Journal of Eye Movement Research
7(4):4, 1-14

Human-Robot Interaction Based on Gaze Gestures for the Drone Teleoperation

Mingxin Yu
School of Automation, Beijing Institute of Technology, Beijing, China
Intelligent Human-Machine Systems Lab, Northeastern University, Boston, USA

Yingzi Lin
Intelligent Human-Machine Systems Lab, Northeastern University, USA

Xiangzhou Wang
School of Automation, Beijing Institute of Technology, China

David Schmidt
Intelligent Human-Machine Systems Lab, Northeastern University, USA

Yu Wang
School of Automation, Beijing Institute of Technology, China

Teleoperation has been widely used to perform tasks in dangerous and unreachable environments by replacing humans with controlled agents. The idea of human-robot interaction (HRI) is very important in teleoperation. Conventional HRI input devices include keyboard, mouse and joystick, etc. However, they are not suitable for handicapped users or people with disabilities. These devices also increase the mental workload of normal users due to simultaneous operation of multiple HRI input devices by hand. Hence, HRI based on gaze tracking with an eye tracker is presented in this study. The selection of objects is of great importance and occurs at a high frequency during HRI control. This paper introduces gaze gestures as an object selection strategy into HRI for drone teleoperation. In order to test and validate the performance of gaze gestures selection strategy, we evaluate objective and subjective measurements, respectively. Drone control performance, including mean task completion time and mean error rate, are the objective measurements. The subjective measurement is the analysis of participant perception. The results showed gaze gestures selection strategy has a great potential as an additional HRI for use in agent teleoperation.

Keywords: Human-Robot Interaction, Teleoperation, Gaze gestures, Object Selection, Gaze-controlled Interfaces

Introduction

Teleoperation is a kind of system which controls agents, e.g. robotic vehicles, Drones, etc, from a remote location through wireless signals such as Wi-Fi, GPS, cellular phone signals, etc (Fong & Thorpe, 2001). The system has advantages of being able to replace humans working in dangerous and unreachable environments to reduce mission failure and avoid casualties. In order for agents to perform well, human-robot interactions (HRIs) need to be designed and provided for users as convenient and efficient as possible. Most HRIs for teleoperation are in the form of traditional systems based on hand-controllers, e.g. joystick, keyboard, mouse, touchpad, etc. In the process of teleoperation, generally, the users are required to sit in front of a computer and view real-time images displayed on the computer screen. The image is a live video stream from remote cameras mounted on the agent or a fixed position on the ground. Then, the user sends control commands manually, e.g. by pressing keys, touching the screen, or clicking the mouse, to the agent towards completion of specific tasks. Manual operation is completed by traditional interaction devices, i.e. hand-controllers.

However, there are two problems in traditional HRIs for agent teleoperation. On one hand, some handicapped users, especially those with disable upper limbs, would not be able to control agents using traditional interaction devices. On the other hand, users often control agents

DOI 10.16910/jemr.7.4.4

ISSN 1995-8692

using more than two types of HRI input devices simultaneously for task completion, e.g. controlling agent movement by keyboard, adjusting camera viewing directions by joystick, etc. In this case, users are required to switch hands and attention between interaction devices and interfaces resulted in reduced task efficiency, increasing mental workload and even physical fatigue (Zhu, Gedeon & Thorpe, 2011). For those reasons researchers considered a novel HRI based on eye gaze with an eye tracker as an additional input modality for users. The gaze-based HRI has two features used in agent teleoperation. First, during the process of teleoperation, eye gaze moves toward targets and areas of interest on real-time images displayed on screen, providing valuable information about user intent, which is an "attentive input" (Zhai, 2003). Second, gaze speed is much faster than hand speed with traditional interaction devices when the users look at objects on the screen. Hence, agent teleoperation performance can benefit a lot from the valuable and fast information provided by gaze. In this paper, we use the gaze-based HRI as an input modality to teleoperate a drone, a so called unmanned aerial vehicle (UAV), as an example of agent teleoperation.

The rest of this paper is organized as follows. Some related works about agent teleoperation based on gaze are given in the *Related Works* section. The *Motivation* section gives the motivation of our research. The gaze gestures theory is detailed in the *Gaze Gestures* section. The experimental setup is given in *Experimental Setup* section. The command design of the drone controls based on gaze gestures are detailed in *Commands Design for the Drone Controls* section. The *Experiment* and *Experimental Results* sections present the experiment and evaluation results, respectively. The *Discussion* section gives the benefits and limitations of gaze gestures. The conclusion is summarized in the *Conclusion* section.

Related Works

In general, the teleoperation process of users can be divided into two classes: navigation tasks and object selection tasks, i.e. command selection tasks. Navigation tasks are the eyes moving to see objects on the screen, e.g. cursor movements. The object selection tasks are to trigger actions to perform, e.g. clicking the mouse.

Although eye movements can indicate intent to interact with an object, the eyes lack an activation mechanism. In other words, a user can look at an object on the screen, but it is not clear to the interaction system whether it should issue a control command or not, i.e. explicit control intent is not inherent from gaze information. In our research, the gaze as sole input needs to be able to handle both navigation and object selection tasks. There are many object selection strategies for gaze-based human-computer interactions. Huckauf and Urbina presented a good summary and generalization (Huckauf & Urbina, 2008). However, our attention focuses on human-robot interaction based on gaze input modality for agent teleoperation. Hence, through the selection of previous works, current object selection strategies for gaze input mainly include dwell times activation, combined with other HRIs based on manual modalities (keyboards, joystick, etc), and smooth pursuit in the HRI field. Especially, the smooth pursuit selection strategy principle is similar to dwell times. The following works are thought to be more related to our work.

A robotic arm was controlled using an eye gaze tracking system for handicapped users (Yoo et al., 2002). The operational interface on the screen was divided into a feedback region displaying real-time images from a camera mounted at a fixed location and a commanding region with a number of command buttons. The users observed feedback images and gazed towards the buttons, each corresponding to a robot joint which he/she decides to control. Although the paper did not give the object selection strategy clearly, in general, the on-screen buttons are often activated with dwell times selection strategy.

An eye gaze tracker was developed for gaze-based interaction modality used in mobile robot control (Yu et al., 2014). The interface on the screen was divided into a lot of grid in the horizontal and vertical directions. The feedback image from a camera fixed at a location in the movement field of robot was shown on the whole screen. The users were required to focus their gaze at one of many grid sections for a duration to activate the robotic vehicle actions, e.g. forward, backward, etc. This method has more real states on the screen than the method in (Yoo et al., 2002), but more grid sections make effects screen viewing when users operate the mobile robot. However, the two systems mainly focus on the design work for eye trackers, and do not give a detailed analysis of results on object selection strategies.

