The Use of Gaze to Control Drones

John Paulin Hansen,
PitLab,
IT University of Copenhagen,
Denmark
paulin@itu.dk

Alexandre Alapetite,
Technical University of
Denmark, Department of
Management Engineering
alexandre@alapetite.fr

I. Scott MacKenzie,
Dept. of Electrical
Engineering and Computer
Science, York University,
Canada
mack@cse.yorku.ca

Emilie Møllenbach,
IT University of Copenhagen,
Denmark
emilie@itu.dk

Abstract

This paper presents an experimental investigation of gaze-based control modes for unmanned aerial vehicles (UAVs or "drones"). Ten participants performed a simple flying task. We gathered empirical measures, including task completion time, and examined the user experience for difficulty, reliability, and fun. Four control modes were tested, with each mode applying a combination of *x-y* gaze movement and manual (keyboard) input to control speed (pitch), altitude, rotation (yaw), and drafting (roll). Participants had similar task completion times for all four control modes, but one combination was considered significantly more reliable than the others. We discuss design and performance issues for the gaze-plus-manual split of controls when drones are operated using gaze in conjunction with tablets, near-eye displays (glasses), or monitors.

CR Categories: Information interfaces and presentation (e.g., HCI): Miscellaneous.

Keywords: Drones, UAV, Gaze interaction, Gaze input, multimodality, mobility, head-mounted displays, augmented or mixed reality systems, video gaming, robotics

1 Introduction

Unmanned aerial vehicles (UAVs) or "drones" have a long history in military applications. They are able to carry heavy loads over long distances, while being controlled remotely by an operator. However, low-cost drones are now offering many non-military possibilities. These drones are light-weight, fly only a limited time (e.g., <20 minutes), and have limited range (e.g., < 1 km). For instance, the off-the-shelf A.R. Parrot drone costs around $400 and can be controlled from a PC, tablet, or smartphone (Figure 1). It has a front-facing camera transmitting live-images to the pilot via Wi-Fi. People share videos, tips, and new software applications supported by its open-API.

This paper considers gaze as a potential input in the development of interfaces for drone piloting. When using gaze input for drone control, spatial awareness is directly conveyed without being mediated through another input device. The pilot sees what the drone sees. However, it is an open research question how best to include gaze in the command of drones. If gaze is too difficult to use, people may crash or lose the drones, which is dangerous and
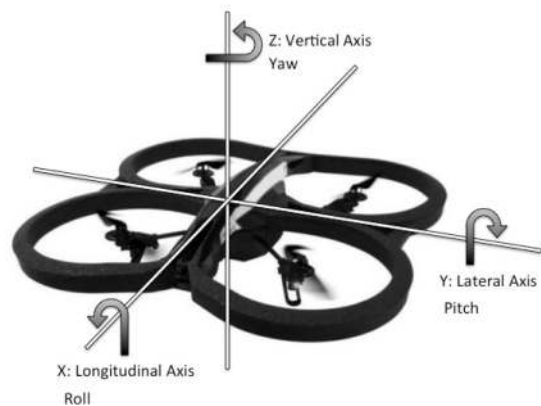

Figure 1 *Commodity drone (A.R. Parrot 2.0). .The three main control axes are pitch, roll and yaw.*

costly.

First we present our motivation with examples of use-cases where gaze could offer a significant contribution. Then we present previous research within the area. We designed an experiment to get feedback from users on their immediate, first-time impression of gaze-controlled flying. The experiment is presented in section 5. The paper then finishes with general discussion on how best to utilize gaze for drone control.

2 Motivation

Why would anyone like to steer a drone with gaze? Gaze interaction has been successful in providing accessibility for people who are not able to use their hands. For instance, disabled people can type and play video games with their eyes only. Drones may offer people with low mobility the possibility to visually inspect areas they could not otherwise see. We imagine a person in a wheelchair who may dispatch a drone to examine inaccessible areas. This could improve, for instance, the experience of hiking in nature areas. Similarly, a paralyzed

person in a hospital bed could participate remotely in home-life activities [Hansen, Agustin, & Skovsgaard 2011].

Drones are increasingly used for professional purposes. For instance, drones are used as a machine on modern farms to optimize the spread of fertilizers and pesticides. Thousands of drone pilots are licensed to inspect and spray paddy fields in Japan, a practice that extends back more than 10 years [Nonami et al. 2010]. Professional photographers can mount a camera on a drone and deliver free-space video data at a modest cost. The view of the camera may be aligned with the direction of travel (i.e., "eyes forward" [Valimont and Chappell 2011]) or the camera may be mounted on a motor frame and turned independently of the drone. In the latter case, the recordings require both a pilot and a cameraman. If controls where less demanding, this task could be performed by one person.

When a pilot must conduct several operations at once, for instance flying and spraying, or commanding multiple drones, there are reasons to consider eye movements as part of an ensemble of inputs, because this would leave hands free for other manual control tasks. For instance, Zhu et al. [2010] examined hands-busy situations in tele-operation activities where gaze could potentially provide support to the operator by controlling the camera.

