# Single-Handed Gesture UAV Control for First Responders - A Usability and Performance User Study

**Konstantinos Konstantoudakis***
Visual Computing Lab (VCL),
Information Technologies Institute (ITI),
Centre for Research and Technology - Hellas (CERTH),
Thessaloniki, Greece †
k.konstantoudakis@iti.gr

**Georgios Albanis**
Visual Computing Lab (VCL),
Information Technologies Institute (ITI),
Centre for Research and Technology - Hellas (CERTH),
Thessaloniki, Greece

**Emmanouil Christakis**
Visual Computing Lab (VCL),
Information Technologies Institute (ITI),
Centre for Research and Technology - Hellas (CERTH),
Thessaloniki, Greece

**Nikolaos Zioulis**
Visual Computing Lab (VCL),
Information Technologies Institute (ITI),
Centre for Research and Technology - Hellas (CERTH),
Thessaloniki, Greece

**Anastasios Dimou**
Visual Computing Lab (VCL),
Information Technologies Institute (ITI),
Centre for Research and Technology - Hellas (CERTH),
Thessaloniki, Greece

**Dimitrios Zarpalas**
Visual Computing Lab (VCL),
Information Technologies Institute (ITI),
Centre for Research and Technology - Hellas (CERTH),
Thessaloniki, Greece

**Petros Daras**
Visual Computing Lab (VCL),
Information Technologies Institute (ITI),
Centre for Research and Technology - Hellas (CERTH),
Thessaloniki, Greece

## ABSTRACT

Unmanned aerial vehicles (UAVs) have increased in popularity in recent years and are now involved in many activities, professional and otherwise. First responders, those teams and individuals who are the first to respond in crisis situations, have been using UAVs to assist them in locating victims and identifying hazards without endangering human personnel needlessly. However, professional UAV controllers tend to be heavy and cumbersome, requiring both hands to operate. First responders, on the other hand, often need to carry other important equipment and need to keep their hands free during a mission. This work considers enabling first responders to control UAVs

*corresponding author
†http://vcl.iti.gr/

with single-handed gestures, freeing their other hand and reducing their encumbrance. Two sets of gesture UAV controls are presented and implemented in a simulated environment, and a two-part user study is conducted: the first part assesses the comfort of each gesture and their intuitive association with basic flight control concepts; and the second evaluates two different modes of gesture control in a population of users including both genders, and first responders as well as members of the general populace. The results, consisting of both objective and subjective measurements, are discussed, hindrances and problems are identified, and directions of future work and research are mapped out.

**Keywords**

First Responders, UAV, Gesture Recognition, User Study.

## INTRODUCTION

UAVs have recently become very popular and been utilized in many tasks in order to assist humans, such as exploration (Tomic et al. 2012), inspection (Özaslan et al. 2017), mapping (Özaslan et al. 2017; Loianno et al. 2015), interaction with the environment (Forte et al. 2012) and search and rescue operations (Michael et al. 2012). Especially in search and rescue missions, significant manpower and resources are required in order to cover large areas in as little time as possible. With drones examining the ground from above with high-definition video and thermal imaging, the time it takes to locate and rescue a missing person is greatly reduced. At the same time, potential hazards can be identified early with a UAV, without putting the human personnel in needless danger. However, flying a drone in such scenarios demands highly trained pilots and the use of cumbersome hand-held controllers that require both hands. Aiming to increase usability and intuitiveness, drones can be coupled with Mixed Reality headsets and controlled by gesture-based systems, offering new, non-invasive forms of interaction between humans and robots (L. Yuan et al. 2019).

Nevertheless, the design of such systems requires thorough investigation and the consideration of several human factors in order to eventually select natural, intuitive and ergonomic gestures (Sanders et al. 2018), especially when drones are employed by First Responders (FRs) in emergency situations. Following academic research, commercial drone manufacturers have recently introduced hand gestures as an alternative control mechanism, following two different approaches (Natarajan et al. 2018). The first approach employs specially designed gloves and wrist-mounted Inertial Measurement Unit (IMU) sensors for capturing the gesture, while the second leverages computer vision techniques by exploiting the on-board camera of the drone, thus forcing the drone to continuously point its camera towards the pilot. However, commercially available gesture commands often lack versatility and precision and are restricted to generic commands. Towards this end, the present work aims at introducing a finer control method which will eventually allow FRs to fly drones in emergency situations both intuitively and precisely. This paper focuses on multi-rotor UAVs, and in particular quadcopters, which are most often used by FR organizations in their operations.