TeleGaze system used an eye gaze tracker to teleoperate a mobile robot (Latif, Sherkat & Lotfi, 2008). A

DOI 10.16910/jemr.7.4.4

ISSN 1995-8692

pan/tilt camera mounted on the mobile robot provided real-time images as feedback displayed on the screen. The transparent graphic regions named as "active zones" were overlaid on the feedback image enabling users to issue control commands as long as users fixated on active zones. Control of the robot consisted of two parts: robot movement and pan/tilt camera controls. The object selection strategy was attributed to dwell times activation. Because of the inherent disadvantage of dwell times selection strategy, Midas touch, a multimodal approach, was further researched and presented for teleoperation of mobile robots (Latif, Sherkat & Lotfi, 2009). This approach also uses "active zones" on the screen, but the activated tasks were completed through an extra pedal instead of gaze fixations with dwell times. The results showed, compared to on-screen dwell times activation, subjects slightly preferred the combination of gaze and pedal.

In (Tall et al., 2009), smooth pursuit was used to guide a robotic vehicle. The direction was determined by looking where the users would like to drive. The interface provided a direct feedback loop with no visible graphic regions delegated for movement commands, e.g. forward, backward, etc. The direction and speed are controlled linearly by the distance between gaze points and center point of the monitor. The system also adopted dwell times selection strategy when turning and stopping the robot. Those designed on-screen buttons were put on the edge of the interface.

Controlling Drone by gaze-based HRI was proposed in (Alapetite, Hansen & Mackenzie, 2012). Like the design in (Tall et al., 2009), the interface provided no visible graphic regions as on-screen buttons for control commands but relies on feedback loops. The object selection strategy was attributed to the smooth pursuit method. However, Drone control requires four degrees of freedom, i.e. speed, rotation, translation, altitude (Detailed in *Commands Design for the Drone Controls* section). The gaze tracker provided an input in two dimensions, i.e. x and y directions. The control strategy is required design mapping from 2D to 3D-world of the Drone. In other words, the object selection strategy can cover two degrees of freedom corresponding to control commands. An additional study was given in (Hansen et al., 2014), where the keyboard worked as an additional HRI input device to compensate for the uncovered degrees of freedom corresponding to control commands. The experiment aims to find out how to pair gaze movements with drone motions to make interaction reliable using eye tracker and keyboard interfaces.

Motivation

As for agent teleoperation, the correctness and timeliness of control commands are very important during operation, since wrong commands and command delay can cause crashes and even lose agents, which are dangerous and costly. However, dwell times and smooth pursuit selection strategies bear a considerable amount of disadvantages that might result in those cases. The main problem is the Midas touch problem (Jakob, 1991), resembling King Midas, who turned everything to gold by touch, in dwell times and smooth pursuit selection strategies. Especially with the smooth pursuit selection, some subjects reflected everything you looked at would get activated (Hansen et al., 2014). In this case, the agent would be sent the wrong commands, something novices may not be able to recognize and correct. Although the dwell times selection strategy can solve this problem through setting an appropriate duration time, it is crucial to achieve optimal performance. Too short dwell times will increase the amount of unintended selections, whereas too long dwell times will make increase command delays to agents.

The visible graphic regions are usually drawn on the screen as "active zones" for producing agent control commands in dwell times selection strategy. In general, agent teleoperation needs a lot of control commands, e.g. camera and robot-self controls. However, more control commands will increase amount of graphic regions that take up screen space. At the same time, briefs are often written on the regions to indicate to the users what the graphic regions delegate (Latif et al, 2008, 2009). In this case, the operational interface would affect the users' field of the vision and even cause confusion. Contrarily, there are fewer visible graphic regions provided in the smooth pursuit selection strategy, which uses the gaze points to directly control. The strategy gives better viewing space, but at the cost of control command variety (Alapetite et al., 2012). The solution is to combine other additional HRI input devices into one system, e.g. eye tracker with keyboard (Hansen et al., 2014).

During agent teleoperation, the command transmissions need to be kept continuous. In the dwell times se-

DOI 10.16910/jemr.7.4.4

ISSN 1995-8692

lection strategy, users looked at active zones for a dwell time, and a command was continuously activated until the gaze moved to another region (Yu et al., 2014; Latif et al., 2008). In the smooth pursuit selection strategy, users keep their gaze towards the direction they would like to drive (Tall et al., 2009; Alapetite et al., 2012; Hansen et al., 2014). However, in the two cases, increased of operational time enhances mental workload. Although combining gaze control with manual controllers can provide the possibility of fast and efficient controls, we accept a fact that this case reduces some of the advantages of gaze control, e.g. having the hands free, due to affording an additional input device (Latif et al., 2009; Hansen et al., 2014).

To sum up, there are several of disadvantages of selection strategies from dwell times and smooth pursuit in gaze-based HRI field. Consequently, this paper tries to introduce gaze gestures as an object selection strategy to improve those problems.

Gaze Gestures

The concept of gaze gestures has been recently proposed in the gaze research field (Drewes et al., 2007). Gaze gestures are based on saccadic movements and defined as sequences of strokes, which are the foundation of gaze gestures. A stroke is defined as the motion between two intended fixations. It is different from eye saccades, which can be defined as the eye movement between gaze fixation points.

There are several types of gaze gestures based on different principles to complete object selection. Urbina and Huckauf proposed a gaze gesture based on boundary crossing called "EyePie writing" (Urbina & Huckauf, 2007). Istance et al. developed a moded approach, which is so-called "Snap Clutch", to solve the Midas touch problem. They recognized gaze gestures are based on gaze strokes crossing the side of the monitor for changing modes of the gaze input (Istance et al., 2008). The gaze gestures based on changes in saccade direction were presented by (Drewes et al., 2007). A gesture was defined based on the order and relative direction of saccades. Heikkilä and Räihä proposed gaze gestures based on shape tracking and presented different types of gestures for eye-drawing (Heikkilä & Räihä, 2009). In our paper, we used gaze gestures based on active zones, also so called fix-points.

Gaze gestures based on active zones consist of single stroke gaze gestures (SSGG) and complex gaze gestures (Mollenbach, 2010). A single stroke gaze gesture is defined as the motion between two intended fixation points. The complex gaze gestures are the movement between more than two intended fixation points. Figure 1 shows the examples of a SSGG and complex gaze gestures, respectively.

