applied sciences

IMPACT FACTOR 2.7

CITESCORE 4.5

Article

GazeGuide: An Eye-Gaze-Guided Active Immersive UAV Camera

Pavan Kumar B. N., Adithya Balasubramanyam, Ashok Kumar Patil, Chethana B. and Young Ho Chai

Special Issue

Human-Robot Systems: Modeling, Control and Prediction

Edited by
Prof. Dr. Rainer Palm

MDPI

https://doi.org/10.3390/app10051668

applied sciences

MDPI

Article

GazeGuide: An Eye-Gaze-Guided Active Immersive UAV Camera

Pavan Kumar B. N., Adithya Balasubramanyam, Ashok Kumar Patil, Chethana B. and Young Ho Chai *

Graduate School of Advanced Imaging Science, Multimedia and Film, Chung-Ang University, Seoul 06974, Korea; pavanbn8@gmail.com (P.K.B.N.); adithyakoundinya@gmail.com (A.B.); ashokpatil03@hotmail.com (A.K.P.); chethanab26@gmail.com (C.B.)

* Correspondence: yhchai@cau.ac.kr; Tel.: +82-2-820-5335

Received: 31 January 2020; Accepted: 26 February 2020; Published: 1 March 2020

Abstract: Over the years, gaze input modality has been an easy and demanding human–computer interaction (HCI) method for various applications. The research of gaze-based interactive applications has advanced considerably, as HCIs are no longer constrained to traditional input devices. In this paper, we propose a novel immersive eye-gaze-guided camera (called GazeGuide) that can seamlessly control the movements of a camera mounted on an unmanned aerial vehicle (UAV) from the eye-gaze of a remote user. The video stream captured by the camera is fed into a head-mounted display (HMD) with a binocular eye tracker. The user's eye-gaze is the sole input modality to maneuver the camera. A user study was conducted considering the static and moving targets of interest in a three-dimensional (3D) space to evaluate the proposed framework. GazeGuide was compared with a state-of-the-art input modality remote controller. The qualitative and quantitative results showed that the proposed GazeGuide performed significantly better than the remote controller.

Keywords: eye tracking; HRI; eye-gaze; gaze-based interaction; HMD; robotics; gaze input; virtual reality; surveillance and monitoring

1. Introduction

The eyes are a rich source of information for grouping contexts in our everyday lives. Gaze has served throughout history as a mode of interaction and communication among people [1]. It has been postulated as the best proxy for attention or intention [2]. Nowadays, eye tracking has matured and become an important research topic in computer vision and pattern recognition, because human gaze positions and movements are essential information for many applications ranging from diagnostic to interactive ones [3–6]. Eye tracking equipment, either worn on the body (head-mounted) [7] or strategically located in the environment [8], is a key requirement of gaze-based applications. In contrast, recent advances in the applications of head-mounted displays (HMDs) for virtual reality have driven the development of HMD-integrated eye-tracking technology. These displays are becoming widely accessible and affordable. The increased accuracy of eye-tracking equipment has made the use of this technology for explicit control tasks with robots feasible [9].

The research on human–robot interaction (HRI) strives to ensure easy and intuitive interactions between people and robots. Such interactions need natural communication [10]. For human–human interactions, verbal communication is usually the primary factor; at the same time, non-verbal behaviors such as eye-gaze and gestures can convey the person's mental state, augment verbal communication, and portray what is being expressed [11,12] by humans for interaction with robots. In particular, unmanned aerial vehicles (UAVs) a flying robot that can move freely through the air have a long history in military applications. They are capable of carrying payloads over long distances while being teleoperated remotely. Teleoperation remains a hot topic in HRI [13], enabling an operator to control a robot remotely in a hazardous inaccessible place. It has been widely used in a variety of applications ranging from space exploration, inspection, robotic navigation, and underwater operations. Many researchers in the field of robotics are interested in remotely controllable agents rather than fully autonomous agents [14].

Typically, in surveillance and monitoring applications, UAVs are equipped with some specific sensors such as light detection and ranging (LiDAR) [15–17], and video cameras [18]. They keep monitoring the ground targets of interest (ToIs), such as humans, buildings, pipelines, roads, vehicles, and animals, for the purposes of surveillance, monitoring, security, etc. [18]. In such and many other remotely controlled applications, the operator's basic perceptual link is to continuously monitor the status of the robot [19] visually. As the eyes of the operator are engaged in the monitoring task, the main motivation for our research was to explore the use of gaze to control a UAV-mounted camera effortlessly. An effortless control of a UAV's camera is a novel research topic on fusing human input and output modalities.

This paper presents a novel framework to control a UAV camera maneuvering through eye-gaze as an alternative and sole input modality. Thus, spatial awareness is directly fed without being mediated through remote control. The rest of this paper is organized as follows: In Section 2, related works are discussed. Section 3 discusses the proposed immersive eye-gaze-guided camera (GazeGuide) and its implementation. Section 4 presents a user study followed by the results, conclusion, and future work directions.

2. Related Works

Our work builds on two main relevant strands of previous work: (1) eye-gaze tracking and (2) gaze-based interactions.

2.1. Eye-Gaze Tracking

The eye-gaze tracking technology has matured and become inexpensive. This technology can be built into computers, HMDs, and mobile devices [8]. The direction of the eye-gaze is a crucial and augmented input medium/control modality through which humans express socially relevant information, particularly the individual's cognitive state [7,20,21]. Hence, various studies on gaze tracking have been conducted beginning with a video-based eye tracking study on a pilot-operated airplane [22]. The state-of-the-art eye-tracking devices can be divided into two main categories: table-mounted and mobile (head-mounted) eye-tracking systems, while two-directional (2D) appearance-based and 3D model-based techniques are the two main methods for gaze estimation [23–25].

Gaze tracking has been advanced with the goal of improving accuracy and reducing the constraints on the users [26]. Active advancements in the speed of computation, digital video processing, and low-cost hardware have relatively increased the accessibility of gaze-tracking devices to both expert and novice users. Therefore, gaze tracking has found applications in gaming, interactive applications, diagnostics, rehabilitation, desktop computers, automotive setups, TV panels, and virtual and augmented reality applications [27–32]. These various applications and eye-gaze research include studies on several types of eye movements, as given in Table 1, to collect information about a user's intent, behavior, cognitive state, and attention [33].

While all of these applications use one or another eye-tracking devices, the physiology of the human eye is not uniform throughout the population. The eye tracker has a job at hand to calibrate itself or be calibrated manually before use according to the user's physiology in order to achieve the desired results. It goes without saying that calibration is an intrinsic phenomenon of any eye-tracking system [34,35]. Most eye-movement researchers use algorithms to parse raw data and differentiate among the various types of eye movements (Table 1); however, the process of identifying and separating fixations and saccades in eye tracking remains an essential part of an eye-movement data analysis. Salvucci et al. [36] discussed and evaluated various algorithms such as velocity-threshold identification (I-VT), hidden Markov model fixation identification (I-HMM), dispersion-threshold identification (I-DT), minimum spanning tree identification (I-MST), and area-of-interest fixation identification (I-AOI) for automatic fixation and saccade identification. A qualitative analysis of these algorithms summarized that I-DT provides accurate and robust fixation identification. These automatic identification algorithms enable eye trackers for real-time adoptions [37]. In the proposed GazeGuide, we used I-DT for filtering the raw gaze data from the eye tracker into fixations in real time.

Table 1. Eye-movement classification.

Eye-Movement Type	Definition	Occurrence Duration	Significance
Fixation	Maintaining visual gaze on a single location	200–400 ms	Acquiring information, Cognitive processing
Saccade	Simultaneous eye movements between fixations	20–200 ms	Moving between targets
Smooth Pursuit	Eyes closely following a moving object	100 ms	Following moving targets
Scan path	Series of fixations and saccades	-	Scanning, direct search
Gaze duration	Total fixation time on a target	-	Cognitive processing and conveying intent
Blink	Rapid eyelid closing	300 ms	Indicates behavioral states, stress
Pupil size change	Pupil constriction or dilation	140 ms	Cognitive effort, representing microemotions

During the past several decades, there has been a dramatic increase in the quantity and variety of studies using the eye-tracking technology. Figure 1 illustrates the number of peer-reviewed articles over the past 55 years (merged into successive five-year periods) containing the phrase "eye tracking" and/or "eye movements."