Gaze tracking has been suggested as a novel game control (e.g., by Isokoski et al. [2009]), similar to popular motion controllers. Pfeil et al. [2013] examined how body motions tracked by a Microsoft Kinect should map to the controls of a drone in order to augment entertainment. Similarly, it may be worthwhile to study how gaze movements should map to drone control for the most pleasurable flying experience.

Several studies have examined gaze as a potential input for interaction with virtual and augmented reality displayed in head mounted displays or Near-Eye Displays (NED´s) (e.g., Tanriverdi and Jacob [2000]). Some experienced drone pilots prefer immersive glasses because they provide full mobility and a more engaging experience. Motivated by the growing popularity of these displays, we finish this paper elaborating on the idea of streaming video from the drone to a NED – immersive or overlaid/transparent – that might then be operated by gaze.

In summary, we have several reasons to consider gaze for control of drones: It offers a direct, immediate mode of interaction where the pilot sees what the drone sees; gaze offers a hands-free control option for people with mobility difficulties; and gaze may assist hands-busy operators when combined with other input modalities.

3 Interaction with drones

When piloting a drone with direct, continuous inputs – as opposed to pre-programmed flying – there are a number of highly dynamic parameters to monitor and control simultaneously, and if there are obstacles or turbulence in the air, reactions must be quick. Often, delays on the data link complicate control. In particular, it is a challenge when the drone and pilot are oriented in different directions. The fact that the drone may hurt someone or break if it crashes also puts extra stress on the pilot – this is not just a computer game!

In manual mode, there are several drone actions to monitor: speed (pitch), altitude, rotation (yaw), and translation (roll). In addition, there may be discrete controls for start (takeoff), emergency

landing, video recording, or other actions depending on the payload. Some of the controls may be partly automatized, such as stabilizing the drone at a particular position or altitude. Some drones offer full-automatized flight modes by setting GPS waypoints on a map and a "return to launch" function. The most common interfaces for civilian drones are tablets/smartphones or a so-called RC transmitter, which is a programmable remote control box with two joysticks, mode switches, and pre-defined buttons (Figure 2).

Figure 2 *RC-transmitter (left) and tablet (right).*
(Left: © Wikipedia[1])

Early research in drone interaction is sparsely reported, probably because the technology was pioneered by the military, leaving most results undisclosed and classified. In 2003, Mouloua et al. [2003] provided an overview of human factors design issues in, for instance, automation of flight control, data-link delays, cognitive workload limitations, display design, and target detection. A recent review by Cahillane et al. [2012] addresses issues of multimodal interaction, adaptive automation, and individual performance differences – the latter examined in terms of spatial abilities, sense of direction, and computer game experience. Interaction with drones is studied as a case of human-robot interaction (e.g., by Goodrich and Schultz [2007]) motivating applied research in computer vision and robotics (e.g., by Oh et al. [2011]).

Yu et al. [2012] demonstrated how a person using a wheelchair could control a drone by EEG signals and eye blinking with off-the-shelf components. They describe a "FlyingBuddy" system intended to augment human mobility and perceptibility for people with motor disabilities who cannot otherwise see nearby objects.

The output from the drone to the pilot can take several forms, from a simple third-person view of the drone (i.e., direct sight) and a 2D-map view to more complex video streams from the drone camera to a monitor or handheld display (see Valimont and Chappell [2011] for an introduction to ego- and exocentric frames of reference in drone control). Novice pilots prefer flying straight while standing behind the drone or flying based on the live video stream from the nose camera. Only skilled drone pilots can make maneuvers by direct sight because this requires a difficult change in frame of reference.

To sum up, the design of good interfaces for drone control is a challenge. Multiple degrees of freedom and several input options call for research to systematically combine and compare different possibilities.

4 Motion and views controlled by gaze

Controlling the movement of a game avatar on the screen can be seen as somewhat analogous to controlling a drone in real life. Avatar gaze-control in World of Warcraft has been researched by Istance et al. [2009]. In a study by Nacke et al. [2010], gamers gave positive feedback regarding immersion and spatial presence when using gaze. Nielsen et al. [2012] also reported higher levels of engagement and entertainment when flying by gaze in a video game compared to mouse. They argue that gaze steering provides kinesthetic pleasure because it is both difficult to master and presents a unique direct mapping between fixation and locomotion.

In a paper on gaze-controlled zoom, Bates and Istance [2002] introduced the concept of "flying" toward an object on the screen. One argument for doing this was the intention to provide users with place experience in remote locations. Navigation in 3D requires several functionalities for controlling both vertical and horizontal panning as well as forward and backward zooming. A multimodal approach is often implemented, as in the experiment presented later in this paper. In work by Mollenbach et al. [2008], pan (similar to yaw rotations of a drone) was controlled by the eyes, and forward and backward zoom (similar to speed) was controlled by the keyboard. In Stellmach and Dachselt´s [2012] study, panning was again controlled by the eyes with three different zoom approaches tested: (1) a mouse scroll wheel, (2) tilting a handheld device, and (3) touch gestures on a smartphone.