The rest of this paper is organized as follows: Related work provides a review on current gesture-based UAV interfaces; Gesture-based UAV control describes the proposed gesture-based interface and the different control modes; UAV navigation experiment and user study outlines the experimental setup; Lastly, Conclusions and future work provides a conclusion and discusses future work. The Appendix presents the subjective evaluation questionnaire experiment participants were asked to complete.

## RELATED WORK

Even though humans normally communicate by employing both speech and gestures, gestures seem to be more suitable for representing complex ideas (Pavlovic et al. 1997). Therefore, significant research effort has been focused on developing gesture-based interfaces for controlling UAVs (Chandarana et al. 2017). Iba et al. (1999) developed a Hidden Markov model (HMM)-based recognizer in order to exploit the temporal information of gestures, while Neto et al. (2009) employed a Wii remote sensor for recognizing six different gestures and fully controlling an industrial robot. Both of these methods restricted the natural arm or hand movement by forcing the users to wear or hold a sensor. Waldherr et al. (2000) introduced a vision-based interface that was capable of recognizing both pose gestures and motion gestures; however, it relied on tracking full-body movements. In a recent study, Nagi et al. (2014), employed colored gloves in order to create a dictionary of spatial gestures, while Pfeil et al. (2013) developed a system for controlling a UAV as if it were grasped. Ng and Sharlin (2011) developed a socially motivated gesture-based interaction scheme for collocated UAVs based on a falconry metaphor. Similar to Nagi et al. (2014), Chan et al. (2018) exploited data from three different kinds of sensors, including a camera, an IMU and flex sensors in order to simultaneously track the hand pose and finger motion in a rate of over 100 frames per second. However, the ambiguity of how people perform hand gestures induces an extra challenge in detecting hand-gestures in real-time.

Trying to alleviate these obstacles, and exploiting recent advances in the deep-learning domain, Molchanov et al. (2016) introduced a three dimensional recurrent convolutional neural network (RNN) model leveraging multimodal input from depth, color, and stereo-IR sensors. Additionally, emerging technologies such as Augmented Reality (AR) and Virtual Reality (VR) have gained interest as a new interaction modality and triggered vision-based 3D hand analysis research. More specifically, authors in (Ge, Liang, et al. 2016; S. Yuan et al. 2018) have exploited depth sensors for estimating the 3D hand pose, while others (Zimmermann and Brox 2017; Spurr et al. 2018) have employed monocular RGB images for estimating the 3D locations of key hand points, which cannot however fully express the 3D shape of a hand. Recently, Ge, Ren, et al. (2019) proposed a Graph Convolutional Neural Network (Graph CNN) method to reconstruct the full 3D mesh of hand surface containing richer information of both 3D hand shape and pose. Authors in (Walker et al. 2018) introduced a system that combines the head position detected from the AR head-mounted-device (AR-HMD) with hand gestures detected by the AR-HMD camera for controlling a UAV, while L. Yuan et al. (2019) leverage an eye-tracking device in order to directly navigate a UAV in an uninstrumented environment. In Chandarana et al. (2017), the authors present a gesture-based natural language interface for defining trajectory segments using a library of twelve simple hand gestures. And in Konstantinidis et al. (2018) the authors fuse both skeletal and vision data, including optical flow, in a deep learning approach geared towards sign language gestures. Yet, most of the above emphasize more on technical implementations for prototyping purposes, while the present work presents the qualitative and quantitative metrics of a comparative user study for two different control methods of UAVs, aiming at identifying user preference and measuring the comfort, ease-to-learn, and usability of each.

## GESTURE-BASED UAV CONTROL

### Quadcopter control basics

In this work we will be focusing on small UAVs with four rotors, also known as quadcopters. In quadcopters rotors are positioned in an X shape around the UAV's body. Two of the rotors, located opposite to each other relative to the UAV's center, rotate clockwise and the other two rotate counter-clockwise, to balance angular momentum.