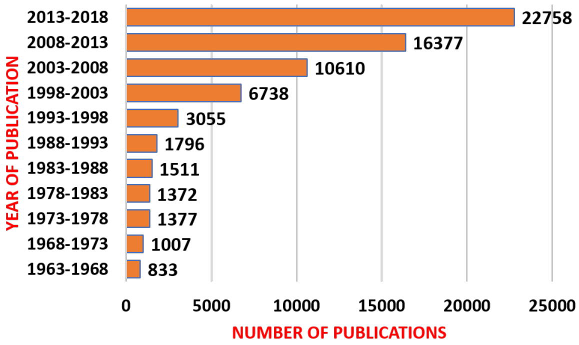

Figure 1. Number of peer-reviewed articles over the past 55 years with the phrase eye tracking and/or eye movements (ProQuest databases).

2.2. Gaze-Based Interaction and Control

As part of the development of interactive applications, many researchers have used the eye-gaze as an input to achieve user interface-based HCIs. The introduction of several fundamental gaze-based techniques by Jakob [20] showed the usage of eye movements as a fast input mode for interaction. In another study, a gaze-based interface was proposed by Latif et al. [38,39] to teleoperate a robot by aiming at minimizing the human body engagement of the operator in the controlling task.

A human eye-gaze-based interface was used to control a wheelchair in [40] and similarly, in [41], a recent study with a case study was reported. Deepak et al. [42] presented a novel gaze-based interface called GazeTorch, comparing between the use of gaze and that of mouse pointers in remote collaborative physical tasks [43]. A study [44] showed the benefits of gaze augmentation in egocentric videos in a driving task. Yoo et al. [45] proposed the teleoperation of a robotic arm from a remote location by using an experimental eye-tracking algorithm. Furthermore, Daniel et al. [46] presented a telerobotic platform using the eye as the input to control the navigation of a teleoperated mobile robot. The demo explored the use of gaze as an input for locomotion. In [47], the researchers presented an experimental investigation of gaze-based control modes for UAV, and in [48], the scholars investigated a gaze-controlled remote vehicle while driving it on a racing track with five different input devices.

On a similar line of thought, Khamis et al [49] presented GazeDrone, a novel gaze-based interaction in public pervasive settings, and Mingxin et al. [50] used gaze gestures as an object selection strategy into HRI to teleoperate the drone. Kwok et al. [51] explored eye tracking for surgical applications by using simultaneous displays of two users' gaze points to indicate their respective intents during robotic surgery tasks.

In a study [52], gaze information obtained by an eye tracker could be used to create a gaze-contingent control system to move the camera; with this, it became possible to move the camera by following the user's gaze position on the target anatomy. In a recent survey [53], the authors explored gaze-based interactions in the cockpit, which actively involved the pilots in the exploration. Furthermore, in the survey report [10], Ruhaland et al. [54] outlined earlier work that used the eye-gaze and explained the high-level outcomes achieved from an application-oriented perspective through the eye-gaze in HRI. Zhai et al. [19] summarized the compelling reasons, advantages, and motivations for gaze-based controls as follows:

- Gaze can be an effective solution for situations that prohibit the use of hands.
- Velocity is another important feature of gaze control, because an object can be looked at considerably faster than it can be reached by hand or any tool. Hence, the use of the eye-gaze increases the speed of the user input.
- The eyes require no training: it is natural for the users to look at the objects of interest.

In contrast, an immersive environment gives an operator the feeling of being present in the remote world and of embodiment [55]. Embodiment occurs when the remote operator experiences similar perception and sensations as the robot experiences. In order to provide an immersive experience, HMDs are commonly used to provide users with the stereoscopic first-person view of the remote scene. Hansen et al. [56] proposed gaze-controlled telepresence around a robot platform, with eye tracking in an HMD.

In summary, the aforementioned research and development motivated us to consider the gaze for UAV camera control; doing so would offer a direct and immediate mode of interaction. Therefore, the focus of this work was to use gaze as a sole control input intuitively to maneuver a UAV camera effortlessly with respect to the operator's interest.

3. Proposed GazeGuide

This section describes the design of GazeGuide in detail (Figure 2 shows an overview of GazeGuide). Raw gaze data are inherently noisy and thus not suitable for direct use [20,42] in GazeGuide as the input for interaction. Therefore, to maneuver the camera, it is more suitable to use filtered data, particularly fixations that are smoother than saccades to avoid the jerky movements of the camera. Hence, the raw gaze data from the eye tracker were filtered using the I-DT algorithm [36] into fixations in real time (we used a time interval of 200 ms). The I-DT algorithm worked with the x and y data, and two fixed thresholds. It filtered the raw gaze data from the eye tracker into fixations. The filtered gaze data $Gaze(eye_x, eye_y)$ were then used as the input modality to maneuver the camera in a 3D space, as detailed in Algorithm 1.

GazeGuide includes two major parts (Figure 2): User-End and UAV-End; these two parts were connected over a network.

User-End: User-End contains real-time video streaming on the HMD, which is integrated with the eye tracker. The eye tracker provides the eye-gaze data as a normalized 2D coordinate, which varies between 0 and 1. These normalized 2D gaze data are mapped onto the pitch and yaw control inputs for the UAV camera.

When the user is ideally looking at the center of the field of view, the gaze value will be $(0.5, 0.5)$. The implemented GazeMapper receives these $Gaze(eye_x, eye_y)$ eye-tracker values to map to the corresponding camera pitch ($Pitch_x^{map}$) and yaw (Yaw_y^{map}) movements, as shown in Figure 3.

UAV-End: UAV-End contains the UAV camera and receives the corresponding mapped ($Pitch_x^{map}$, Yaw_y^{map}) values over the network from User-End. The UAV camera maneuvers according to the user's eye movements at a constant angular speed (0.1°/s).

Algorithm 1 GazeGuide Maneuvering Algorithm

```
1. Retrieve RawGazeData ← Eye tracker.
2. Subjected to Fixation algorithm ← RawGazeData.
3. $Gaze(eye_x, eye_y)$ ← Fixation data.
4. GazeMapper ← $Gaze(eye_x, eye_y)$.
       $FoV = (\Theta, \Phi)$                                        (1)
       $Pitch_x^{map} = (eye_x(C\Theta)) - ((C\Theta)/2))$            (2)
       $Yaw_y^{map} = (eye_y(C\Phi)) - ((C\Phi)/2))$                  (3)
   where $FoV$ is the field of view ($\Theta$ = 120°, $\phi$ = 60°) and C is a multiplicative constant (here, 10).
5. Camera Controller ← ($Pitch_x^{map}$, $Yaw_y^{map}$)
       $UAV_{input}^{cam} = (Yaw_y^{map}, Roll, Pitch_x^{map})$       (4)
   where $UAV_{input}^{cam}$ is the camera controller input to maneuver the UAV camera and $Roll$ is the constant value ($Roll$ = 0) throughout the experiments.
```

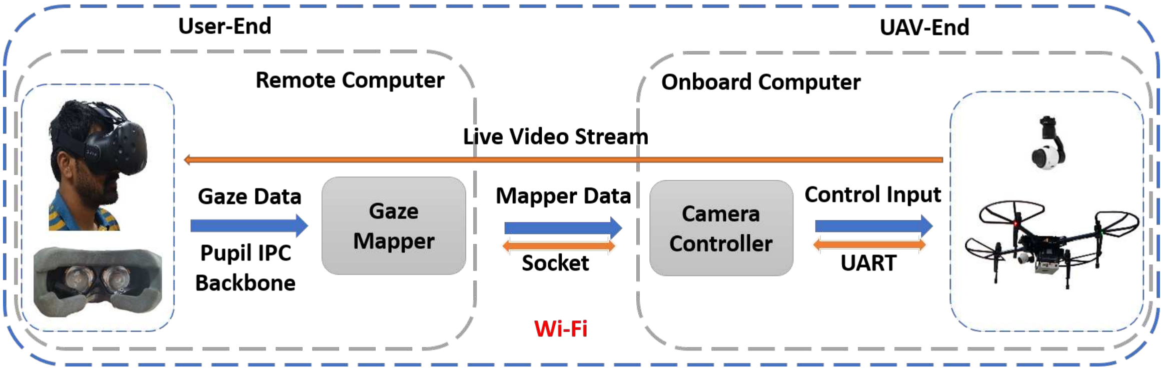

Figure 2. Overview of the proposed GazeGuide.