A few other studies have looked into the possibility of controlling remote cameras with gaze. Zhu et al. [2010] compared head and gaze motion for remote camera control and found advantages of gaze. Tall et al. [2009] steered a robot by direct gaze interaction with the live video images transmitted from it. Latif et al. [2009] studied gaze operation of a mobile robot applying an extra pedal to control acceleration. Subjects slightly preferred the combination of gaze and pedal compared to on-screen dwell activation of speed. Noonan et al. [2008] examined how fixations of a surgeon can be used to command a robotic probe onto desired locations.

Alapetite et al. [2012] demonstrated the possibility of gaze-based drone control. They expected this to be a very simple task, since the goal was just to fly straight ahead indoors with no wind or obstacles. But it turned out to be rather difficult. Only four of twelve subjects could get the drone through the target without crashing. They observed at least three complicating factors: Some subjects could not resist looking directly at the drone instead of looking at the video image. Some subjects became victims of a variant of the "Midas touch" problem that is common in gaze interaction: Everything you look at will get activated. In this case, the drone would be sent in the wrong direction if the subject just looked away from the target. Since most of the participants were novices, they didn't know this. Subjects also complained that there was a noticeable lag between their gaze shifts and the movement of the drone, which possibly also complicated control. Most importantly, the mapping of the controls may not have been optimal – something that we investigate in the current experiment.

To sum up, several previous studies of gaze controlled zooming, gaming, and locomotion have pointed to the advantages of using gaze for navigating in virtual and real environments. However, the mixed experiences gained by Alapetite et al. [2012] call for additional study on how to pair gaze movements with drone motion to make interaction reliable and intuitive.

5 Experiment

There are yet no well-established experimental paradigms for evaluating drone control. The well-known Fitts' law target acquisition test, commonly used to evaluate input devices, may serve as an inspiration for drone interaction research. Flying through an open or partly opened door is a good example of a real-life target acquisition task performed when flying indoors. The breadth of the passage relative to the size of the drone obviously affects the difficulty and the distance to the target plus the initial orientation with regard to the target will influence task time.

5.1 Participants

Ten male students from IT University of Copenhagen volunteered to participate (mean age 27.7 years, SD = 5.4). They were selected on the requirement that they were regular video game players (mean weekly playing 7.2 hours, SD = 9.0). All participants had normal vision; two used contact lenses but none wore glasses. Two of the participants had tried gaze interaction before and three had tried flying a remote mini-helicopter. None had experience flying a drone.

5.2 Task

We created a simple task that included a change in altitude, rotation, and target acquisition, materialized by a pair of vertical pylons (orange inflatable AR.Race Pylons, height = 250 cm). The pylons where placed 200 cm apart, on tables 120 cm above the ground, thus requiring the drone to gain altitude before passing. As a starting point, the drone itself was placed 3 meters in front of the pylon on the right-hand side. Participants were instructed to pass the pylon by the right, make a U-turn, then pass through the target. This should be done as quickly and safely as possibly (Figure 3).
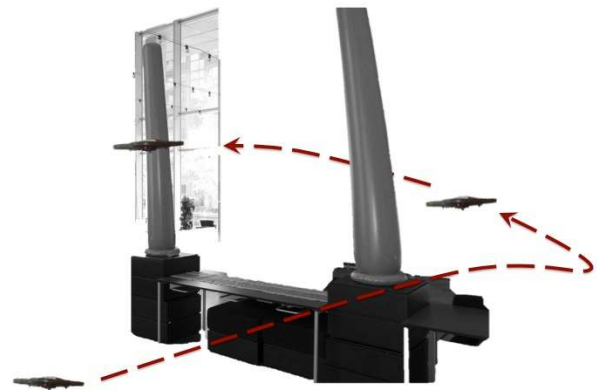

Figure 3 *The experiment task.*

Unlike previous work by Tall et al. [2009] on gaze control of a robot driving on the floor – for which there is a quite obvious mapping between the *x-y* gaze input and movements of the robot on the 2D plan – addressing the problem in 3D is more complicated. The goal is to control the drone in 3D, making the best possible use of the *x-y* gaze input. However, there are several candidate mappings. Even though the drone used here has simple controls compared to a helicopter, there are four degrees of freedom to consider:

• Speed: longitudinal motion (forward/backward translation in the horizontal plane), controlled by a variation of pitch

• Rotation: turning around the *z*-axis (i.e., turning left/right on itself), controlled by a variation of yaw

• Translation: lateral displacement (left/right translation in the horizontal plane), which is controlled by a variation of the roll