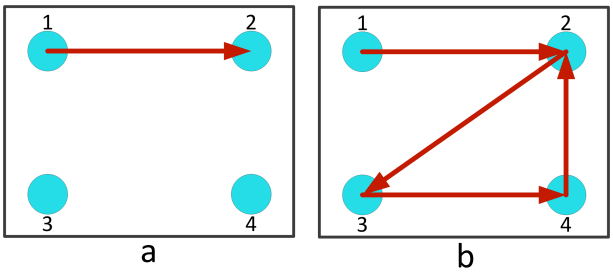

Figure 1: A single stroke gaze gesture and complex gaze gestures.

Intuitively, a SSGG is easier and consumes little time to complete selection, but a potential problem may happened that overlaps between natural inspection patterns and intended eye movement patterns, known as accidental gesture completion. Complex gaze gestures have an advantage of increasing the interaction 'vocabulary', but this brings some difficulties on cognitive and physiological load for users, since numerous gaze gestures need to be remembered and more than single gaze stroke is needed to complete selection.

In addition, some factors, such as the number of strokes, layout of active zones, and length between active zones, have an effect on the completion of gaze gestures selection. In (Istance et al., 2010), researchers used 2 and 3 strokes as gaze gestures and a line path between active zones to control the player's avatar in the game World of Warcraft. The average stroke completion times were reported as 494ms for 2 strokes and 879ms for 3 strokes. Although the 3 stroke time is longer than average 490ms dwell times (Mollenbach, 2010), this case can more effectively avoid the Midas touch problem. Mollenbach proposed two design criterions regarding the layout of active zones on the screen interface. On one hand, the center of the screen should be unaffected by eye gaze. On the other hand, the initial point of the gaze itself should have no effect on the system. Hence, the active zones are on the periphery of the screen, since fixations rarely occur there and the risk of accidental gesture completion is

DOI 10.16910/jemr.7.4.4

ISSN 1995-8692

largely decreased. Based on this active zone layout on the interface, long and short SSGGs were examined and compared to dwell selection with five increments of fixation duration time. The long and short SSGGs were related to stroke length to complete selection. The results show no significant difference regarding the SSGG and dwell times for solving similar tasks, but did show some differences between short and long SSGG. Consequently, according to different tasks, the researchers need to design the number of strokes, layout of active zones and distance between active zones on the interface in terms of effective strategies for solving selection tasks (Mollenbach et al., 2013).

Compared to dwell times and smooth pursuit object selection tasks, there are some advantages to the gaze gestures selection strategy. First, the Midas touch problem can be solved by gaze gestures because it can distinguish between natural navigational eye movements and selections. Second, gaze gestures do not require much screen space, because the active zones can be drawn with semi-transparent, opaque or hollow circles. At the same time, complex gaze gestures require less active zones to constitute a lot of object selections, compared to dwell time selection, which requires more active zones. Third, gaze gestures can complete object selection quickly, since the elapsed time to cover a 1° to 40° visual angles is 30-120ms (Duchowski, 2007). Finally, gaze gestures can decrease mental workload compared with smooth pursuit object selection tasks, because the method does not require users to keep their gaze within active zones for command continuity but to complete one time for each selection task.

This paper introduces gaze gestures as an object selection strategy for human-robot interaction. Through observation of the literature, gaze gestures as the control object selection method are seldom researched in the agent teleoperation field. We found one paper (Mollenbach, Hansen, Liholm & Gale, 2009) that used single stroke gaze gesture to control wheelchair movement in a real world environment. In our research, we give an example of agent teleoperation based on gaze gestures for drone control. In the following sections, we detail the design of gaze gestures as control commands for drone control.

Experimental Setup

The system consists of a remote and teleoperation station, as seen in Figure 2.

The remote station is an off-the-shelf A. R. Parrot drone 2.0. This model has four in-runner motors and, with the indoor hull, weighs 420g. The drone is a low cost UAV at less than $400. A HD camera with resolution of 1024×720 pixels at 30fps is mounted on the nose of drone in order to provide live video to a laptop for teleoperation via Wi-Fi.

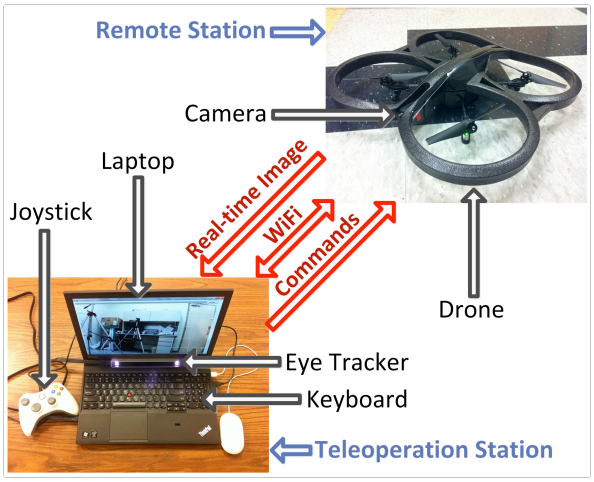

Figure 2: Experimental Setup.

The teleoperation station includes an eye tracker, a laptop, and a joystick. The eye tracker is a low cost gaze tracking system ($99) from The Eye Tribe company. It has two infrared lights and a camera, sampling at 30Hz or 60Hz, and an average accuracy of 0.5 degrees. The size W/H/D of the eye tracker is 20×1.9×1.9 cm. It is small enough to be placed behind the keyboard and below the screen of a laptop (2.4GHz i7-4700MQ CPU and 8G DDR3) and connected via USB 3.0 . The videos are displayed on a 15.6-in LED backlit anti-glare laptop monitor controlled by a graphics workstation (Quadro K2100, NVIDA corp.). The spatial resolution of the screen display used in our research is 2048×1152 pixels. The eye tracker allows a small range of head movement after calibration. The operational distance between subjects and computer screen is about 60cm. The joystick is used as an additional HRI input device provided for comparison with the eye tracker.

DOI 10.16910/jemr.7.4.4

ISSN 1995-8692

Gaze data is calculated and captured using an open source SDK provided by The Eye Tribe. The laptop is used as the interface where interaction takes place between user and drone.

Commands Design for the Drone Controls

In this section, we design a set of commands based on gaze gestures for drone controls.

In Figure 3, the drone controls cover four degrees of freedom (Hansen et al., 2014):

Speed: Longitudinal motion (forward/backward translation in the horizontal plane) is controlled by variation in pitch.

Rotation: Turing about the vertical axis, i.e. turning left/right on itself, is controlled by variation of yaw.

Translation: Lateral displacement (left/right translation in the horizontal plane) is controlled by variation of roll.

Altitude control: Vertical translation.