Quadcopters have four basic controls to which we will be referring in this work. All movements and manoeuvers are performed using a combination of these basic controls. These are:

- **Throttle**, which regulates the average rotation speed of its rotors. It provides a forces perpendicular to the plane of the UAV's rotors. Throttle is used to counter gravity and keep the UAV airborne, and also to propel the UAV in any direction, including altitude, according to its orientation, as effected by pitch and roll.
- **Yaw**, which rotates the UAV along the axis perpendicular to the plane of its rotors. This is performed by introducing a difference between the rotation speed of the clockwise-turning rotors and the counterclockwise-turning rotors.
- **Pitch**, which tilts the UAV forward or backward. This is performed by introducing a difference between the rotation speed of the forward rotors and the back rotors. Pitching forward or backward will make the UAV move forwards or backwards respectively.
- **Roll**, which tilts the UAV left or right. This is performed by introducing a difference between the rotation speed of the left-side rotors and the right-side rotors. Rolling left or right will make the UAV move to the left or to the right respectively, without changing the direction it is facing.

**Figure 1. A manual UAV remote controller.**

Classic UAV controllers (example shown in Figure 1) have two joysticks controlled by the operator's thumbs: one controlling throttle and yaw, the other controlling pitch and roll. As such, they require the use of both hands.

### Requirements

In designing finger-based UAV control for FRs in the field, a number of initial requirements were identified and used to guide the subsequent processes of selecting the modes of control as well as the individual gestures to be used. These requirements aim at providing FRs with an easy-to-use, intuitive and robust method of controlling UAVs in the field, without compromising their freedom of movement, safety, or ability to carry out their mission. Early identified requirements include:

- use of a **single hand**, so that the other hand remains free to perform other necessary actions, carry equipment, or manipulate objects even as the FR is controlling the UAV.
- that gestures are sufficiently **different from each other**. This both makes them easier to recognize and differentiate, and smooths the learning curve for the operator.
- that gestures are sufficiently **different from everyday gestures** FRs may perform during a mission. This will allow the operator to include everyday gestures when communicating with colleagues without them being mistaken for UAV commands.
- that gestures should **not be too tiring, uncomfortable, or complex**. The aim is to provide a comfortable and intuitive mode of control.
- that **the set of control gestures should be as small as possible**, to avoid confusion and be easy to learn.
- that **opposite UAV movements** (e.g. "up" and "down") should correspond to **similar but opposite gestures**, which will help to keep the gesture set small and increase control intuitiveness.
- that gestures can be interpreted and **mapped to a continuous range**, so that when giving a command (e.g. "go forward") the FR operator can also specify the intensity level of that command (e.g. "slowly" or "fast").
- that **while no command is given, the UAV should stop** and remain stationary at a constant altitude, waiting for the next command. This will allow the FR operator to temporarily shift their focus from the UAV to other matters without fearing that it may fly off or collide into a wall. It also provides an easy way to halt the UAV without requiring a specific command gesture to do so.
- that gestures should support **interoperability**, i.e. their recognition should not be tied to a particular hardware sensor, but be compatible with different modes as much as possible (e.g. infrared, IMU, vision, etc.).

### Control modes

In accordance with the requirements above, two different modes of control were identified: **finger-based** control and **palm-based** control.

In **finger-based** control, the minimum set of movements required to navigate a UAV to the desired target point and orientation was corresponded to a set of gestures. Each gesture is characterized by which fingers are extended, and the intensity level is defined by the pointing direction of a selected extended finger, or the palm itself. In order to keep the gesture set to a minimum, the following UAV commands were implemented:

- **Up/down**, to change the altitude;
- **Left/right**, to rotate along the axis perpendicular to the plane of rotors;
- **Forward**, to move forward;
- **Stop**, at the event of no other active command.

This minimum set of controls excludes some UAV movements, namely going backward or sideways. Since only a single gesture can be active at any moment, movement combinations are impossible: the operator must perform each command sequentially to position the UAV at the desired target location, making for more tedious but also more precise navigation.