• Altitude control: up/down translation

The question then is which of the above degrees of freedom should be assigned to the *x*-axis of gaze input and which should be assigned to the *y*-axis. Intuitively, some combinations that do not make sense can be ruled out, and we are left with two options for *x*-axis gaze control (lateral movement, rotation) and two for *y*-axis gaze control (longitudinal motion, altitude control). The two options for the *x*-axis combined with the two options for the *y*-axis are the four conditions examined in this research. Figure 4 shows the four combinations and the split on key and gaze control. The control models proposed are best suited to drones with an automatic control of attitude (i.e., position in the air) and would not work well for airplanes with a different set of dynamics.
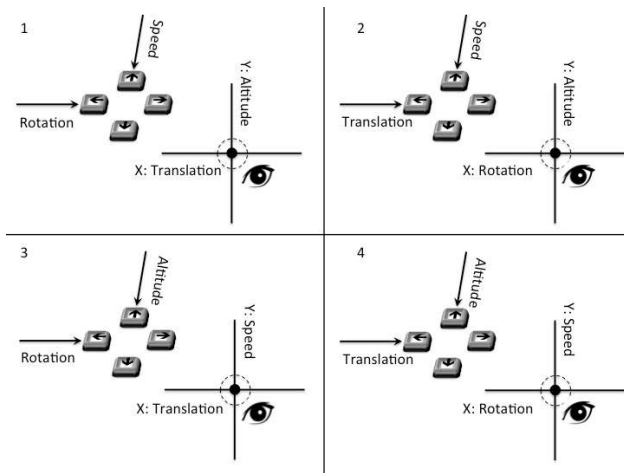

Figure 4 *The four experimental control modes applied in the experiment. Arrow-keys were operated by left hand while the pointer (x,y) was controlled by gaze.*

5.3 Apparatus

Gaze tracking was handled by a system from The Eye Tribe company. The binocular tracker had a sampling rate of 30 Hz, an accuracy of 0.5 degree (average), and a spatial resolution of 0.1 degree (RMS). The tracker (size W/H/D = 20 × 1.9 × 1.9 cm) was placed behind the keyboard and below the screen of a laptop (HP EliteBook 8470w, 15″ screen, Windows 7), and connected by USB.

The drone was quad-rotor A.R. Parrot 2.0 (Figure 1). This model has four in-runner motors and weighs 420 g (with indoor hull). Live video was streamed to the laptop via Wi-Fi from a HD nose camera (1280 × 720 pixels, 30 fps) with a wide-angle lens of 92 degrees (diagonal). During flying, this live stream video was shown full-screen on the laptop.

The application controlling the drone was a modified version of an open-source software by Ruslan Balanukhin (https://github.com/Ruslan-B/AR.Drone). We included a modification to implement our four modes of gaze-control. The modified version is available from (https://github.com/Alkarex/AR.Drone). Control coefficients for speed, altitude, rotation, and drafting where manually tuned according to our assessment of ease of control and responsiveness. The result was an average turn-rate of 22.2 °/s. and an average climbing speed of 0.24 m/s. Default settings of the Parrot 2.0 for maximum speed and altitude (3 m) were used. Figure 5 shows how the control logic was mapped onto the live video image on the laptop. Note that this illustration of controls was not shown on the live video image during experiments. Actually, a unique virtue of this interface is that it is transparent with regard to the changes in viewpoint and the motions it triggers.
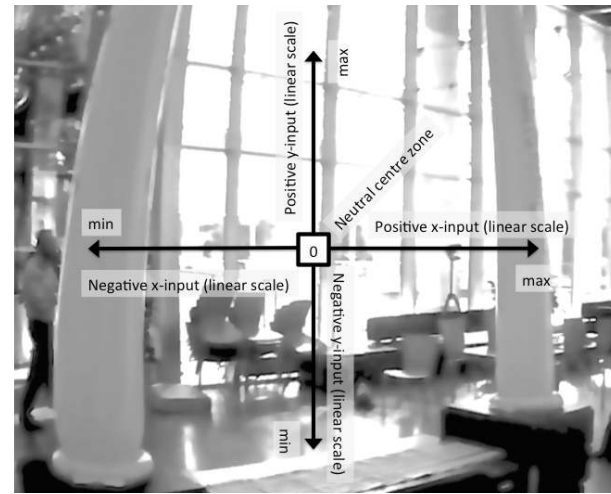

Figure 5 *Visualization of the gaze control on top of a live scene image (not shown during experiments).*

Figure 5 depicts how the point of *x-y* regard on the live image stream related to the movement of the drone. The neutral center box is 2% of the screen height and 2% of the screen width. For roll control, a gain of 0.3 was applied with input limited to 10% of the drone's maximum. For speed control, a gain of 0.3 was applied with input limited to 10% of the drone's maximum. For rotation control, a gain of 0.5 was applied with input limited to 20% of the drone's maximum. For altitude control, a gain of 0.5 was applied with input limited to 50% of the drone's maximum.

In the case of temporary data loss (e.g., during head movements outside the tracking box), the drone would continue according to the last input command received. In a number of gaze-interaction setups, there is a need to smooth the input data at the application level to reduce noise in the signal. In our case however, we did not experience a need for this, because the drone is a physical object with inertia and latency, which inherently filter out the effect of rapid contradictory inputs (e.g., from a noisy signal) and/or brief involuntary inputs (such as eye saccades), as well as brief interruptions in the signal (e.g., due to blinking).