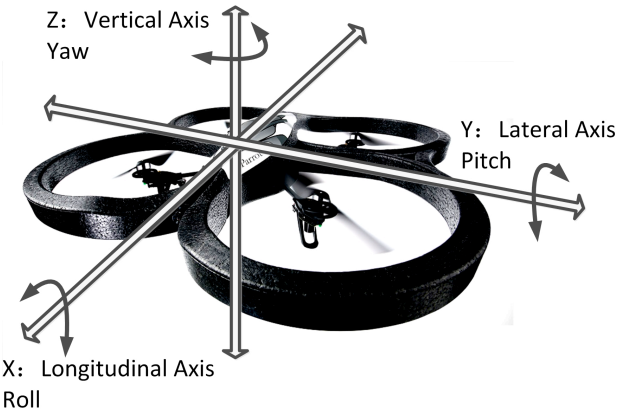

Figure 3: The three control axes: pitch, roll and yaw of the A. R. Parrot 2.0.

In our research, gaze gestures are based on active zones, which include single stroke gaze gestures and complex gaze gestures. The problem is which of the above degrees of freedom should be appropriately assigned to corresponding to gaze gestures. According to the description in the *Gaze Gestures* section, the number, size and position of active zones on the screen interface are required for consideration when designing a set of commands for drone control.

During the process of drone control, we observed that users have a bias tendency to fixate towards the center region of the screen, similar to how drivers are required to keep gaze forward when driving. As for the periphery of the screen, fixations rarely occur there, only when turning the drone (left/right) and changing drone altitude (up/down) would users gaze at the periphery. Here, we divided the drone control degrees of freedom into two groups to design control commands. One contains longitudinal motion and lateral movement, While the other includes rotation and altitude control.

According to the above analysis, we draw eight active zones on the screen interface. The distribution of active zones is such that the four for the first group are located at each corner of the interface, and the four for the second group are put on the neighboring regions corresponding to the four corners. In the following, we give a detailed design of drone control commands.

First group: The variation in pitch controls forward and backward translation of the drone. Because user fixations often fall into the center region of the screen, complex gaze gestures are adopted for control command selection in order to avoid accidental gaze completion.

One complex gaze gesture, including two strokes across three active zones, is completed through hitting each active zone sequentially along the gaze gesture path. The first active zone is called an initiation field, and the last zone a completion field. The whole process is done within 1500ms. The timer event has three steps. First, the event is fired when the boundary of the initial field is crossed. Then the gaze enters the middle active zone on the path. Finally, it ends when gaze moves across the the completion field boundary.

There are three possible reasons cancel the event. First of all, if the gaze gesture is not completed within 1500ms, the event will be reset. Second, if a complex gaze gesture is initiated but the gaze enters one active zone other than one on the path, the object selection task will be cancelled. Lastly, if a complex gaze gesture is initiated and completed, but the middle zone is not hit, the system will be reset. Figure 4 (a, b) shows the two complex gaze gestures for longitudinal motion: forward and backward.

Next, we design the drone lateral movement commands. According to observations from the process of drones controlled by users, the unintentional gaze path

DOI 10.16910/jemr.7.4.4

ISSN 1995-8692

rarely occurs between two diagonal regions on the screen. Hence, the SSGG is applied to design the lateral movement commands. At the same time, in order to avoid potential overlap with longitudinal motion commands, the middle active zones on the pitch variation control path are selected as the initial gaze gesture fields for lateral movement commands, as shown in Figure 5 (a, b). The duration is set to 1000ms. The timer event is fired when an initiation field boundary is crossed and ends when gaze moves across a completion field boundary of the diagonally opposite region. Each gaze gesture has to be completed within 1000ms or the event will be reset. At the same time, if an SSGG is initiated but the gaze enters an active zone not on the path, the command selection task will be cancelled.

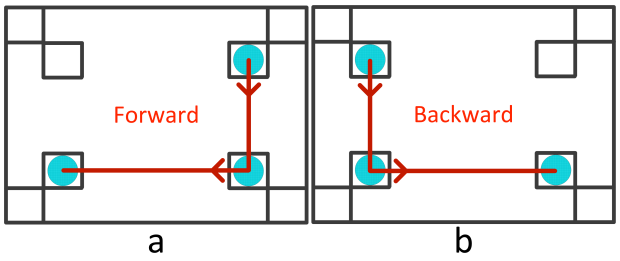

Figure 4: Complex gaze gestures represented the longitudinal motion of the drone: (a) Forward and (b) Backward.

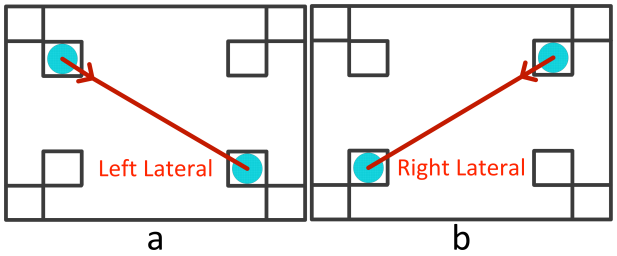

Figure 5: SSGGs represented the lateral movement of the drone: (a) Left Lateral and (b) Right Lateral.

Second group: In general, users move their gaze from the center region to the screen periphery when turning left/right and translating up/down. At the same time, fixations rarely occur on the screen periphery. Hence, we choose SSGGs as the object selection method for rotation and altitude controls.

For drone rotation, the initial fields are the active zones in two upper corners of the screen, and completion fields are the two lower corners of the screen. Each SSGG rotation command is completed within 1000ms. The SSGGs are shown in Figure 6 (a, b). The timer event is similar to that of lateral movement commands in the first group.

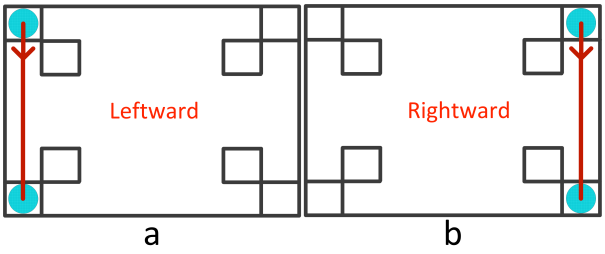

Figure 6: SSGGs represented the rotation motion of the drone: (a) Leftward and (b) Rightward.

Gaze gestures for drone altitude control are designed at the upper and lower regions of the screen. For upward translation, the initial filed is in the upper left corner of the screen and completion field is on the upper right corner of the screen. For downward translation, the initial and completion fields are located at the two lower corners of the screen. These processes have to be completed within 1000ms. Figure 7 (a, b) shows the gaze gestures for altitude control. The timer event is similar to lateral movement commands in first group.