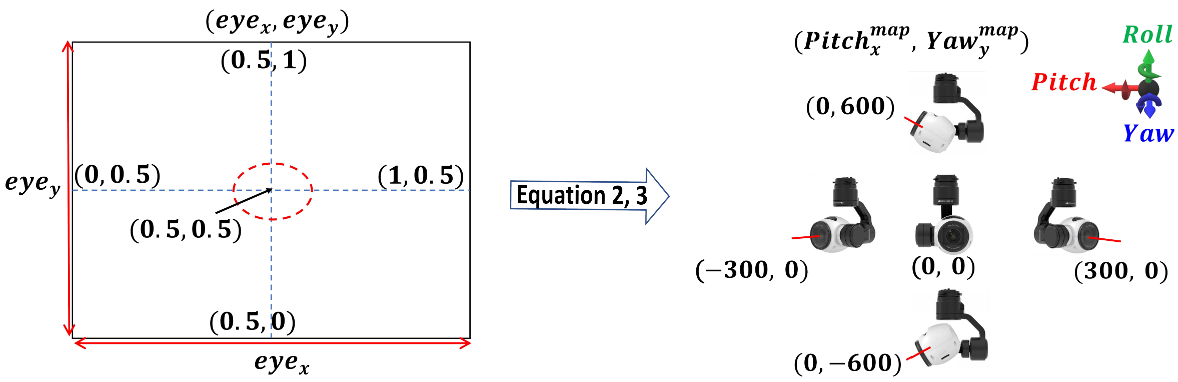

Figure 3. GazeMapper: Mapping of eye-gaze values to camera-controlled input.

4. User Study

We conducted a user study to evaluate the usefulness of the proposed GazeGuide by considering a surveillance and monitoring application-based experiment. The first objective was to compare GazeGuide's performance with that of the UAV's remote controller (RC)-based control method. The second was to investigate the usability of GazeGuide.

4.1. Apparatus, Experiment, and Experimental Design

4.1.1. Apparatus

Our hardware equipment for the implementation of GazeGuide included the following:

1. A laptop running on Windows 10 operating system with Intel (R) Core (TM) i9-8950 HK, 32-GB random access memory, and an NVIDIA Geforce GTX 1080 GPU (remote computer at User-End).
2. A pair of Pupil Labs eye trackers [7] with a binocular mount for the HTC vive HMD (virtual reality setup) running at 120 Hz connected to the laptop.
3. A DJI Zenmuse-X3 gimbal (Table 2) camera mounted on the DJI Matrice-100 UAV with v.1.3.1.0 firmware and the TB47D battery.
4. DJI Manifold [57] based on the ARM architecture with Quad-Core running on Ubuntu 14.04 LTS with 2-GB random access memory mounted (as the onboard computer at UAV-End) and connected to UAV through a UART cable.

Table 2. Technical specifications of DJI Zenmuse-X3 gimbal camera.

Parameters	Values
Weight	247 g
Field of View	94°
Controllable Range	Pan: ±320°
Maximum Controllable Speed	Pan: 180°; Tilt: 120°

The software setup included the following:

1. User-End: Pupil Lab's APIs were used to roll our GazeMapper (Algorithm 1) in Python. OpenCV was used to stream real-time video from the UAV's camera to the HMD.
2. UAV-End: DJI provided a well-documented software development kit (SDK) and APIs to control the UAV and the mounted gimbal camera. Hence, the camera controller was implemented by DJI SDK using an open-source robot operating system [58] framework and its ecosystem in Python. All the data exchange between the different parts of GazeGuide was achieved over a wireless network through the TCP/IP sockets.

4.1.2. Experiment

We conducted the user study with two different experiments.

Experiment 1: A static target gazing experiment was conducted for the evaluation with an objective to bring the targets to the center of view (Figure 4). During the experiment, the user was asked to gaze at the seven static targets, which were positioned in the user's field of view. The user was instructed to gaze at the center of the numbers with four different variations in directions (Table 3). These variations were introduced to evaluate the performance of the GazeGuide in simple and complex operations. During the experiment, the user was instructed to hold his/her gaze at the target for a gaze duration of at least 1 s, therefore ensuring that the camera maneuvered in the direction of the fixation.

Table 3. Experiment 1: Direction variation.

Order	Gaze Target
Left-to-Right (L–R)	1 → 2 → 3 →4 → 5 → 6 → 7
Front-to-Back (F–B)	1 → 7 → 2 →6 → 3 → 5 → 4
Right-to-Left (R–L)	7 → 6 → 5 →4 → 3 → 2 → 1
Back-to-Front (B–F)	4 → 3 → 5 →2 → 6 → 1 → 7

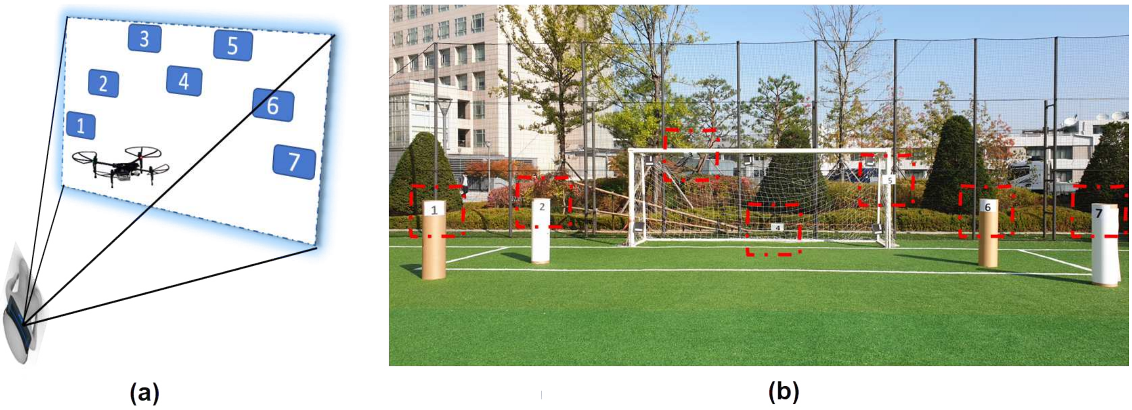

Figure 4. Experiment 1: (**a**) Conceptual design; (**b**) Physical target setup in an outdoor environment.

Experiment 2: A moving target-based experiment with a total track length of 37.1 m was designed in an outdoor environment (Figure 5) by considering three ToIs with an objective to maintain the ToI in the center of the view. Three ToIs moved in their unique predefined pattern (Figure 6). The experiment was carried out with two variations:

Variation 1: ToIs were asked to move over the track individually one after another until they completed their patterns (here, the start and the end points were the same).

Variation 2: ToIs were asked to move simultaneously in their respective patterns. During the experiment, each ToI was tracked for a particular duration before switching to another ToI; the process was continued for a specific time duration (here, 3 min, divided into 30 s for each ToI and tracked twice).

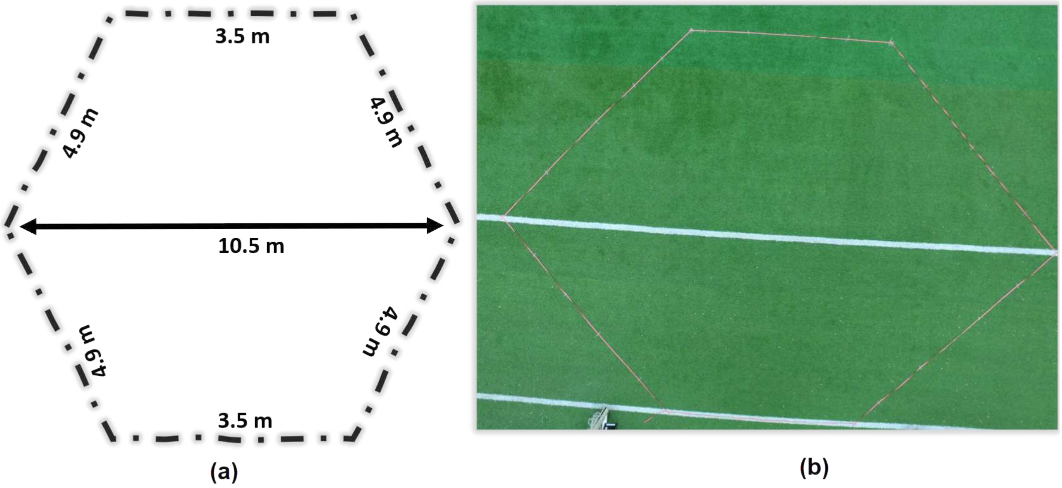

Figure 5. Experiment 2: (**a**) Conceptual; (**b**) Physical design.