In **palm-based** control, the operator uses the positioning and orientation of their open palm to control the UAV:

- A palm held **still** and horizontal to the ground denotes a stationary hovering UAV.
- **Tilting** the palm commands the UAV to tilt (**pitch** and/or **roll**) in the same direction, facilitating forward, backward, or sideways movement accordingly.
- **Turning** the palm to the left or right corresponds to **yawing** in that direction for so long as the extended fingers point otherwise than forward.
- Moving the whole palm **up or down** manipulates **throttle** and effects a change in altitude.

These palm-based commands cover all basic quadcopter controls and can be combined, allowing the full range of control for the operator.

### Gesture selection

For **finger-based** control, the gestures associated with each command (besides "Stop", which activates on an absence of any recognized gesture) were selected so as to maximize alignment with the requirements listed above. Towards this aim, a preliminary user study based on a questionnaire was performed, to ensure that selected gestures are comfortable, easy to perform and intuitive.

A set of 10 single-hand gestures, or hand states, was compiled so as to be mostly different from everyday gestures. Each gesture has certain fingers extended while all others are retracted. The list included the following:

1. Pinky extended: 
2. Pinky and thumb extended: 
3. Index and middle fingers extended: 
4. Index, middle and thumb extended: 
5. Index and pinky extended: 
6. Index, pinky and thumb extended: 
7. Middle finger extended: 
8. Middle finger and thumb extended: 
9. Index finger and thumb extended: 
10. Index, middle and thumb at right angles: 

The list excludes overly common and useful gestures, like pointing with the index finger extended, as well as overly complicated or hard combinations.

A group of 29 volunteers were then asked to perform each gesture, grade how comfortable and easy it is to perform, and note which, if any, of the three finger-based commands they can naturally associate it with. Figure 2 shows the average ease/comfort rating for each gesture as well as the popularity of associating each gesture with each of the three commands.

The results show that gestures #3 and #9 are clear favorites. The middle finger is favored only when mimicking the index finger. The pinky, on the other hand, proves difficult to combine with the index finger; however, it combines well with the thumb.

Gesture #3, besides being the easiest, was the one most readily associated with going forward, and so it was reserved for that role. Gesture #9, the second easiest, was mostly associated with the left/right command. However, in practice it proved difficult to point towards the outside of the arm with this gesture, and so it was assigned to the up/down command. Lastly, left/right was assigned to #2, second in association preference for that command.