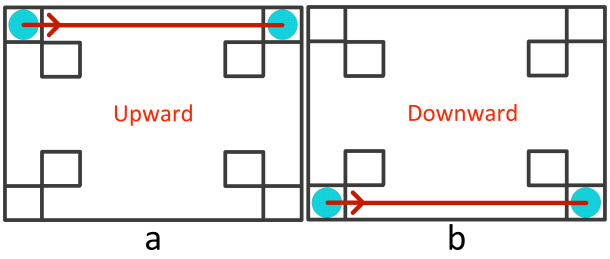

Figure 7: SSGGs represented the altitude translation of the drone: (a) Upward and (b) Downward.

We have designed control commands based on gaze gestures corresponding to the four degrees of freedom. However, we did not provide a stop command (Hover) for drone control. Stopping in drone control represents stable flying at certain distance from the ground. In order to provide an easy to remember stop command, any command in reverse order is designed to be a stop command.

According to the above designed gaze gestures for drone control, we provide a design for a screen interface between human and drone, shown in Figure 8. The white

DOI 10.16910/jemr.7.4.4

ISSN 1995-8692

rectangle regions are defined as active zones with transparent regions. The transparency of the active zones helps prevent obstruction images on the screen.

Figure 8: The designed interface between the human and drone.

Experiment

In this section, we design and execute an experiment utilizing gaze gestures object selection strategy for drone teleoperation. In order to evaluate experimental results, we also adopt other selection strategies, i.e. keyboard, joystick and dwell times, as a comparison.

Dwell Times

We adopted the similar graphic region (active zones) interface design used in (Latif, et al, 2008), as shown in Figure 9. Active zones size is the same as the ones in Figure 8. In order to better understand the meaning of active zones, labels are displayed on each active zone. The center active zone is divided into two regions, left for forward control, and right for backward control. Dwell time durations are set 300ms, 400ms, 420ms, 440ms, and 500ms. Throughout the experiment, we selected 420ms as the optimal duration time (See *Discussion* section). A control command is activated when gaze at an active zone is fixated for 420ms. If the fixation moves away from the active zone, the drone will stop (hover).

Participants

Eight subjects from different countries, three female and five male, ages 23-30 years (mean 26.3 years, SD 2.3), took part in the experiment. Some participants have vision problems; three wear glasses, the rest do not need glasses. Five participants have eye tracker experience and four had tried flying a drone controlled via smart phone. None have experience controlling a drone with an eye gaze tracker.

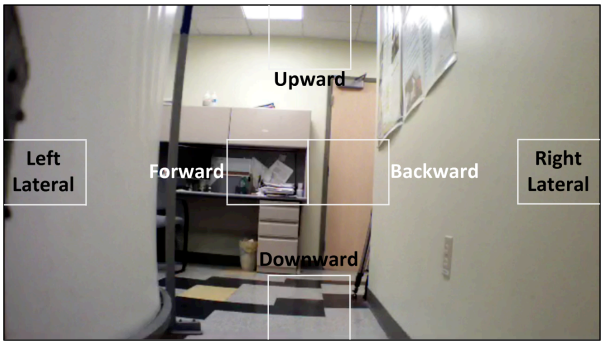

Figure 9: The designed interface for dwell times object selection strategy.

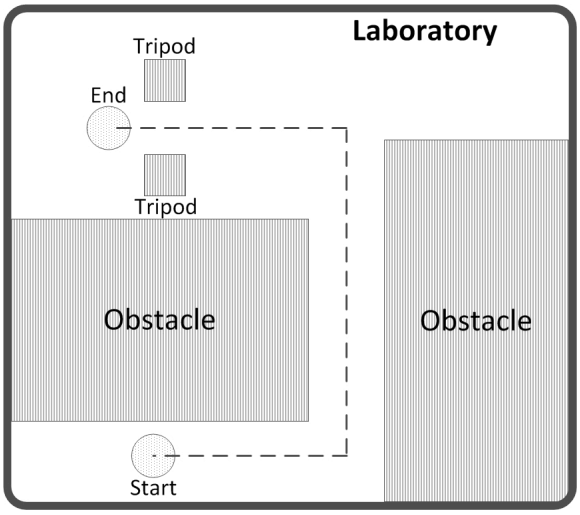

Figure 10: The pre-defined route of the drone flying.

Task

To evaluate system performance, we designed a simple task for users to perform within our laboratory. The layout in the experimental environment and pre-defined navigational route are shown in Figure 10. The start and end positions are used as landing platforms for the drone. The challenge tasked users to navigate the drone along a narrow aisle, as can be seen in Figure 10. Two tripods were placed 150cm apart and their height was set at 150cm. The drone needed to gain altitude to pass these obstacles. Participants needed to navigate the drone to pass between the tripods, and then land at the end position. The actions in designed task require use of all four

DOI 10.16910/jemr.7.4.4

ISSN 1995-8692

degrees of freedom, i.e. speed, rotation, translation and altitude control.

Study of Commands

In order to better control the drone during the experiment, we first allowed users to practice the drone locomotion commands corresponding to the four selection strategies, i.e. keyboard, joystick, dwell times, and gaze gestures. With other words, we allowed subjects practice time with the interface to learn the command gestures.

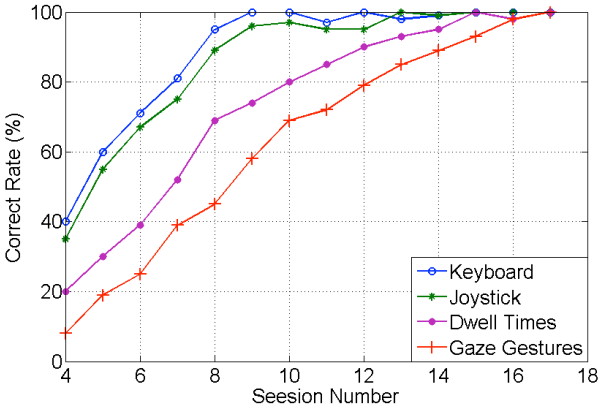

Figure 11: The distribution of learning curves for gaze gestures.

Each participant is required to consecutively complete 8 locomotion commands corresponding to longitudinal motion, lateral movement, rotation and altitude control. The process is called one seesion. The next three sessions are conducted as training sessions, where the participants are briefed about drone control commands with four selection strategies and what they to do. After the three practicing sessions, 30 minutes are given to each participant for learning commands and testing. Sessions 4 to 17 are regular testing sessions. Each session is a total of 8 commands × 8 participants = 64 commands. The correct rate is defined as the number of correct commands divided by total commands (64). The learning curve distribution for four selection strategies is shown in Figure 11. As for gaze gestures selection mode, correct rates start lower than average, and then go up to 100% after the 16th session. This indicates improvement in performance of operation over sessions. Through subjective responses regarding low correct rates, participants most commonly forgot the correct gaze gesture control commands.

