1145/344949.344959

[24] M. Nießner, M. Zollhöfer, S. Izadi, and M. Stamminger. Real-time 3d reconstruction at scale using voxel hashing. *ACM Transactions on Graphics (TOG)*, January 2013.

[25] S. Orts-Escolano, C. Rhemann, S. Fanello, W. Chang, A. Kowdle, Y. Degtyarev, D. Kim, P. L. Davidson, S. Khamis, M. Dou, et al. Holoportation: Virtual 3d teleportation in real-time. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, pp. 741–754, 2016.

[26] S. Park, Y. Jung, and J. Bae. A tele-operation interface with a motion capture system and a haptic glove. In *2016 13th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, pp. 544–549, Aug 2016. doi: 10.1109/URAI.2016.7625774

[27] T. Pejsa, J. Kantor, H. Benko, E. Ofek, and A. Wilson. Room2room: Enabling life-size telepresence in a projected augmented reality environment. In *Proceedings of the 19th ACM conference on computer-supported cooperative work & social computing*, pp. 1716–1725, 2016.

[28] K. Pfeil, S. L. Koh, and J. LaViola. Exploring 3d gesture metaphors for interaction with unmanned aerial vehicles. In *Proceedings of the 2013 International Conference on Intelligent User Interfaces*, IUI '13, pp. 257–266. ACM, New York, NY, USA, 2013. doi: 10.1145/2449396.2449429

[29] C. Pittman and J. J. LaViola, Jr. Exploring head tracked head mounted displays for first person robot teleoperation. In *Proceedings of the 19th International Conference on Intelligent User Interfaces*, IUI '14, pp. 323–328. ACM, New York, NY, USA, 2014. doi: 10.1145/2557500.2557527

[30] I. Poupyrev, M. Billinghurst, S. Weghorst, and T. Ichikawa. The go-go interaction technique: Non-linear mapping for direct manipulation in vr. In *Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology*, UIST '96, pp. 79–80. ACM, New York, NY, USA, 1996. doi: 10.1145/237091.237102

[31] T. Qin, P. Li, and S. Shen. Vins-mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Transactions on Robotics*, 34(4):1004–1020, 2018.

[32] B. Reitinger, C. Zach, and D. Schmalstieg. Augmented reality scouting for interactive 3d reconstruction. In *2007 IEEE Virtual Reality Conference*, pp. 219–222, March 2007. doi: 10.1109/VR.2007.352485

[33] C. Rognon, S. Mintchev, F. Dell'Agnola, A. Cherpillod, D. Atienza, and D. Floreano. Flyjacket: An upper body soft exoskeleton for immersive drone control. *IEEE Robotics and Automation Letters*, 3(3):2362–2369, July 2018. doi: 10.1109/LRA.2018.2810955

[34] A. Sanna, F. Lamberti, G. Paravati, and F. Manuri. A kinect-based natural interface for quadrotor control. *Entertainment Computing*, 4(3):179 – 186, 2013. doi: 10.1016/j.entcom.2013.01.001

[35] T. Schöps, T. Sattler, C. Häne, and M. Pollefeys. Large-scale outdoor 3d reconstruction on a mobile device. *Computer Vision and Image Understanding*, 157:151–166, 2017.

[36] Z. Shang and Z. Shen. Real-time 3d reconstruction on construction site using visual slam and uav. *arXiv preprint arXiv:1712.07122*, 2017.

[37] T. Shibata. Head mounted display. *Displays*, 23(1):57 – 64, 2002. doi:

10.1016/S0141-9382(02)00010-0

[38] A. R. Smith. Color gamut transform pairs. *SIGGRAPH Comput. Graph.*, 12(3):12–19, Aug. 1978. doi: 10.1145/965139.807361

[39] S. Tachi. Real-time remote robotics-toward networked telexistence. *IEEE Computer Graphics and Applications*, 18(6):6–9, Nov 1998. doi: 10.1109/38.734972

[40] S. Tachi. Telexistence: Enabling humans to be virtually ubiquitous. *IEEE Computer Graphics and Applications*, 36(1):8–14, Jan.-Feb. 2016. doi: 10.1109/MCG.2016.6

[41] J. Tukey. Multiple comparisons. *Journal of the American Statistical Association*, 48(263):624–625, 1953.

[42] S. Wang, X. Xiong, Y. Xu, C. Wang, W. Zhang, X. Dai, and D. Zhang. Face-tracking as an augmented input in video games: Enhancing presence, role-playing and control. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '06, p. 1097–1106. Association for Computing Machinery, New York, NY, USA, 2006. doi: 10.1145/1124772.1124936

[43] M. J. Wells, B. N. Peterson, and J. Aten. The virtual motion controller: a sufficient-motion walking simulator. In *In Proceedings of VRAIS '97*, pp. 1–8, 1996.

[44] S. Zhai. User performance in relation to 3d input device design. *SIGGRAPH Comput. Graph.*, 32(4):50–54, Nov. 1998. doi: 10.1145/307710.307728