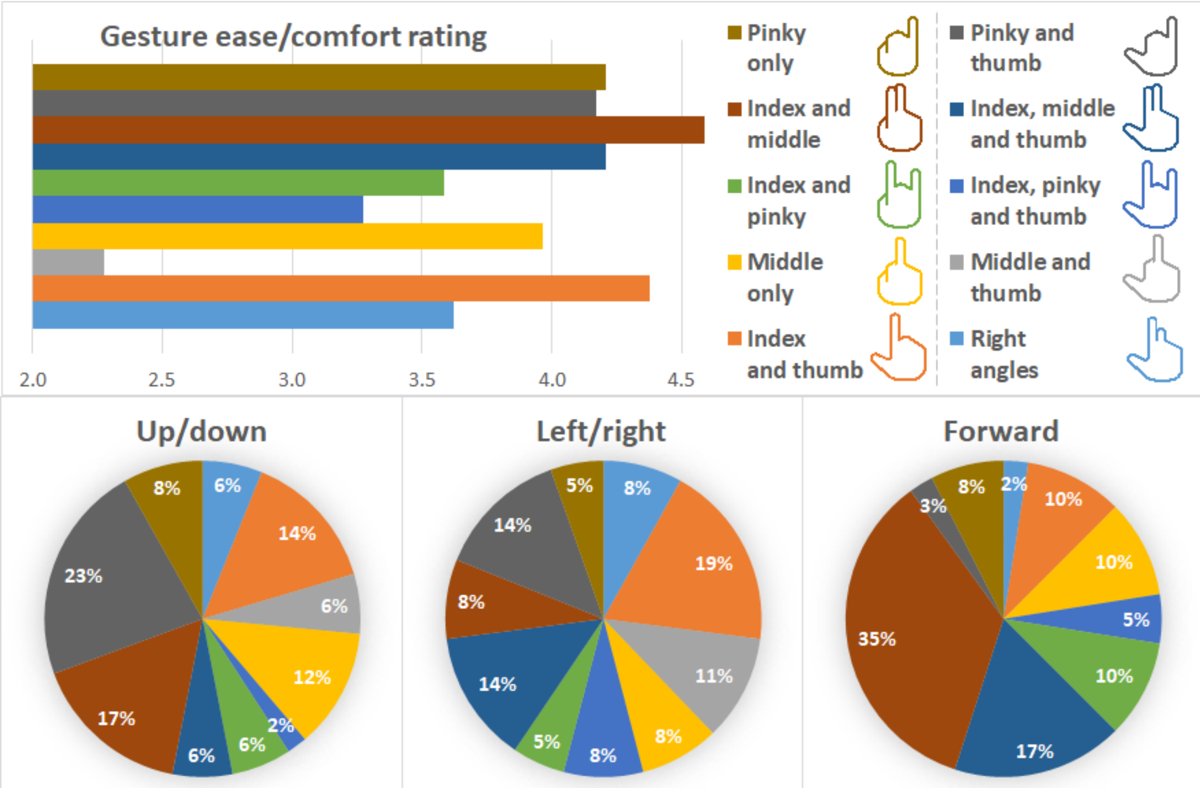


**Figure 2. Gesture questionnaire results. Top graph shows the ease and comfort rating of each gesture. The bottom graphs show the preference of gestures to be associated with each command.**

Moreover, in each gesture one of the extended fingers was designated as the pointing finger, its direction controlling the direction and intensity of each command. The final correspondence between commands and gestures, then, was defined as follows:

- **Index finger and thumb extended for up/down**, with the index finger controlling direction and intensity.
- **Pinky and thumb extended for left/right**, with the thumb pointing at the direction of yaw.
- **Index and middle fingers extended for forward**, with the index finger's angle to the vertical controlling pitch and thus velocity.

### Gesture recognition

Two different modes of gesture recognition were tested, along with their associated hardware. In this context, gesture recognition includes the recognition of the three gestures corresponding to finger-based control, the open-palm gesture (all fingers extended) for palm-based control, as well as the orientation of the palm and pointing directions for significant fingers (namely the index finger and the thumb).

One method used LeapMotion (*Leap Motion Developer* 2020) - now a part of Ultraleap (*Ultraleap* 2020) - a USB peripheral device which uses three infrared LEDs and two infrared cameras to capture the position of hands and fingers, including joints, with sub-millimeter accuracy and a high frame rate (Weichert et al. 2013). LeapMotion has the advantage of being relatively unobstructive to the user, as it needs no physical contact. On the negative side, it must be connected to the processing computer via USB and the sensor must remain stationary to receive an accurate reading of the hands, or else readings must be compensated by the sensor's own motion.

**Figure 3. A LeapMotion sensor connected to a laptop showing the virtual hands.**

The other method tested utilized AvatarVR (*AvatarVR* 2020), a wearable glove with IMUs (inertial measurement units) on each finger and the back of the palm. AvatarVR includes further features (haptic feedback, thumb flex sensing, wrist and arm IMU add-ons) that were not necessary in this test. It connects to the controlling computer by either USB or Bluetooth, and can thus allow more independence. On the downside, as a glove wearable it can be restricting. The developers urge to exercise caution when wearing it for fear of damaging it, which can be problematic for use by FRs in a crisis scenario.

**Figure 4. The AvatarVR glove.**

Preliminary testing showed that, for the purposes of finger-based UAV control, the LeapMotion sensor is more robust and accurate. AvatarVR tended to report finger curl when there was none, as a result of wrist rotation. This significantly hampered the recognition of gestures, as well as the perceived pointing direction. Upcoming SDK updates for AvatarVR may resolve this issue. For these reasons, as well as the utility concerns mentioned above, LeapMotion was selected and used for the experiment and user study presented in the following section.

## UAV NAVIGATION EXPERIMENT AND USER STUDY

### Technical setup

The two different control methods were compared in a Virtual Environment (VE) based on AirSim (Shah et al. 2018). AirSim is an open-source, cross platform simulator for drones, and cars, built on Unreal Engine but also compatible with Unity3D, supporting hardware-in-loop with popular flight controllers such as PX4 for physically and visually realistic simulations. The control methods were evaluated in two different scenarios, one indoors and one outdoors.