5.5 Procedure

Participants were first given a general explanation of the task while standing in front of the pylons. Then, they were seated 5 meters away from the track in front of the laptop with their back toward the drone. They were unable to see the drone by direct sight. They were then given a detailed explanation about the current control mode and asked to do a test run using the mouse in lieu of the *x-y* gaze mapping. The drone took off automatically and elevated to about 30 cm. Then the participant was given full control of the steering by a key-press from the experimenters who said "Go!" Once the target had crossed at the end of the trial, the experimenter pressed another button for an auto-controlled landing. Task time and keystrokes were logged by the software from the onset of user control to the activation of a landing.

After the test flight with the mouse, the participant underwent a short interview, answering three questions about the user experience: "How difficult was it to control the drone on a scale from 1 to 10 where 1 is very difficult and 10 is very easy?" Similar questions were asked for how reliable and how fun it was to control the drone in this mode. Then the subject was asked about their general impression while the experimenter took notes.

Now began the gaze-controlled trial. The subject was again to fly using the same control mode as just used with the mouse. With the help of the experimenter, there was a 9-point calibration with the gaze tracking system. A manual check was made by pointing at various locations on the monitor to ensure the accuracy was acceptable. Then the drone was launched and the task conducted with gaze control. This was the same task as just performed with the mouse, except gaze was used for the *x-y* mapping in that control mode. Each trial was finished with the same interview as for the mouse trial. All 10 participants tested the four modes in one session in a similar manner. The within-session order was balanced between subjects. It took approximately 45 minutes to complete a full session.

5.6 Design

Although the setup, explanation, calibration, and interview were quite involved, the experiment design was simple. There was one independent variable with four levels:

The four control modes were:

>   M1 – Translation and altitude by gaze; rotation and speed by keyboard

>   M2 –Rotation and altitude by gaze; translation and speed by keyboard

>   M3 – Translation and speed by gaze; rotation and altitude by keyboard

>   M4 – Rotation and speed by gaze; translation and altitude by keyboard.

Each level of control mode had a unique combination of *x-y* gaze control and arrow-key control for the four degrees of freedom noted earlier. Figure 4 shows the particular mappings for each control mode. The order of presenting the control modes was counterbalanced, as noted in the previous section. The dependent variables were task completion time, key presses, and subjective ratings for easiness, reliability, and fun. We also computed an aggregate subjective measure for "user experience", based on the

responses for ease-of-use, reliability, and fun. In addition, it was manually logged if the drone crashed or if poles were contacted while maneuvering the drone. With 10 subjects trying four control modes, there were a total of 40 trials. Each trial had a mouse practice trial followed by a gaze trial.

6   Results

Data from two trials where the drones crashed and from one trial that timed-out were not included in the analysis of performance. However, these trials were included in the subjective analysis.
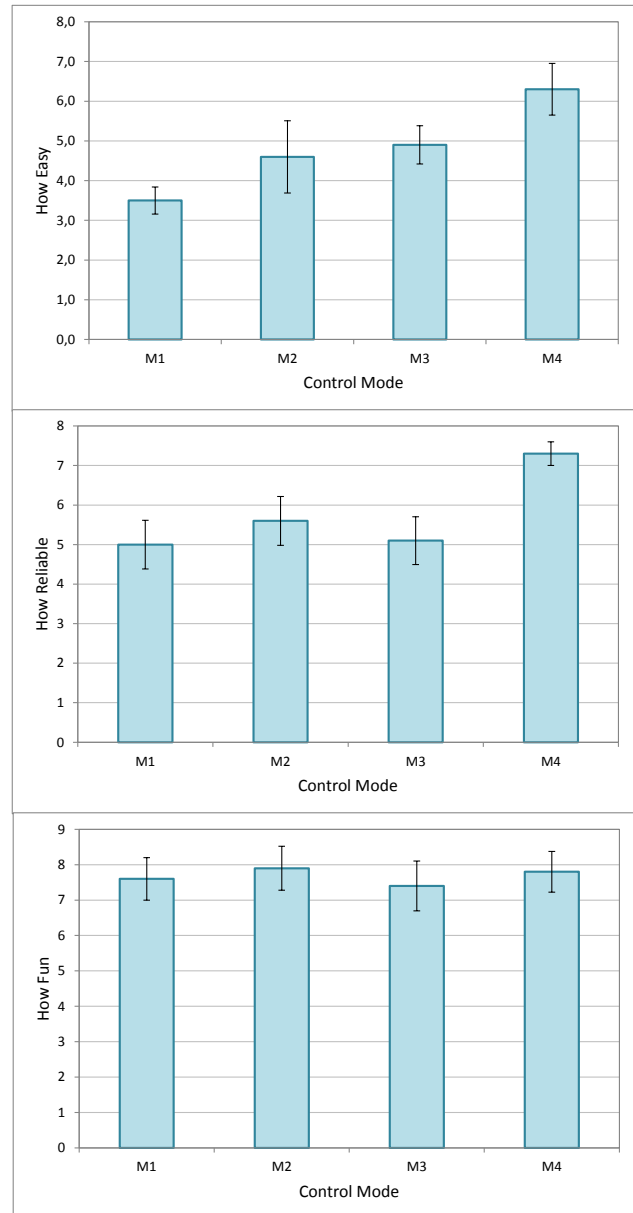

Figure 6 *Mean values for the four control modes to questions on how easy, reliable and fun the modes were perceived (on a 10-point Likert scale). Error bars show ±1 SE.*

The mean task completion time for gaze controlled flying was 44.0 seconds (*SE* = 3.2 s). The effect of control mode on task completion time was not statistically significant ($F_{3,18} = 0.776$,

ns). The mean number of manual keystrokes per trial was 13.5 (SE = 1.7). The effect of control mode on the number of keystrokes was not statistically significant ($F_{3,18}$ = 0.839, ns) nor was the effect of control mode on keystrokes per second ($F_{3,18}$ = 0.132, ns).