Procedure

After studying and practicing the drone locomotion commands, the gaze gesture selection strategy evaluation experiment is executed.

We first provide a general explanation of the task to all participants. The HRI input devices used in the experiment include keyboard, joystick, and eye tracker. The former two devices are used for comparison with the eye tracker. Then, the user sits in front of the laptop. They are given a detailed explanation about the control strategies. Each participant needs to complete a session (Different with a session in *Study of Commands* subsection) which has four independent experiments for the four object selection strategies.

Two participants ran the full experiment a day, for four days. Each participate was given 40 minutes to practice the session before starting the test. For each test session, the drone took off automatically and elevated to about 50cm. The participant was then given full control of steering. Once the drone flew to the end position, the subject pressed a button for auto-controlled landing. Some screenshots of the drone are shown in Figure 12.

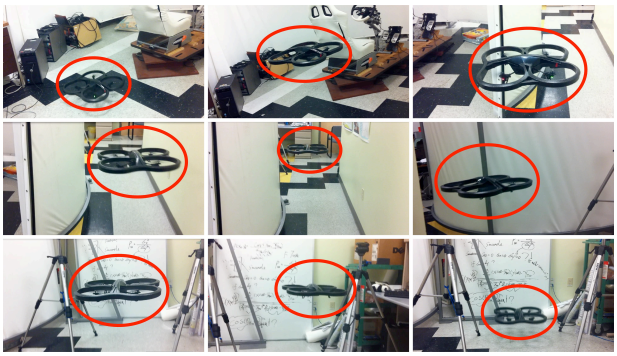

Figure 12: The screenshots of the drone flying.

Results

Objective and subjective measurements were collected for evaluation of each object selection strategy. The objective measurement is the analysis of the drone control task, which includes mean task completion time and mean error rate. The subjective measurement is the analysis of participant perception. Three one-way repeated measures ANOVAs are used as the test the hypothesis of significant effect on the four object selection

DOI 10.16910/jemr.7.4.4

ISSN 1995-8692

strategies as the independent variable with the mean task completion time, mean error rate and perception as the dependent variables.

Mean Task Completion Time

The mean task completion time is the mean elapsed time from drone take-off from the start position to landing at the end position along the pre-defined route for each subject. The recorded mean task completion time for the four object strategies are shown in Figure 13.

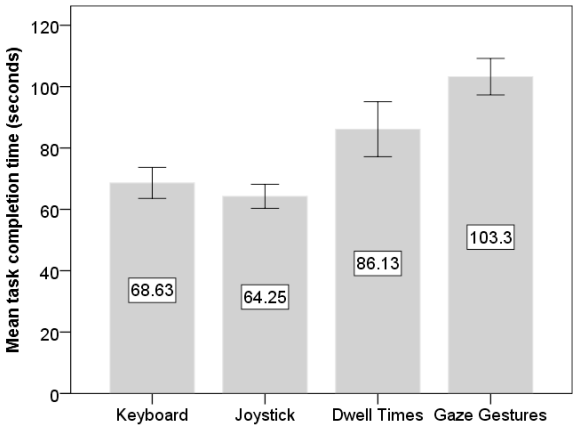

Figure 13: Mean task completion time.

A one-way within subjects repeated measures ANOVA was conducted to compare the effect of object selection strategy on the mean task completion time for keyboard, joystick, dwell time, and gaze gesture conditions. There was a significant effect of object selection strategies, $F(3,21)=79.715, \rho<0.05$. Joystick object selection provided the fastest mean task completion time $(M=64.25s, SD=1.386)$ and the gaze gestures obtained slowest mean task completion time $(M=103.25s, SD=2.2102)$. Four paired sample Bonferroni-tests were used to make post-hoc comparisons between conditions. The first and second paired sample Bonferroni-tests indicated that there were significant differences for dwell times and gaze gestures compared to keyboard and joystick. The third and fourth paired sample Bonferroni-tests indicated there also was a significant difference between dwell times and gaze gestures. The reason for the significant difference between the gaze gestures object selection strategy and other object selection strategies is that the proposed object selection strategy needs more time to complete each command corresponding to each gaze gesture.

Mean Error Rate

The mean error rate is the number of errors for each object selection strategy that occurred in the whole task, as can be seen in Figure 14. The errors include obstacle collisions and wrong commands.

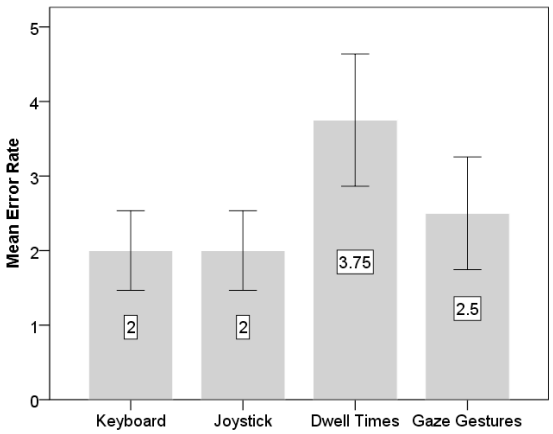

Figure 14: Mean error rate of the task.

In Figure 14, two object selection strategies with lower error rates were the keyboard and joystick. A one-way within subjects repeated measures ANOVA was conducted to compare the effect of object selection strategy on mean error rate in keyboard, joystick, dwell times, and gaze gestures conditions. There was a significant effect on object selection strategy, $F(3,21)=6.681, \rho<0.05$. Four paired sample Bonferroni-tests were used to make post-hoc comparisons between conditions. The first paired sample Bonferroni-test indicated that there was no significant difference in the score for gaze gestures, $(MD=-0.5, SD=0.378)$, compared to the keyboard. The second paired sample Bonferroni-test indicated that there was also no significant difference in the score for gaze gestures, $(MD=-0.375, SD=0.324)$, compared to the joystick. The third paired sample Bonferroni-tests indicated there was a significant difference for dwell times between the keyboard and joystick selection strategies. The fourth paired sample Bonferroni-tests indicated there was no significant difference for gaze gestures between the other three object selection strategies. As for dwell times, the users became the victim of the "Midas Touch" problem.

DOI 10.16910/jemr.7.4.4

ISSN 1995-8692

Perception

Perception evaluation is a subjective metric. In our research, we adopted the NASA Task Load Index (NASA-TLX), which is a subjective mental workload assessment tool for users working with human-machine systems, as a tool for perception evaluation of the four object selection strategies. The results are shown in Figure 15.