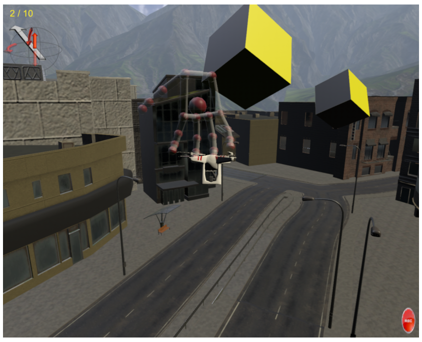

**Figure 5. Palm-based controls in the outdoor scene. Note the transparent virtual hand in an open palm position.**

For developing the outdoor scenario, the "Windridge City" scene was used, which is specifically designed for facilitating the simulations of autonomous vehicles including UAVs. The indoor scene utilises a realistic building derived from the Matterport 3D dataset (Chang et al. 2017). An introductory scene prompts users to enter their usernames and select the parameters of the experiment (i.e. the control method and the scene).

**Figure 6. finger-based controls in the indoor scene. Note the transparent virtual hand performing gesture #9.**

Additionally, collectible items were placed in each scene to encourage users to follow a path that requires the use of a diverse set of UAV movements and turn it into a playable gamified experience with a concrete goal: to collect all items, avoid collisions and take as little time as possible. In both cases, users control a virtual model of a DJI Phantom, with dimensions of 28.9 cm x 28.9 cm x 19.6 cm. Figures 5 and 6 show screenshots of the outdoor and indoor scenes respectively, including the transparent virtual hand, the virtual UAV and collectible yellow cubes.

AirSim by itself uses the four basic UAV controls: throttle, yaw, pitch and roll. Navigating the drone directly with these proved very challenging, especially for inexperienced users. Thus, a number of navigation aids were implemented to enhance user experience. These in no way detract from the experiment's realism, as most commercial UAVs already implement similar aids. These included: the application of a steady offset to throttle to counteract the UAV's weight and keep it aloft in the absence of any command; the temporary application of pitch and roll opposite to the direction of travel in the absence of command, effectively braking the UAV; and an altitude stabilization module, adjusting throttle to compensate for pitch and roll.


**Figure 7. A user trying the palm-based controls. The open palm, held above the LeapMotion sensor is tilted to the left, corresponding to a left roll for the virtual UAV.**

### Experiment parameters

The user study was conducted among a population of 31 members of the general populace (not first responders) and 8 first responders, the total consisting of 27 men and 12 women.

During the experiment, the participants had to navigate the drone firstly on the outdoor and then in the indoor scene. Their objective was to collect all the collectable items in each scene making reasonably good time while at the same time avoiding collisions. Participants played through each scene using both control methods (palm- and finger-based) and as such, they had to complete 4 rounds. A user could begin playing the outdoor scene using either the palm orientation or the finger gestures, and likewise for the indoor scene. Thus, there were 4 possible ways a user could finish all rounds. As the second playthrough of a scene could confer an advantage due to familiarization, we equally split the participants among these 4 possible combinations. Nevertheless, no correlation was detected between the order of playthroughs and the preference or performance of participants. Before the actual game-play started, they had a chance to acclimate with the controls during a 5 minute training period in the outdoor scene. In order to compare the two navigation methods participant playthrough timings and number of collisions (with walls or other scene objects) were recorded by Unity.

After completing the experiment, participants were asked to complete an online subjective evaluation questionnaire (included in the Appendix) recording independent variables such as their handedness, their job role (FR or not), their preference regarding the two different control methods, and whether they would prefer a combination of both. Additionally, subjective workload measures such as physical demand, frustration, and performance were included in the questionnaire inspired by NASA Task Load Index (TLX)(*NASA TLX* 2020), which is a workload assessment tool specifically designed for performing subjective workload assessments on operators working with human-machine interface systems.

### Results

Results were extracted from both the subjective questionnaires and the objective timings and collision counts measured by Unity.

Figure 8 shows how the general populace and first responders evaluated the two modes of UAV control in three categories: physical ease/comfort, ease to learn, and usability. Across both groups, the palm-based controls were voted as more comfortable, easy to learn, and easy to use in the simulation.