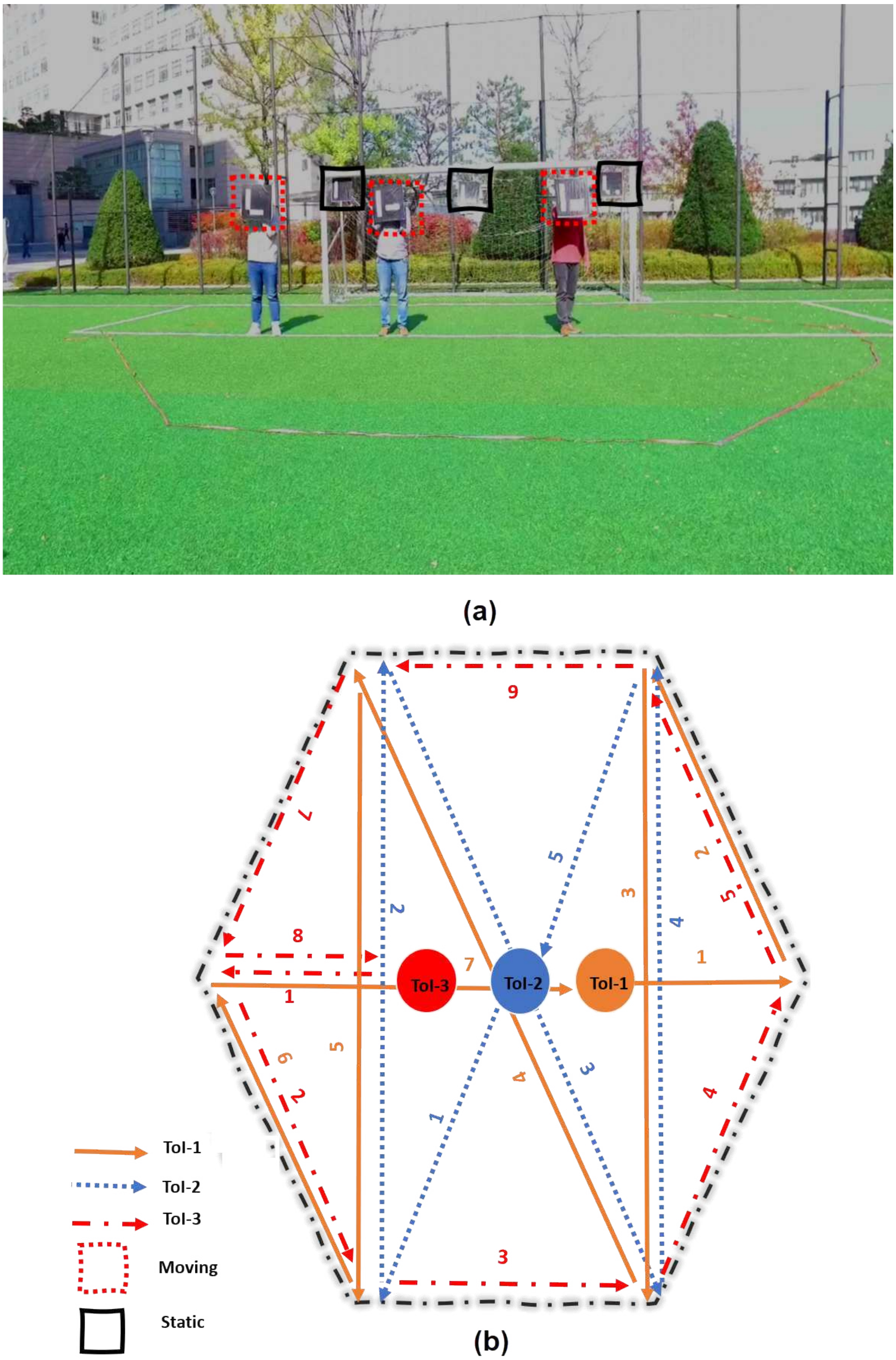

Figure 6. Experiment 2: (**a**) ToI physical setup in an outdoor environment; (**b**) ToI pattern.

4.1.3. Experimental Design

Experiments 1 and 2 of the user study had a within-subject design in repeated measures with the major independent variable being the camera control method; therefore, there were two control methods (GazeGuide and RC). The participants performed each experiment including the variations with three trials.

4.2. Participants

A group of 15 participants volunteered to take part (male: 10; female: 5) in this study. The age of the participants ranged from 25 to 36 years ($M = 29.6$ and $SD = 3.65$); these participants had normal/corrected vision, and their eye-gaze could be calibrated successfully. All the participants were regular computer users (at least 4 h per day), and a few had experience in immersive virtual reality and eye tracking.

4.3. Procedure

The participants performed each experiment individually. Before starting the experiment, the participants were briefed about the nature of the experiment and provided an introduction and the instructions on how to use GazeGuide and RC to maneuver the camera. After familiarizing themselves with the control methods, the participants sat 14 m away in front of the physical setup in an outdoor environment (Figure 7). They put on the HMD and ensured that it fit firmly. After calibration, the participants were allowed to practice for some time. Once they were confident with the control methods, the experiment was carried out. The participants were also informed that they could stop the experiment whenever they wanted to.

The video stream from the UAV camera when each participant was performing the experiment with different camera controls was recorded for further measurement. After the end of the experiment, the participants completed a post-experiment questionnaire and provided feedback on their impressions of each interaction method.

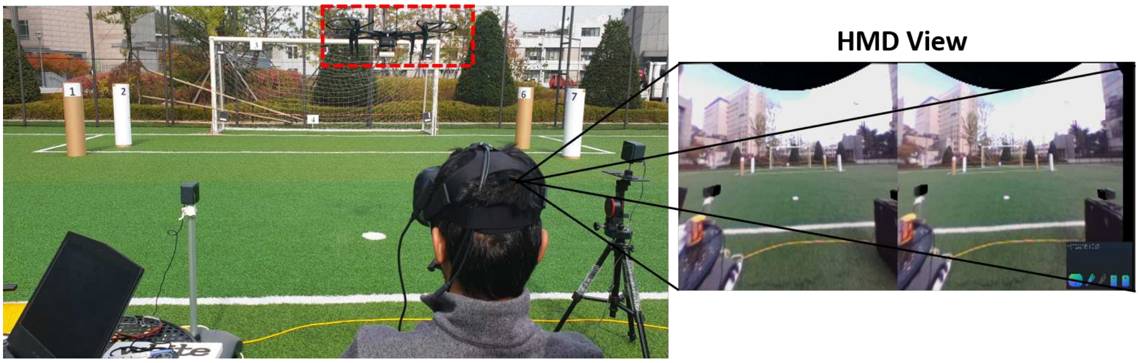

Figure 7. Experimental setup and participant performing the experiment.

4.4. Calibration of the Eye Tracker and UAV

Each participant performed the calibration procedure after wearing and firmly adjusting the HMD. The standard calibration procedure consisted of targets (9-point) at various positions spanning the HMD's visual field. The UAV compass was calibrated with the payload (Table 4) by using the DJI calibration procedure mentioned in [59] in the experimental environment to achieve stabilized flight during the experiments. DJI Matrice ensured stabilized flight of up to 12 min with a payload of 1000 g.

Table 4. Payload distribution of the devices mounted on the UAV.

Component	Weight (g)
Camera	247
Manifold	200
Wi-Fi Dongle	80
Propeller Guard	200
Total	**727**

5. Results and Discussion

Here, we report the participants' performance during the experiment by conducting a statistical analysis on the observed and the recorded data in three parts. First, we describe the performance analysis of Experiment 1 and then that of Experiment 2. Lastly, the subjective analysis of the user experience of GazeGuide and RC is presented.

5.1. Experiment 1

In this experiment, efficiency (time to complete) was considered the major objective measure for the evaluation. The overall time to complete the experiment was noted down for each participant. A statistical analysis was conducted on both the interaction methods.

Figure 8 shows the required mean time to complete the experiment including all the four variations (Table 3). On the basis of the time-to-complete data of the specified experiment, we inferred that for variation order 1 (L-R), GazeGuide took 39.72% ($M = 47.20$, $SD = 5.80$) less time than RC ($M = 70.60$, $SD = 10.73$). In variation order 2 (F-B), GazeGuide took 45.15% ($M = 49.67$, $SD = 7.02$) less time than RC (($M = 78.65$, $SD = 13.85$), whereas in variation order 3 (R-L), GazeGuide took 42.28% ($M = 43.61, SD = 8.25$) less time than RC ($M = 67.00$, $SD = 16.79$). For variation order 4 (B-F), GazeGuide took 59.85% ($M = 40.66, SD = 7.91$) less time than RC ($M = 75.41, SD = 12.14$) to complete the experiment. Overall, GazeGuide took 46.75% less time-to-complete the specific experiment than RC.