All subjective responses were analysed using the Friedman non-parametric test. The responses to the question on "how difficult" (re-calculated to ease-of-use by flipping the scale) were not statistically significant (H_3 = 4.63, p > .05). Figure 6 shows mean values for the four conditions. The responses to the question on reliability were statistically significant (H_3 = 9.54, p < .05). A post-hoc pairwise comparisons test using Conover's F revealed that mode 4 (M4) was deemed more reliable that the other modes (p < .05). The responses to the question on fun were not statistically significant (H_3 = 1.24, p > .05). Finally, a composite of the questionnaire items was computed as the mean of the responses for the three questions. This aggregate we consider a measure of the overall "user experience" (UX) with the control mode under test. The UX scores were not statistically significant (H_3 = 6.79, p > .05).

A Wilcoxon Signed-Rank non-parametric test was used to compare the mouse and gaze questionnaire responses. For "ease-of-use", the differences were statistically significant, favoring the mouse (z = -2.80, p < .01). For "how reliable", the results again were statistically significant favouring the mouse (z = -2.60, p < .01). On "how fun", gaze is rated slightly better than the mouse however, the difference was not statistically significant (z = -0.07, p > .05).

After each trial, participants gave spontaneous comments. Here are a few typical examples:

> *It was hard because you did translation by accident a lot. You were whopping from left to right. It took me some time to get used to that and look to the middle* (Comment, mode 1).

> *It's almost impossible for me to control this. When something is about to go wrong – like flying into the pole – you look at it and then you fly directly into it.* (Comment, mode 2)

> *It was confusing when I looked up. I forgot that I was translating when looking to the side. I looked at objects that I was afraid to collide with – and this made things even worse.* (Comment, mode 3)

> *Much easier to control altitude with eyes – you felt more safe because you would use the fingers to go forward and backwards – the only thing that could wrong with eye control would be going up and down. (Comment, mode 4)*

The subjects also made comments on gaze control in general:

> *Loosing control with gaze is more stressful compared to mouse because you need to look around to regain control but this is difficult when you are controlling something with the gaze at the same time.* (Comment, mode 1)

> *Very difficult because you get to look somewhere and it flies there. Very counter-intuitive that you should keep your eyes on where you would like to go. It was really fun.* (Comment, mode 1)

> *I was not able to look at the keyboard so I was annoyed by that. It was more exhaustive to control with your eyes, you need to look more carefully.* (Comment, mode 4)

Six participants specifically mentioned that they preferred associating altitude control to the *y*-axis of gaze (as opposed to associating the *y*-axis to longitudinal motion). Eight participants specifically mentioned that they preferred associating rotation control to the *x*-axis of gaze (as opposed to associating the *x*-axis to lateral displacement). Furthermore, four participants specifically mentioned that they preferred having the two most important types of controls for this type of mission, i.e., rotation control and longitudinal motion, split on two distinct input modalities, i.e., one on the keyboard and the other by gaze.

7   Discussion of results

The main observation in this experiment was that gamers could actually control the drone by gaze, independently of control mode, and with just a small amount of practice. This is indicated by the low error rate, namely 2 crashes out of 40 trials. One control mode (M4 – Rotation and speed by gaze; translation and altitude by keyboard) was judged significantly more reliable than the others and this mode was also rated slightly easier, although insignificantly. The mode had no impact on task time, but we expect that a more detailed study, including more subjects and several trials (e.g., with different levels of difficulty) might show differences in time and error rates.

We believe there are two reasons why M4 was considered the most reliable. It offers a natural mapping from gaze movements to rotations, similar to what would be the consequences of a lateral turning of the eye. It is also the control mode most similar to the one gamers use in 3D games, where the mouse is commonly used to turn the viewpoint.

In the demo by Alapetite et al. [2012], most participants crashed. There are perhaps several reasons for this. First, participants used one of the most difficult control conditions (similar to M2 in our experiment). Participants received no training with a mouse before trying. As well, our subjects had substantial experience in 3D-navigation from video games. Alapetite et al. just tested gaze controlled flying with a random selection of visitors at an exhibition. Finally, the previous test used the A.R. Parrot version 1 while we used version 2 in this experiment, which has improved stability.