Regarding overall preference, 62% of users reported preferring the palm-based control. Figure 9 additionally shows preference statistics for different groups: FRs, members of the general populace (non-FRs), women and men. Additionally, most users (66%) reported that they would rather have the option to mix both palm and finger controls, depending on the situation, rather than be confined to a single control mode.

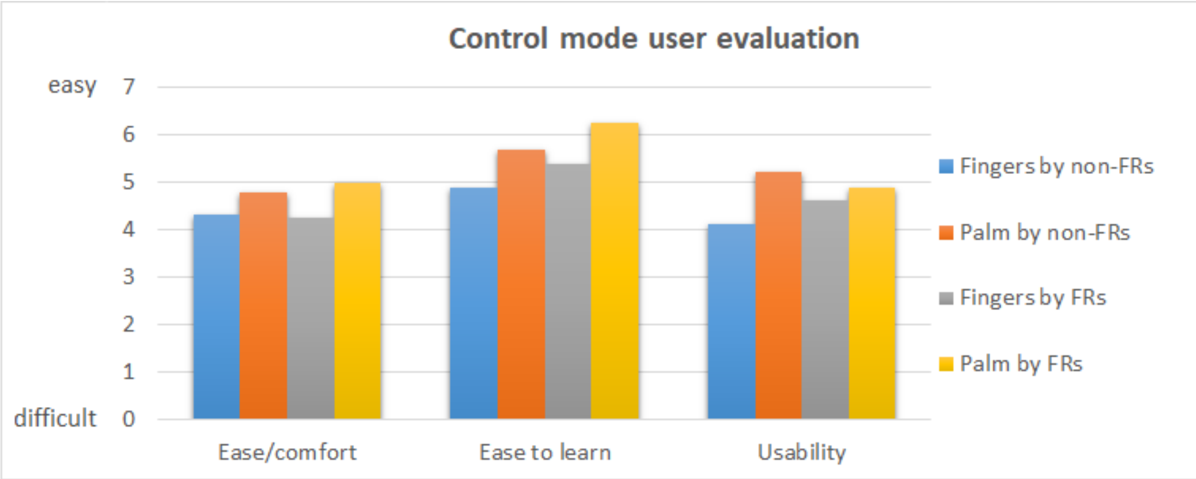


**Figure 8. Evaluation of the two control modes by users.**

Looking more closely at individual groups, it can be noted that women favored finger gestures rather than palm orientation. FRs, which will be the principal end-users of this research, were split equally between the two.

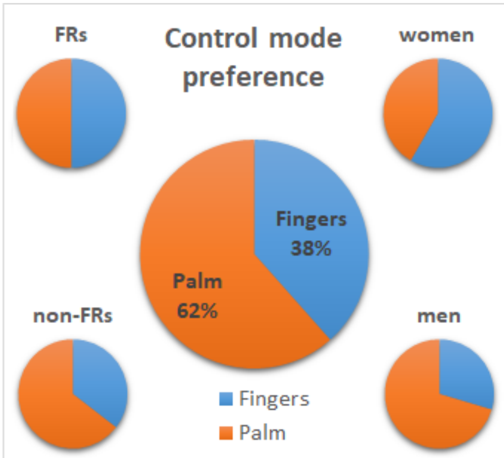


**Figure 9. Overall preference percentages for the two modes of control. The center graph shows the total among all users; the graphs on the left split users by role (FR or non-FR); and the graphs on the right split users by gender.**

Among FR participants, two, a man and a woman, were experienced UAV operators. As such, their individual evaluations are of particular interest. They both opted for the palm-based controls, as they support the same degree of versatility and combinations a conventional, manually operated UAV controller allows. They both stated, independently, that with sufficient practice they would readily swap manual controllers for palm-orientation navigation, if the latter does prove to be robust and accurate on real UAVs.

Figures 10 and 11 show objective metrics measured by Unity. As shown in Figure 10, users achieved faster navigation using the palm-based controls. This is at least partly due to the palm-based controls being able to be used in simultaneous combinations, while finger-based commands must be performed in succession. The palm-based control's superior speed is especially evident in the outdoor scene and less pronounced in the indoor one, which requires more precise movements to navigate the tighter spaces involved.