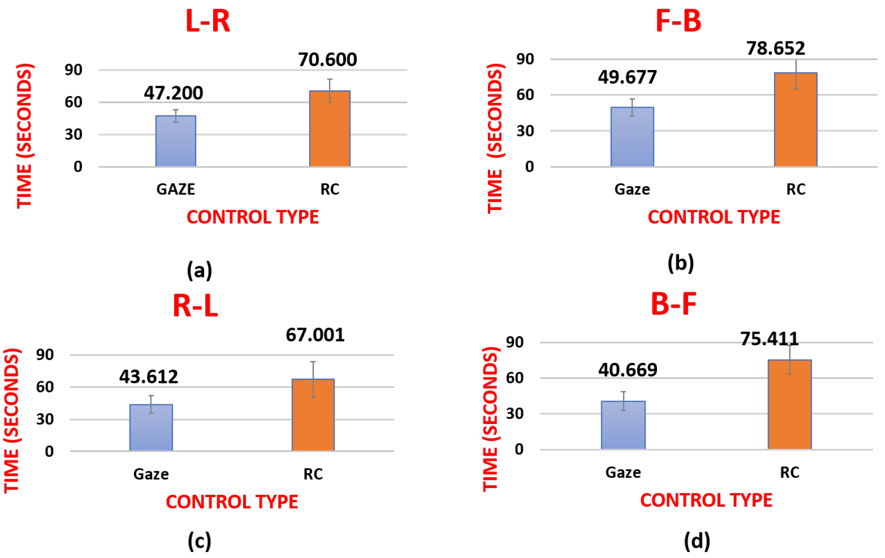

Figure 8. Mean time-to-complete Experiment 1 with: (**a**) variation 1 (L-R), (**b**) variation 2 (B-F), (**c**) variation 3 (R-L), and (**d**) variation 4 (B-F).

Further, the paired *t*-test showed a highly significant difference in the time to complete the experiment using GazeGuide and RC. For variation 1 (L-R), $T(14) = -5.95$, $p = 0.00021$, variation 2 (F-B) $T(14) = -5.69$, $p = 0.00029$, variation 3 (R-L) $T(14) = -3.82$, $p = 0.0040$, and variation 4 (B-F) $T(14) = -5.69, p = 0.00029$. Overall (Figure 8), we can clearly see that when using GazeGuide, the participants significantly took less time to complete the experiment than when using RC.

5.2. Experiment 2

This experiment was aimed to measure the focus precision (accuracy) and change in viewpoint for the specified proposed GazeGuide and RC methods.

Accuracy: For the effective focus precision analysis of the experiment, unique ArUco markers were attached to the ToIs. These moving markers (highlighted in red in Figure 6) specified the position of the ToI in a given frame by using the frame-by-frame marker detection algorithm (Algorithm 2). Focus precision is the distance between the center of the frame ($scene_{center} = 960 \times 540$ in $pixels$) and the marker position.

Algorithm 2 Frame-by-Frame Marker Detection Algorithm

```
1.  Input video ← OpenCV frame extractor.
2.  Identify reference marker in extracted frames.
3.  Compute position ($ref^{pos}_{mark} = (x, y)$) of the reference marker in extracted frames.
4.  Record and continue the process until last frame.
```

Variation 1: Figure 9 shows the focus precision from the center of the frame to the position of the ToI marker. In variation 1, each ToI was tracked until the completion of its respective pattern.

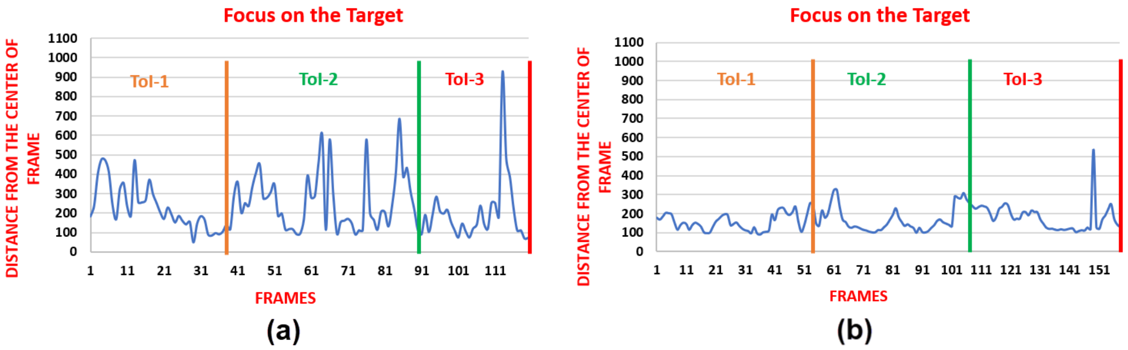

Figure 9. Experiment 2 focus precision (pixels) for variation 1 (**a**) RC, (**b**) GazeGuide.

Figure 9 represents the focus precision during the tracking of each ToI. It clearly shows that in the case of RC, more deviation occurred because of the abrupt movement/control of the camera. In contrast, GazeGuide provided a very smooth maneuvering functionality to control the camera in a natural and effortless manner. Hence, the achieved focus precision was very less (540 pixels was the maximum observed focus precision) than RC (940 pixels was the maximum observed focus precision).

Variation 2: Here, all the ToIs were moving simultaneously in their respective patterns, thus giving an impression to the participant as if the ToIs were moving randomly. The participants tracked each ToI for a specific interval before switching to the next ToI. This cycle was repeated twice in the 3-min duration.

Figure 10 clearly shows that the maximum focus precision was higher in the case of RC (1000 pixels) than in that of GazeGuide (650 pixels). Therefore, we inferred that throughout the experiment, ToI was tracked successfully with a significantly low focus precision. Figure 11 presents the overall focus precision for both the variations. It clearly shows that for variation 1, GazeGuide ($SD = 137.89$) achieved 48.96% more accuracy than RC ($SD = 227.29$), whereas for variation 2, GazeGuide ($SD = 60.59$) achieved 81.52% more accuracy than RC ($SD = 143.98$). Upon considering the overall performance of the participants, we concluded that GazeGuide achieved significantly outstanding performance even when the ToIs moved simultaneously.

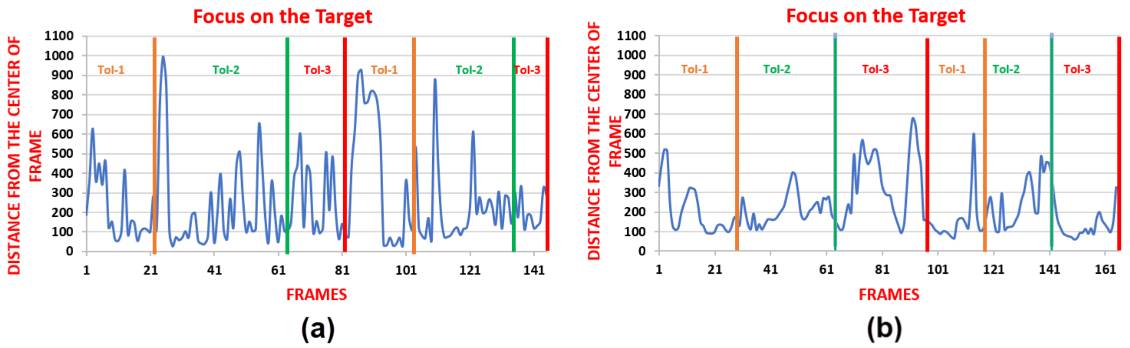

Figure 10. Experiment 2 focus precision (pixels) for variation 2 (**a**) RC, (**b**) GazeGuide.

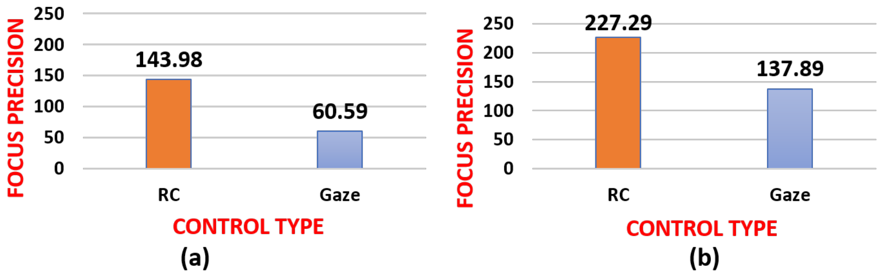

Figure 11. Focus precision (pixels) for (**a**) variation 1, (**b**) variation 2.

Change in Viewpoint: To compute the change in viewpoint, unique ArUco markers were placed at different locations within the field of view. These static markers (highlighted in black in Figure 6) specified the position of the static markers in a given frame by using the frame-by-frame marker detection Algorithm 2. These positions indicated the changes in the viewpoint (x, y in pixels).