The subjects in this experiment were all experienced gamers. Some of the comments strongly reflect this. For instance, they tend to regard challenging controls to be extra fun. This is most likely not how the general user population would think. However, gamers could become an interesting first-mover group that would help developing the design of controls. We also regard gamers to be compatible to the professional drone pilot, for instance tele-operators. While it might be difficult to get professional users committed for longitudinal studies, gamers – whom are common among university students – could serve as a good substitute, at least in testing an initial design. The disadvantage of using gamers is that they often have a preferred split of controls and may not be as willing to change this – or would have a more difficult time learning new mappings. In fact, it might be interesting to re-do the experiment with non-gamers to see if the preference for mode M4 still holds. If not, it is perhaps an effect of game skills rather than an effect due to the direct mapping between lateral gaze movements and turns of viewpoint.

Our subjects favored the mouse in terms of ease-of-use and reliability – while this did not spoil the fun of using gaze. Some subjects commented on a temporary loss of gaze tracking which would have an immediate effect on a drone in motion. Some complained that a slight offset made it difficult for them to rest in the neutral center zone. Several noticed the need to put gaze on hold while orienting themselves.

The Midas touch problem is important to address when designing gaze interfaces to drones. The fact that mode 4 was considered easiest strongly suggests that this mode's keyboard control of forward/backward movements is most welcome, since this effectively serves as a engage-disengage clutch, enabling the pilot to hover in a stationary position while orienting himself prior to flying off in the desired direction. In the next section we suggest designs to encompass this.

8   General discussion

In our future research we intend to study gaze-based drone interaction while wearing a near-eye display, since this is already a display form preferred by drone professionals. This approach is intuitively appealing, as visual input is obtained by moving the eyes. If successfully interpreted, control becomes implicit, meaning we do not increase the task load by using complicated control mappings. Inspired by the work of Duchowski et al. [2011], we suggest investigating the efficacy of binocular eye data to measure convergence as a means of automatically selecting among multiple interfaces at various distances. Exploring the viability of working with layered drone control interfaces, within several concurrent user contexts, from a perceptual point of view, and with a variety of eye, head, face, and muscle input combinations, then becomes mandatory. The present study did not consider saccades, micro-saccades or eye blinking for drone control. However, it would be relevant for future research to explore how these physiological parameters could potentially add to the orchestra of eye input to the drone via a near-eye display.

Some previous studies have used an extra input to control speed while maneuvering with gaze. For instance Latif et al. [2009] and Stellmach and Dachselt [2013] examined the use of a pedal for this purpose. Gyroscopes and accelerometers are now extensively used in handheld devices and in game consoles. Tilting a tablet or smartphone could become an appropriate mobile control of lateral displacement and longitudinal motion [Stellmach and Dachselt 2012], because these properties oscillate around a zero value. Conversely, rotating the drone by rotating the telephone, or controlling the altitude by moving the telephone up and down would only be doable for modest amplitudes of control. Therefore, in the case of tablets/smartphones, it makes sense to assign rotation as well as altitude to gaze, and lateral displacement as well as longitudinal motion to the device's tilt sensor (i.e., similar to our control mode M2).

Applying motion sensor technology to a near-eye display will enable the system to be sensitive to head movements. Speed could then be controlled by slight forward or backward head tilts. Facial movements are easy to control for most people; such can be monitored by sensors embedded in the glasses [Rantanen et al. 2012]. Finally, yet another camera may be placed on the front frame of the glasses. This can record all hand gestures as input for the system. We suggest exploring the possibility of using gaze in conjunction with hand gestures to enable richer interaction, and to afford an effective filter for accidental commands: Only when the user looks at her hands should the gesture be interpreted as input.

What would people use a personal drone for – except for fulfilling an ever-fascinating dream of flying free and safe? Our belief is that personal drones are, for instance, an opportunity for open-community data collection of the visual environments (indoors and outdoors). Video recordings may be tagged with the steering commands and fixation points that the human pilot produced when generating them. This constitutes a complementary new set of information to the recorded environment that no previous research has yet explored and with a potential breakthrough in providing vision robots high-level human perceptual intelligence and behavioral knowledge.

9   Conclusion

People with game experience are able to control a drone by gaze without much training. Mapping of controls are important to the user experience. If the research succeeds, it may constitute a change in human-machine interactions as well as bringing into the world an intriguing hardware/software solution that will allow people to virtually fly around, using only their bodies and line of sight as a guide.

References

ALAPETITE, A., HANSEN, J. P., & MACKENZIE, I. S. 2012. Demo of gaze controlled flying. In *Proceedings of the 7th Nordic Conference on Human-Computer Interaction: Making Sense Through Design*. New York: ACM. 773–774.

BATES, R., & ISTANCE, H. 2002. Zooming interfaces!: enhancing the performance of eye controlled pointing devices. In *Proceedings of the fifth international ACM conference on Assistive technologies*. New York: ACM. 119–126.

CAHILLANE, M., BABER, C., & MORIN, C. 2012. Human Factors in UAV. *Sense and Avoid in UAS: Research and Applications*, *61*, 119.

DUCHOWSKI, A. T., PELFREY, B., HOUSE, D. H., & WANG, R. 2011. Measuring gaze depth with an eye tracker during stereoscopic display. In *Proceedings of the ACM SIGGRAPH Symposium on Applied Perception in Graphics and Visualization* New York: ACM. 15–22.