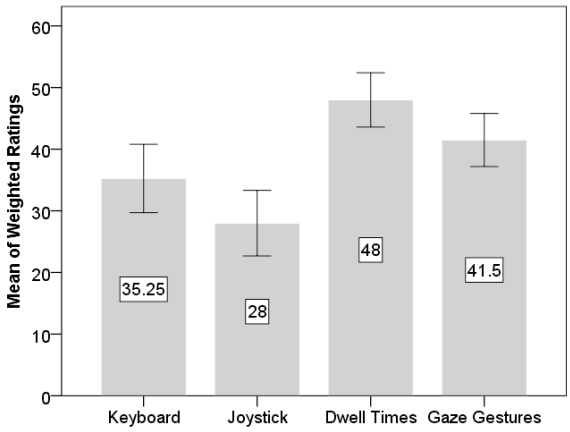

Figure 15: Mean NASA-TLX Values of the task.

A one-way within subjects repeated measures ANOVA was conducted to compare the effect of object selection strategy on mean of weighted ratings of NASA-TLX for keyboard, joystick, dwell times, and gaze gestures conditions. There was a significant effect of object selection strategy, $F(3,21) = 22.061, \rho < 0.05$. Four paired sample Bonferroni-tests were used to make post-hoc comparisons between conditions. The first paired sample Bonferroni-test indicated that there was no significant difference in scores for gaze gestures, $(MD = -2.5, SD = 1.427)$, compared to the keyboard. The second paired sample Bonferroni-test indicated that there was a significant difference in score for gaze gestures, $(MD = -9.75, SD = 2.469)$, compared to the joystick. The third paired sample Bonferroni-tests indicated there was a significant difference for dwell times between the three object selection strategies. According to feedback from users, the reasons for that are mainly attributed to the "Midas Touch" problem. The fourth paired sample Bonferroni-tests indicated there was a significant difference for gaze gestures between the joystick and dwell times.

Discussion

From the work proposed in this paper, it can be concluded that gaze gestures selection strategy is likely to play a significant role in Human-Robot Interaction applications. Although the result of mean task completion time for gaze gestures did not achieve the same level of conventional selection strategy efficiency, i.e. joystick and keyboard, from the results presented, gaze gestures showed great potential. The subjective metric showed that the gaze gestures selection strategy had less subjective mental workload for the task of teleoperating a drone, and could carry the advantage of releasing the users' hands.

In addition, we compared the results of our proposed method with the dwell times selection strategy. We had two reasons for this comparison with the dwell times condition in HRI. One is that the dwell times selection strategy has been widely used in many HRI applications (See *Related Works* section). Another is to determine what dwell durations the proposed gaze gestures selection strategy could be compared to. As for the second reason, we further explored the dwell durations and showed the compared results with the proposed gaze gestures selection strategy for our proposed task (See *Dwell Times* subsection).

In Table 1, we gave five mean task completion times with different dwell durations, ranging from 300ms to 500ms. The fastest completion time for the dwell times selection strategy is 86.13ms, with standard deviation of 8.95ms. Hence, the 420ms duration is used as the optimal time for the experiment (see *Dwell Times* subsection). As for the gaze gestures selection strategy, the mean task completion time is longer than any dwell duration. The reason is that there is always a fixation that starts the stroke, so actual selection time is longer. However, the shortest dwell duration of 300ms achieves a longer mean task completion time than other dwell durations. The Midas touch problem can be used as an explanation for that.

In Table 2, the gaze gestures selection strategy achieves lower error rate than dwell times. On one hand, active zones are put on the four corners of the screen, resulting in less accidental gesture completions. On the other hand, the gaze gestures selection strategy requires just a gaze gesture to complete a corresponding control command. However, for the dwell times selection strategy, the Midas touch problem could produce more opera-

DOI 10.16910/jemr.7.4.4

ISSN 1995-8692

tional errors according to drone control observations. In addition, shorter dwell durations had higher error numbers than the longer durations in Table 2.

Table 3 shows the performance of the gaze gestures selection strategy is much better than the dwell times selection strategy for perceptions evaluation, i.e. less metal workload, since the users need not to fixate their gaze on active zones. In addition, the gaze gestures selection strategy provides on-screen active zones and requires users to remember gaze gestures corresponding to different control commands. However, the dwell times selection strategy requires labels on the active zones for indicating what the graphic regions delegate. In this case, more active zones on the screen provide a worse effect of view for users.

To sum up, we think that the gaze gestures selection strategy has a greater potential than the dwell times selection strategy as a HRI for agents teleoperation.

Table 1

Descriptive statistics for Mean Task Completion Time of Gaze Gestures and Dwell Times.

Dependent Variable	Independent Variable	Mean	Std. Deviation
Mean Task Completion Time	Gaze Gestures	103.30	5.95
	Dwell Times 300ms	91.23	12.50
	400ms	87.60	10.42
	420ms	86.13	8.95
	440ms	86.80	10.10
	500ms	87.13	11.20

Table 2

Descriptive statistics for Mean Error Rate of Gaze Gestures and Dwell Times.

Dependent Variable	Independent Variable	Mean	Std. Deviation
Mean Error Rate	Gaze Gestures	2.50	0.89
	Dwell Times 300ms	4.50	1.01
	400ms	3.92	0.82
	420ms	3.75	0.76
	440ms	3.70	0.86
	500ms	3.81	0.95

Table 3

Descriptive statistics for Perception of Gaze Gestures and Dwell Times.

Dependent Variable	Independent Variable	Mean	Std. Deviation
Perception	Gaze Gestures	41.50	4.31
	Dwell Times 300ms	49.80	5.10
	400ms	48.90	4.20
	420ms	48.00	4.41
	440ms	49.10	4.60
	500ms	48.96	6.01

Conclusion

Teleoperated agents have been widely used to complete tasks in dangerous and unreachable environments instead of humans. The means of teleoperation are usually completed with conventional HRI input devices, e.g. keyboard, mouse and joystick, etc., for agent control. However, conventional HRI input devices are not suitable

DOI 10.16910/jemr.7.4.4

ISSN 1995-8692

for handicapped users. At the same time, users often control agents using more than two types of HRI input devices simultaneously for task completion. In this case, users are required to switch hands and attention between those interaction devices and interfaces, resulting in reduced task efficiency, increased mental workload, and even physical fatigue. Consequently, researchers have considered a novel HRI based on eye gaze with an eye tracker as an additional user input modality.