Figure 12 illustrates the obtained viewpoint changes during variation 1 on the basis of the static marker position (x, y) in the frame. We observed that there were more viewpoint changes and abrupt movements in the case of the RC method than in that of GazeGuide, mainly because of two reasons: (1) GazeGuide adopted incremental changes in the camera movement on the basis of the eye-gaze, whereas the RC method led to random changes in the camera movement depending on the participants' response time. (2) GazeGuide was an algorithmic implementation and did not anticipate the ToIs' next position, thus enabling the camera to follow the target smoothly, whereas in the RC method, the participants inherently anticipated the next position of the ToI and moved the camera well in advance abruptly.

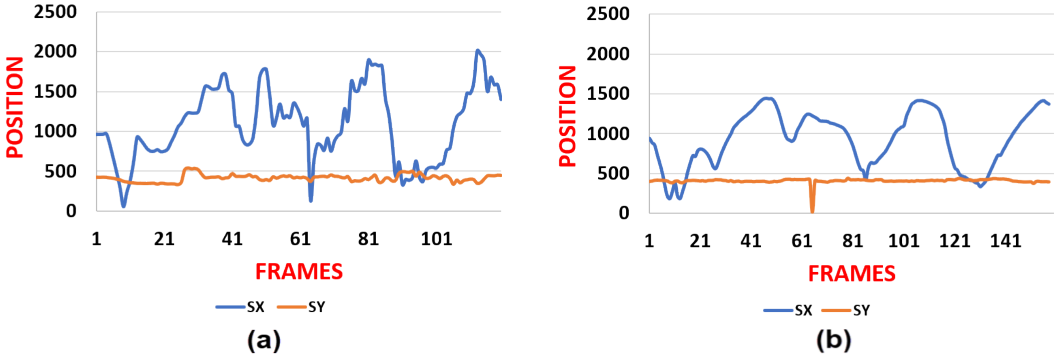

Figure 12. Change in viewpoint for variation 1 (pixels) (**a**) RC, (**b**) GazeGuide.

Similarly, Figure 13 presents the viewpoint changes during variation 2 on the basis of the static marker position in the frame (x, y). Overall, we inferred that the participants experienced a graceful and smooth transition in viewpoint changes when performing the experiment using GazeGuide to track the ToIs.

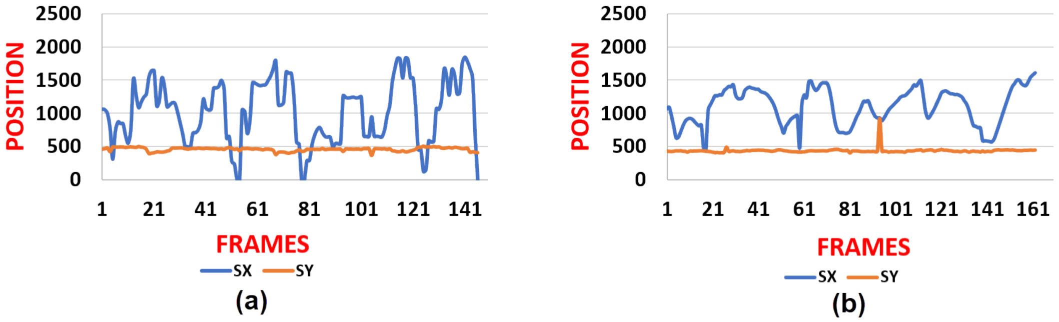

Figure 13. Change in viewpoint for variation 2 (pixels) (**a**) RC, (**b**) GazeGuide.

5.3. Qualitative Results

After the completion of the experiments, the participants were asked to fill out a qualitative feedback questionnaire on the proposed GazeGuide. In the feedback, the participants responded to the questions in the questionnaire according to a five-point Likert scale. Besides these questions, the participants were required to express their preference with the two control methods on the basis of their experimental experience. Most of the participants preferred GazeGuide. The following are the questions used for the qualitative study:

- Q1: It was easy to get familiar with the framework and how it worked.
- Q2: It was easy to identify targets.
- Q3: I did not have to pay much attention to the camera control method.
- Q4: I was able to focus on the experiment actively.
- Q5: I did not feel distracted while using the framework.
- Q6: I experienced eye strain, discomfort, headache, and fatigue while using the framework.

Table 5 presents the results of the statistical analysis of the subjective responses from all the participants by using a paired *t*-test. With almost all of the ratings falling between neutral and strongly agree, it clearly shows that there were no negative impressions about the proposed framework. Most of the participants felt that they could learn GazeGuide (*M* = 4.16, *SD* = 0.73) more quickly (Q1) than RC (*M* = 2.74, *SD* = 1.01). Furthermore, for Questions 2, 3, and 4, the participants' ratings showed that they did not have to pay much attention to camera control in order to participate actively in the experiment and to find the targets using GazeGuide (Q2(*M* = 3.75, *SD* = 0.88), Q3(*M* = 4.80, *SD* = 0.27), and Q4(*M* = 4.73, *SD* = 0.40)) as compared to RC (Q2(*M* = 2.63, *SD* = 1.07), Q3(*M* = 2.80, *SD* = 0.52), and Q4(*M* = 3.03, *SD* = 0.60)). This showed that the participants had lesser cognition on camera control during the experiment. As the camera was maneuvered according to their current eye movements, the focus was on the ToI rather than the control. All of the participants experienced no distraction (Q5) during the experiment, because there was less physical movement in GazeGuide (*M* = 4.06, *SD* = 0.49) than in RC (*M* = 3.00, *SD* = 0.71). Q6 measured the motion sickness aspect of the proposed GazeGuide. Novice participants experienced a little motion sickness; however, the overall participants' result (*M* = 3.23, *SD* = 0.41) showed that motion sickness was neutral. From Table 5, we can say that the proposed GazeGuide scales (average of the mean value = 4.3) higher than the RC (average of the mean value = 2.84) on five-point scale. Finally, these results proved that GazeGuide is very intuitive and effective and has considerable potential as an input modality in HRI.

Table 5. Subjective questionnaire results (five-point scale: 1—strongly disagree, 2—disagree, 3—neutral, 4—agree, 5—strongly agree).

Questions	Control Type				Paired *t*-Test	
	GazeGuide		RC			
	Mean	SD	Mean	SD	T(14)	P
Q1	4.16	0.73	2.74	1.01	4.59	0.00041
Q2	3.75	0.88	2.63	1.07	3.54	0.00324
Q3	4.80	0.28	2.80	0.52	18.3	1.73×10^{-11}
Q4	4.73	0.40	3.03	0.60	8.50	6.7×10^{-7}
Q5	4.06	0.49	3.00	0.71	4.30	0.00073
Q6	3.23	0.41	NA	NA	NA	NA

6. Response Time

The measured response time from the eye movement to the corresponding camera control input was less than 220 ms, which was fairly tolerable in the experiments. This total response time included the following:

1. 200 ms (introduced time interval) for the eye fixation on the target.
2. A transmission delay (TCP/IP) between User-End and UAV-End of ~20 ms.

7. Applications

UAVs are becoming increasingly refined in terms of their hardware capabilities. Improvements in sensor systems, on-board computational platforms, energy storage, and other technologies have made it possible to build a huge variety of UAVs for different applications [60], such as aerial surveillance, environmental monitoring, traffic monitoring, precision agriculture and inspection of agricultural fields, aerial goods delivery, natural disaster areas surveillance, and forest, border, and fire monitoring [61,62]. In these applications, UAVs are equipped with some specific sensors (LiDAR and cameras) to continuously monitor the ground ToIs. Basically, in video-based surveillance and monitoring applications, the onboard camera provides a rich visual feed to the monitor. These application tasks can be very exhausting for a human operator because of such rich visual feeds. Hence, any reduction of the operator workload frees the cognitive capacity for the actual surveillance task. In video-based surveillance and monitoring applications, a frequently occurring task is to keep track of a moving target. If the target moves out of the displayed scene, then the operator has to readjust the camera viewpoint to bring the target within the scene. Moreover, aerial cinematography systems require considerable attention and effort for the simultaneous identification of the actor, the predication of how the scene is going to evolve, and UAV control; thus, avoiding obstacles and reaching the expected viewpoints is very challenging. In such situations, GazeGuide appears compelling and keeps the target focused just by looking at the target. Hence, the operator's visual attention could be focused on the primary (surveillance) task, and the secondary (navigational) task is achieved effortlessly at the same time.

8. Challenges Faced During Implementation of GazeGuide

Eye trackers, although in their infancy, are very sensitive instruments. The Pupil Labs eye tracker used for this study has some limitations. Although it had good accuracy, calibration was required every time the tracker was used, making it difficult to test. The eye cameras often heated up, causing jitters in the gaze data. We did not attempt to measure this error; rather, we just re-calibrated the Pupil every time this occurred. We could achieve acceptable results (to the participants) with the Pupil eye tracker, which was the economical and best solution.