GOODRICH, M. A., & SCHULTZ, A. C. 2007. Human-robot interaction: a survey. *Foundations and Trends in Human-Computer Interaction*, *1*(3), 203–275.#

HANSEN, J. P., AGUSTIN, J. S., & SKOVSGAARD, H. 2011. Gaze interaction from bed. In *Proceedings of the 1st Conference on Novel Gaze-Controlled Applications*. New York: ACM. 11:1–11:4.

ISOKOSKI, P., JOOS, M., SPAKOV, O., & MARTIN, B. 2009. Gaze controlled games. *Univers. Access Inf. Soc.*, *8*(4), 323–337.

ISTANCE, H., VICKERS, S., & HYRSKYKARI, A. 2009. Gaze-based interaction with massively multiplayer on-line games. In *Proceedings of the 27th international conference extended abstracts on Human factors in computing systems.* New York: ACM. 4381–4386.

LATIF, H. O., SHERKAT, N., & LOTFI, A. 2009. Teleoperation through eye gaze (TeleGaze): a multimodal approach. In *Robotics and Biomimetics (ROBIO), 2009 IEEE International Conference on.* New York: IEEE. 711–716.

MOLLENBACH, E., STEFANSSON, T., & HANSEN, J. P. 2008. All eyes on the monitor: gaze based interaction in zoomable, multi-scaled information-spaces. In *Proceedings of the 13th international conference on Intelligent user interfaces.* New York: ACM. 373–376.

MOULOUA, M., GILSON, R., & HANCOCK, P. 2003. Human-centered design of unmanned aerial vehicles. *Ergonomics in Design: The Quarterly of Human Factors Applications*, *11*(1), 6–11.

NACKE, L. E., STELLMACH, S., SASSE, D., & LINDLEY, C. A. 2010. Gameplay experience in a gaze interaction game. Retrieved from http://arxiv.org/abs/1004.0259

NIELSEN, A. M., PETERSEN, A. L., & HANSEN, J. P. 2012. Gaming with gaze and losing with a smile. In *Proceedings of the Symposium on Eye Tracking Research and Applications* New York: ACM. 365–368.

NONAMI, K., KENDOUL, F., SUZUKI, S., WANG, W., & NAKAZAWA, D. 2010. *Autonomous Flying Robots: Unmanned Aerial Vehicles and Micro Aerial Vehicles.* Springer Publishing Company, Incorporated.

NOONAN, D. P., MYLONAS, G. P., DARZI, A., & YANG, G.-Z. 2008. Gaze contingent articulated robot control for robot assisted minimally invasive surgery. In *Intelligent Robots and Systems, 2008. IROS 2008. New York: IEEE.* 1186–1191.

OH, H., WON, D. Y., HUH, S. S., SHIM, D. H., TAHK, M. J., & TSOURDOS, A. 2011. Indoor UAV Control Using Multi-Camera Visual Feedback. *Journal of Intelligent & Robotic Systems*, *61*(1), 57–84.

PFEIL, K., KOH, S. L., & LAVIOLA, J. 2013. Exploring 3d gesture metaphors for interaction with unmanned aerial vehicles. In *Proceedings of the 2013 international conference on Intelligent user interfaces.* New York: ACM. 257–266.

RANTANEN, V., VERHO, J., LEKKALA, J., TUISKU, O., SURAKKA, V., & VANHALA, T. 2012. The effect of clicking by smiling on the accuracy of head-mounted gaze tracking. In *Proceedings of the Symposium on Eye Tracking Research and Applications. New York: ACM.* 345–348.

STELLMACH, S., & DACHSELT, R. 2012. Investigating gaze-supported multimodal pan and zoom. In *Proceedings of the Symposium on Eye Tracking Research and Applications.* New York: ACM. 357–360.

STELLMACH, S., & DACHSELT, R. 2013. Still looking: investigating seamless gaze-supported selection, positioning, and manipulation of distant targets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.* New York: ACM. 285–294.

TANRIVERDI, V., & JACOB, R. J. 2000. Interacting with eye movements in virtual environments. In *Proceedings of the SIGCHI conference on Human factors in computing systems.* New York: ACM. 265–272.

VALIMONT, R. B., & CHAPPELL, S. L. 2011. Look where I'm going and go where I'm looking: Camera-up map for unmanned aerial vehicles. In *Proceedings of the 6th international conference on Human-robot interaction (HRI '11).* New York: ACM. 275–276.

YU, Y., HE, D., HUA, W., LI, S., QI, Y., WANG, Y., & PAN, G. 2012. FlyingBuddy2: a brain-controlled assistant for the handicapped. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing.* New York: ACM. 669–670.

ZHU, D., GEDEON, T., & TAYLOR, K. 2010. Head or gaze?: controlling remote camera for hands-busy tasks in teleoperation: a comparison. In *Proceedings of the 22nd Conference of the Computer-Human Interaction Special Interest Group of Australia on Computer-Human Interaction.* New York: ACM. 300–303.