In HRI control, object selection strategy is the most frequent and important action. In our research, we introduce gaze gestures as object selection strategy for agent teleoperation. A drone is used as an example of agent teleoperation for our research. We give detailed control commands designed around gaze gestures. In order to test and validate performance of the gaze gestures selection strategy, evaluations of objective and subjective measurements are given. The objective measurement is the analysis of drone control performance, including mean task completion time and mean error rate. The subjective measurement is the analysis of participant perception. Three one-way repeated measures ANOVAs are used as the test the hypothesis of significance effect on the four object selection strategies. The results show that the gaze gestures object selection strategy has a great potential as an additional HRI used in agent teleoperation.

However, we also need to solve a problem that there can be an overlap between natural inspection patterns and intended eye movement patterns, resulting in accidental gesture completion. For gaze interaction purposes, it is desirable to minimize unintended gaze recognition (Mollenbach, 2010). Complex gaze gestures have the advantage of increasing the gaze interaction 'vocabulary', but this introduces cognitive and physiological difficulties for users, since more gaze gestures need to be remembered. Consequently, in the future, we will introduce machine learning algorithms to enhance the ability of discriminating intentional gaze gestures from otherwise normal gaze activity in agent teleoperation, e.g. gaze gestures identification based on Hierarchical Temporal Memory (HTM) algorithm (Rozado et al., 2014).

Acknowledgements

This work has been financially supported by Program of the National "985" Project -- Phase III of Beijing Institute of Technology, China Scholarship Council (No. 201306030055), U.S. National Science Foundation (NSF) through the grant No. 0954579 and No. 1333524. The suggestions from the anonymous reviewers are greatly acknowledged. Special thanks also go to the participants who have participated in this work.

References

Alapetite, A., Hansen, J. P., & Mackenzie, I., S. (2012). Demo of gaze controlled flying. *Proceedings of the 7th Nordic Conference on Human-Computer Interaction* (pp. 773-774). Copenhagen, Demark.

Drewes, H., Luca, A. D., & Schmidt, A. (2007). Eye-gaze interaction for mobile phones. *Mobility '07 Proceedings of the 4th international conference on mobile technology, applications, and systems and 1st international symposium on computer human interaction in mobile technology* (pp. 364-371).

Duchowski, A. T. (2007). *Eye tracking methodology: Theory and practice*. New York Springer.

Rozado, D., Rodriguez, F. B., & Varona, P. (2012). Low cost remote gaze gesture recognition in real time. *Applied Soft Computing, 12(8),* 2072-2084.

Fong, T., & Thorpe, C. (2001). Vehicle teleoperation interfaces. *Autonomous Robots, 11(1)*, 9-18.

Hansen, J. P., Alapetite, A., Mackenzie, I. S., & Mollenbach, E. (2014), The use of gaze to control drones. *ETRA'14 Proceedings of the Symposium on Eye Tracking Research and Applications* (pp. 27-34). Florida, US.

Heikkilä, H., & Räihä, K. J. (2009). Simple Gaze Gestures and the Closure of the Eyes as Interaction Techique. *ETRA '12 Proceedings of the Symposium on Eye Tracking Research and Applications* (pp. 147-154). Santa Barbara, US.

Huckauf, A., & Urbina, M. H. (2008). On object selection in gaze controlled environment. *Journal of Eye Movement Research, 2(4)*, 1-7.

Istance, H., Bates, R., Hyrskykari, A., & Vickers, S.(2008). Snap clutch, a moded approach to solving the Midas touch problem. *ETRA 2008 Proceedings of the 2008 symposium on Eye tracking research & applications* (pp. 221-228). New York, NY, USA.

DOI 10.16910/jemr.7.4.4

ISSN 1995-8692

Istance, H., Hyrskykari, A., Lauri, I., Mansikkamaa, S., & Vickers, S. (2010). Designing gaze gestures for gaming: An investigation of performance. *Proceedings of ETRA 2010: ACM Symposium on Eye-Tracking Research and Application* (pp. 323-330). Austin, USA.

Jacob, R. J. K. (1991). The use of eye movements in human-computer interaction techniques: What you look at is what you get. *ACM Transactions on Information Systems, 9*, 152-169.

Latif, H. O., Sherkat, N., & Lotfi, A. (2008). TeleGaze: Teleoperation through eye gaze. *7th IEEE International Conference on Cybernetic Intelligent Systems* (pp. 1-6). London, UK.

Latif, H. O., Sherkat, N., & Lotfi, A. (2009). Teleoperation through eye gaze (TeleGaze): A multimodal approach. *IEEE International Conference on Robotics and Biomimetics (ROBIO)* (pp. 711-716). Guilin, China.

Mollenbach, E. (2010). Selection strategies in gaze interaction. *PhD dissertation*, Loughborough University.

Mollenbach, E., Hansen, J. P., Liholm, M., & Gale, A. (2009). Single stroke gaze gestures. *Proceedings of the 27th International Conference: Extended Abstracts on Human Factors in Computing Systems* (pp. 4555-4560). Boston, USA.

Mollenbach, E., Hansen, J. P., & Lillholm, M. (2013). Eye movements in gaze interaction. *Journal of Eye Movement Research, 6(2)*, 1-15.

Tall, M., Alapetite, A., Agustin, J. S., Skovsgaard, H. T., Hansen, J. P., Hansen, D. W., & Mollenbach, E. (2009). Gaze-controlled driving. *Proceeding of the 27th International Conference Extended Abstracts on Human Factors in Computing Systems* (pp. 4387-4392). Boston, USA.

Urbina, M., & Huckauf, A. (2007). Dwell time free eye typing approaches. *Communication by Gaze Interaction COGAIN 2007* (pp. 65-70). Leicester, UK.

Yoo, D. H., Kim, J. H., Kim, D. H., & Chung, M. J. (2002). A human-robot interface using vision-based eye gaze estimation system. *In IROS 2002: IEEE/RSI International Conference on Intelligent Robots and Systems* (pp. 1196-1201). Lausanne, Switzerland.

Yu, M. X., Wang, X. Z., Lin, Y. Z., & Chung, M. J. (2014). Gaze tracking system for teleoperation. *26th Chinese Control and Decision Conference* (pp. 4617-4622). Changsha, China.

Zhai, S., Morimoto, C., and Ihde, S. (1999). Manual and gaze input cascaded (MGAGIC) pointing. *CHI '99. ACM Press* (pp. 246-253). Pittsburgh, USA.

Zhu, D., Gedeom, T., & Taylor, K. (2011). "Moving to center": A gaze-driven remote camera control for teleoperation. *Interacting with Computers, 23(1)*, 85-95.

DOI 10.16910/jemr.7.4.4

ISSN 1995-8692