9. Conclusion and Future Work

In this paper, we presented GazeGuide, a novel framework that controls UAV camera maneuvering through the eye-gaze of a remote user. We described the implementation of GazeGuide and reported on an experimental study. A surveillance and monitoring application-based experimental set up was prepared in an outdoor environment, and a user study with 15 participants was conducted through both quantitative and qualitative measures. The objective of the evaluation was to compare the efficiency of the proposed GazeGuide against that of the RC-based control method. The quantitative measure showed that the participants using GazeGuide performed significantly better than those using the RC-based control. In addition, the qualitative measure provided clear evidence that GazeGuide performed significantly better than the RC-based control. Overall, these measures revealed that the GazeGuide performed well under all the selected conditions as an input modality in HRI.

The gimbal camera used in this research was mounted on a UAV for a specific reason. The camera behaved like a human eyeball, and the UAV behaved like the human head while tracking an object. The current research focused on the maneuvering of the camera with the UAV fixed in a predefined direction. In the future, we intend to explore how GazeGuide would behave when both the camera (eyeball) and the UAV (head) are moving. The future work will also include enhancements in providing more degrees-of-freedom to control the camera by integrating interactions using eye blinks and other combinations.

Author Contributions: Conceptualization and methodology: P.K.B.N., A.B., and Y.H.C.; software, validation, formal analysis, and investigation: P.K.B.N., A.B., C.B., and A.K.P.; writing-original draft preparation: P.K.B.N.; writing-review and editing: P.K.B.N., A.K.P., and A.B.; data curation: P.K.B.N., A.B., A.K.P., and C.B.; supervision, project administration: A.K.P. and Y.H.C.; funding acquisition: Y.H.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Ministry of Science & ICT of Korea, under the Software Star Lab. program (IITP-2018-0-00599) supervised by the Institute for Information & Communications Technology Planning & Evaluation and the Chung-Ang University research grant in 2020.

Acknowledgments: We would like to thank Bharatesh Chakravarthi B.C. and Jae Yeong Ryu for their time and support in conduction of the experiments.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report with respect to this paper.

Abbreviations

The following abbreviations were used in this manuscript:

UAV	Unmanned Aerial Vehicle
HMD	Head-Mounted Display
HRI	Human-Robot Interaction
HCI	Human-Computer Interaction
ToI	Target of Interest
ToIs	Target of Interests
RC	Remote Controller

References

1. Decker, D.; Piepmeier, J.A. Gaze tracking interface for robotic control. In Proceedings of the 2008 40th Southeastern Symposium on System Theory (SSST), New Orleans, LA, USA, 16–18 March 2008; pp. 274–278.
2. Zhai, S. What's in the eyes for attentive input. *Commun. ACM* **2003**, *46*, 34–39. [CrossRef]
3. Yu, M.; Wang, X.; Lin, Y.; Bai, X. Gaze tracking system for teleoperation. In Proceedings of the 26th Chinese Control and Decision Conference (2014 CCDC), Changsha, China, 31 May–2 June 2014; pp. 4617–4622.
4. Lopez-Basterretxea, A.; Mendez-Zorrilla, A.; Garcia-Zapirain, B. Eye/head tracking technology to improve HCI with iPad applications. *Sensors* **2015**, *15*, 2244–2264. [CrossRef] [PubMed]

5. Alapetite, A.; Hansen, J.P.; MacKenzie, I.S. Demo of gaze controlled flying. In Proceedings of the 7th Nordic Conference on Human-Computer Interaction (NordiCHI), Copenhagen, Denmark, 14–17 October 2012; pp. 773–774.
6. Zhang, G.; Hansen, J.P.; Minakata, K. Hand-and gaze-control of telepresence robots. In Proceedings of the 11th ACM Symposium on Eye Tracking Research & Application, Denver, CO, USA, 25–28 June 2019; p. 70.
7. Kassner, M.; Patera, W.; Bulling, A. Pupil: An open source platform for pervasive eye tracking and mobile gaze-based interaction. In Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication, Seattle, WA, USA, 13–17 September 2014; pp. 1151–1160.
8. Zhang, X.; Sugano, Y.; Bulling, A. Evaluation of Appearance-Based Methods and Implications for Gaze-Based Applications. *arXiv* **2019**, arXiv:1901.10906.
9. Williams, T.; Szafir, D.; Chakraborti, T.; Amor, H.B. Virtual, augmented, and mixed reality for human-robot interaction. In Proceedings of the Companion of the 2018 ACM/IEEE International Conference on Human-Robot Interaction, Chicago, IL, USA, 5–8 March 2018, pp. 403–404.
10. Admoni, H.; Scassellati, B. Social eye gaze in human-robot interaction: A review. *J. Hum.-Robot Interact.* **2017**, *6*, 25–63. [CrossRef]
11. Argyle, M. Non-verbal communication in human social interaction. In *Non-Verbal Communication*; Cambridge University Press: Cambridge, UK, 1972.
12. Goldin-Meadow, S. The role of gesture in communication and thinking. *Trends Cognit. Sci.* **1999**, *3*, 419–429. [CrossRef]
13. Dautenhahn, K. Methodology & themes of human-robot interaction: A growing research field. *Int. J. Adv. Robot. Syst.* **2007**, *4*, 15.
14. Olsen, D.R.; Goodrich, M.A. Metrics for evaluating human-robot interactions. In Proceedings of the PERMIS, Gaithersburg, MD, USA, 16–18 September 2003; Volume 2003, p. 4.
15. Christiansen, M.; Laursen, M.; Jørgensen, R.; Skovsen, S.; Gislum, R. Designing and testing a UAV mapping system for agricultural field surveying. *Sensors* **2017**, *17*, 2703. [CrossRef]
16. Kumar, G.A.; Patil, A.K.; Patil, R.; Park, S.S.; Chai, Y.H. A LiDAR and IMU integrated indoor navigation system for UAVs and its application in real-time pipeline classification. *Sensors* **2017**, *17*, 1268. [CrossRef]
17. B. N., P.K.; Patil, A.K.; B., C.; Chai, Y.H. On-Site 4-in-1 Alignment: Visualization and Interactive CAD Model Retrofitting Using UAV, LiDAR's Point Cloud Data, and Video. *Sensors* **2019**, *19*, 3908.
18. Savkin, A.V.; Huang, H. Proactive deployment of aerial drones for coverage over very uneven terrains: A version of the 3D art gallery problem. *Sensors* **2019**, *19*, 1438. [CrossRef]
19. Zhai, S.; Morimoto, C.; Ihde, S. Manual and gaze input cascaded (MAGIC) pointing. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Pittsburgh, PA, USA, 15–20 May 1999; pp. 246–253.
20. Jakob, R. The use of eye movements in human-computer interaction techniques: what you look at is what you get. In *Readings in Intelligent User Interfaces*; Morgan Kaufmann: Burlington, MA, USA, 1998; pp. 65–83.
21. Macrae, C.N.; Hood, B.M.; Milne, A.B.; Rowe, A.C.; Mason, M.F. Are you looking at me? Eye gaze and person perception. *Psychol. Sci.* **2002**, *13*, 460–464. [CrossRef] [PubMed]
22. Mohamed, A.O.; da Silva, M.P.; Courboulay, V. A History of Eye Gaze Tracking. 2007. Available online: https://hal.archives-ouvertes.fr/hal-00215967/document (accessed on 29 February 2020).
23. Wang, K.; Ji, Q. Real time eye gaze tracking with 3d deformable eye-face model. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1003–1011.
24. Krafka, K.; Khosla, A.; Kellnhofer, P.; Kannan, H.; Bhandarkar, S.; Matusik, W.; Torralba, A. Eye tracking for everyone. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 2176–2184.
25. Li, B.; Fu, H.; Wen, D.; Lo, W. Etracker: A mobile gaze-tracking system with near-eye display based on a combined gaze-tracking algorithm. *Sensors* **2018**, *18*, 1626. [CrossRef]
26. Sibert, L.E.; Jacob, R.J.K. Evaluation of eye gaze interaction. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, The Hague, The Netherlands, 1–6 April 2000; pp. 281–288.
27. Ishii, H.; Okada, Y.; Shimoda, H.; Yoshikawa, H. Construction of the measurement system and its experimental study for diagnosing cerebral functional disorders using eye-sensing HMD. In Proceedings of the 41st SICE Annual Conference, Osaka, Japan, 5–7 August 2002; Volume 2, pp. 1248–1253.

28. Morimoto, C.H.; Mimica, M.R.M. Eye gaze tracking techniques for interactive applications. *Comput. Vis. Image Underst.* **2005**, *98*, 4–24. [CrossRef]
29. Kumar, P.; Adithya, B.; Chethana, B.; Kumar, P.A.; Chai, Y.H. Gaze-controlled virtual retrofitting of UAV-scanned point cloud data. *Symmetry* **2018**, *10*, 674.
30. Lee, H.C.; Luong, D.T.; Cho, C.W.; Lee, E.C.; Park, K.R. Gaze tracking system at a distance for controlling IPTV. *IEEE Trans. Consum. Electron.* **2010**, *56*, 2577–2583. [CrossRef]
31. Pfeiffer, T. Towards gaze interaction in immersive virtual reality: Evaluation of a monocular eye tracking set-up. In Proceedings of the Virtuelle und Erweiterte Realität-Fünfter Workshop der GI-Fachgruppe VR/AR, Aachen, Germany, 1 September 2008.
32. B., A.; B. N., P.K.; Chai, Y.H.; Patil, A.K. Inspired by Human Eye: Vestibular Ocular Reflex Based Gimbal Camera Movement to Minimize Viewpoint Changes. *Symmetry* **2019**, *11*, 101. [CrossRef]
33. Kar, A.; Corcoran, P. A review and analysis of eye-gaze estimation systems, algorithms and performance evaluation methods in consumer platforms. *IEEE Access* **2017**, *5*, 16495–16519. [CrossRef]
34. Sugano, Y.; Bulling, A. Self-calibrating head-mounted eye trackers using egocentric visual saliency. In Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology, Charlotte, NC, USA, 8–11 November 2015; pp. 363–372.
35. Adithya, B.; Lee H.; Pavan Kumar, B.N.; Chai, Y. Calibration techniques and gaze accuracy estimation in pupil labs eye tracker. *TECHART J. Arts Imaging Sci.* **2018**, *5*, 38–41.
36. Salvucci, D.D.; Joseph, H. Goldberg. Identifying fixations and saccades in eye-tracking protocols. In Proceedings of the 2000 Symposium on Eye Tracking Research & Applications, Florida, FL, USA, 6–8 November 2000; pp. 71–78.
37. Jacob, R.J.K.; Karn, K.S. Eye tracking in human-computer interaction and usability research: Ready to deliver the promises. In *The Mind's Eye: Cognitive and Applied Aspects of Eye Movement Research*; Elsevier: Amsterdam, The Netherlands, 2003; pp. 573-605.
38. Latif, H.O.; Sherkat, N.; Lotf, A. TeleGaze: Teleoperation through eye gaze. In Proceedings of the 2008 7th IEEE International Conference on Cybernetic Intelligent Systems, London, UK, 9–10 September 2008; pp. 1–6.
39. Latif, H.O.; Sherkat, N.; Lotfi, A. Teleoperation through eye gaze (TeleGaze): A multimodal approach. In Proceedings of the 2009 IEEE International Conference on Robotics and Biomimetics (ROBIO), Guilin, China, 19–23 December 2009; pp. 711–716.
40. Lin, C.-S.; Ho, C.; Chen, W.; Chiu, C.; Yeh, M. Powered wheelchair controlled by eye-tracking system. *Opt. Appl.* **2006**, *36*, 401–412.
41. Eid, M.A.; Giakoumidis, N.; el Saddik, A. A novel eye-gaze-controlled wheelchair system for navigating unknown environments: Case study with a person with ALS. *IEEE Access* **2016**, *4*, 558–573. [CrossRef]
42. Akkil, D.; James, J.M.; Isokoski, P.; Kangas, J. GazeTorch: Enabling gaze awareness in collaborative physical tasks. In Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems, San Jose, CA, USA, 7–12 May 2016; pp. 1151–1158.
43. Akkil, D.; Isokoski, P. Comparison of gaze and mouse pointers for video-based collaborative physical task. *Interact. Comput.* **2018**, *30*, 524–542. [CrossRef]
44. Akkil, D.; Isokoski, P. Gaze augmentation in egocentric video improves awareness of intention. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, San Jose, CA, USA, 7–12 May 2016; pp. 1573–1584.
45. Yoo, D.H.; Kim, J.H.; Kim, D.H.; Chung, M.J. A human-robot interface using vision-based eye gaze estimation system. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Lausanne, Switzerland, 30 September–4 October 2002; Volume 2, pp. 1196–1201.
46. Gêgo, D.; Carreto, C.; Figueiredo, L. Teleoperation of a mobile robot based on eye-gaze tracking. In Proceedings of the 2017 12th Iberian Conference on Information Systems and Technologies (CISTI), Lisbon, Portugal, 21–24 June 2017; pp. 1–6.
47. Hansen, J.P.; Alapetite, A.; MacKenzie, I.S.; Møllenbach, E. The use of gaze to control drones. In Proceedings of the Symposium on Eye Tracking Research and Applications, Safety Harbor, FL, USA, 26–28 March 2014; pp. 27–34.

48. Tall, M.; Alapetite, A.; Agustin, J.S.; Skovsgaard, H.H.T.; Hansen, J.P.; Hansen, D.W.; Møllenbach, E. Gaze-controlled driving. In Proceedings of the CHI'09 Extended Abstracts on Human Factors in Computing Systems, Boston, MA, USA, 4–9 April 2009; pp. 4387–4392.
49. Khamis, M.; Kienle, A.; Alt, F.; Bulling, A. GazeDrone: Mobile eye-based interaction in public space without augmenting the user. In Proceedings of the 4th ACM Workshop on Micro Aerial Vehicle Networks, Systems, and Applications, Munich, Germany, 10–15 June 2018; pp. 66–71.
50. Yu, M.; Lin, Y.; Schmidt, D.; Wang, X.; Wang, Y. Human-robot interaction based on gaze gestures for the drone teleoperation. *J. Eye Mov. Res.* **2014**, *7*, 1–14.
51. Kwok, K.-W.; Sun, L.; Mylonas, G.P.; James, D.R.C.; Orihuela-Espina, F.; Yang, G. Collaborative gaze channelling for improved cooperation during robotic assisted surgery. *Ann. Biomed. Eng.* **2012**, *40*, 2156–2167. [CrossRef] [PubMed]
52. Fujii, K.; Gras, G.; Salerno, A.; Yang, G. Gaze gesture based human robot interaction for laparoscopic surgery. *Med. Image Anal.* **2018**, *44*, 196–214. [CrossRef]
53. Rudi, D.; Kiefer, P.; Giannopoulos, I.; Raubal, M. Gaze-based interactions in the cockpit of the future: A survey. *J. Multimodal User Interfaces* **2019**, *14*, 25–48. [CrossRef]
54. Ruhland, K.; Peters, C.E.; Andrist, S.; Badler, J.B.; Badler, N.I.; Gleicher, M.; Mutlu, B.; McDonnell, R. A review of eye gaze in virtual agents, social robotics and hci: Behaviour generation, user interaction and perception. *Comput. Graph. Forum* **2015**, *34*, 299–326. . [CrossRef]
55. Martinez-Hernandez, U.; Boorman, L.W.; Prescott, T.J. Multisensory wearable interface for immersion and telepresence in robotics. *IEEE Sens. J.* **2017**, *17*, 2534–2541. [CrossRef]
56. Hansen, J.P.; Alapetite, A.; Thomsen, M.; Wang, Z.; Minakata, K.; Zhang, G. Head and gaze control of a telepresence robot with an HMD. In Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications, Warsaw, Poland, 14–17 June 2018; pp. 1–3.
57. DJI Manifold as an On-board Computer for UAV. Available Online: https://www.dji.com/kr/manifold (accessed on 29 February 2020).
58. Quigley, M.; Conley, K.; Gerkey, B.; Faust, J.; Foote, T.; Leibs, J.; Wheeler, R.; Ng, A.Y. ROS: An open-source robot operating system. In Proceedings of the ICRA Workshop on Open Source Software, Kobe, Japan, 12–17 May 2009; Volume 3, p. 5.
59. DJI Matrice 100 User Manual. Available Online: https://dl.djicdn.com/downloads/m100/M100_User_Manual_EN.pdf (accessed on 29 February 2020).
60. Bethke, B.; How, J.P.; Vian, J. Group health management of UAV teams with applications to persistent surveillance. In Proceedings of the 2008 American Control Conference, Seattle, WA, USA, 11–13 June 2008; pp. 3145–3150.
61. Meng, X.; Wang, W.; Leong, B. Skystitch: A cooperative multi-UAV-based real-time video surveillance system with stitching. In Proceedings of the 23rd ACM International Conference on Multimedia, Brisbane, Australia, 26–30 October 2015; pp. 261–270.
62. Savkin, A.V.; Huang, H. Asymptotically optimal deployment of drones for surveillance and monitoring. *Sensors* **2019**, *19*, 2068. [CrossRef]