The average number of collisions, shown in Figure 11, provides another measure of the precision afforded by each control mode. The results here corroborate those of Figure 10. In the outdoor scene the palm-based controls are more precise. However, in the indoor scene, where spaces are much tighter and collision opportunities plentiful, finger-based controls performed better with less collisions. Hence, while palm-based controls might be on average faster, it is important to note that very precise movements in tight spaces may be more reliably performed with finger-based gestures.

In order to merge time and collision count into a single score metric, a 10 second penalty was applied to the time for each collision. Figure 12 shows these average penalized timings among all participants and for individual groups.

This combined objective performance graph shows that overall, the palm-based controls are slightly better for performance. While non-FRs and men achieved lower timings with the palm-based controls, FRs and women performed better with finger gestures. This is in line each group's subjective preference as shown in Figure 8.

It can be noted that groups which favored the palm-based controls performed better, overall, compared to groups who favored fingers. Hence, a conclusion may be reached that palm orientation, which allows more complex moves

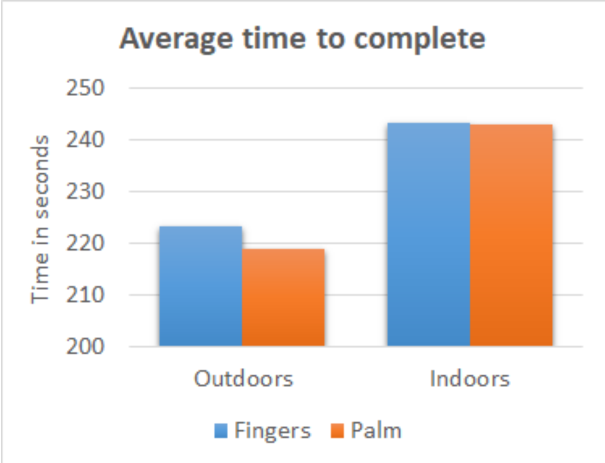


**Figure 10. Average time in seconds for completion of the outdoor and indoor scenes using each control mode.**

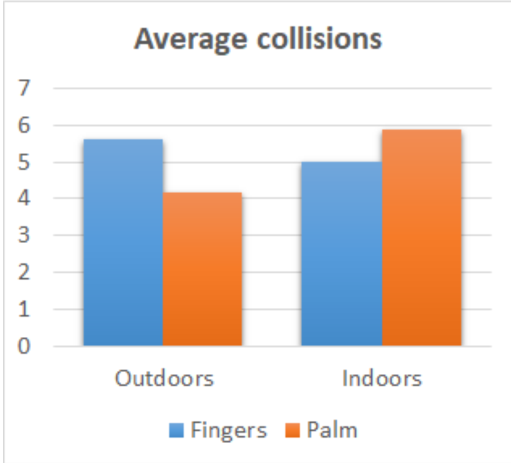


**Figure 11. Average collisions in the outdoor and indoor scenes using each control mode.**

and combinations, is better for more experienced operators while finger-based gestures, which are geared towards simplicity and precision, are more suitable for less experienced ones.

Regarding handedness, left-handed participants naturally performed more poorly on average, as they were force to use their right hand. They uniformly achieved both faster timings and less collisions using the finger-based controls compared to palm orientation, which is in accord with the above conclusion.

#### Identified problems

Besides the subjective and objective results, testing with a wide group of users provided a chance to identify problems that were not readily apparent during initial development.

One major hindrance proved to be the robustness of the hand state estimation. This was not so much of a problem with the palm-based controls, as the open palm, all fingers extended, was detected easily and consistently. However, the individual gestures of the finger-based control set were often not recognized correctly. Recognition robustness varied depending on user, and some participants experienced intense frustration as their hands and fingers were not recognized correctly. It was noted that the fingers and hand states of users with thicker fingers and big hands were recognized more consistently.

Recognition robustness also varied with the hand's position relative to the sensor, as well as the attempted gesture. In some combination of gestures and relative positions, key fingers were occluded by other fingers or the palm; users overcame this by changing the positioning of their hands when gestures were not recognized.

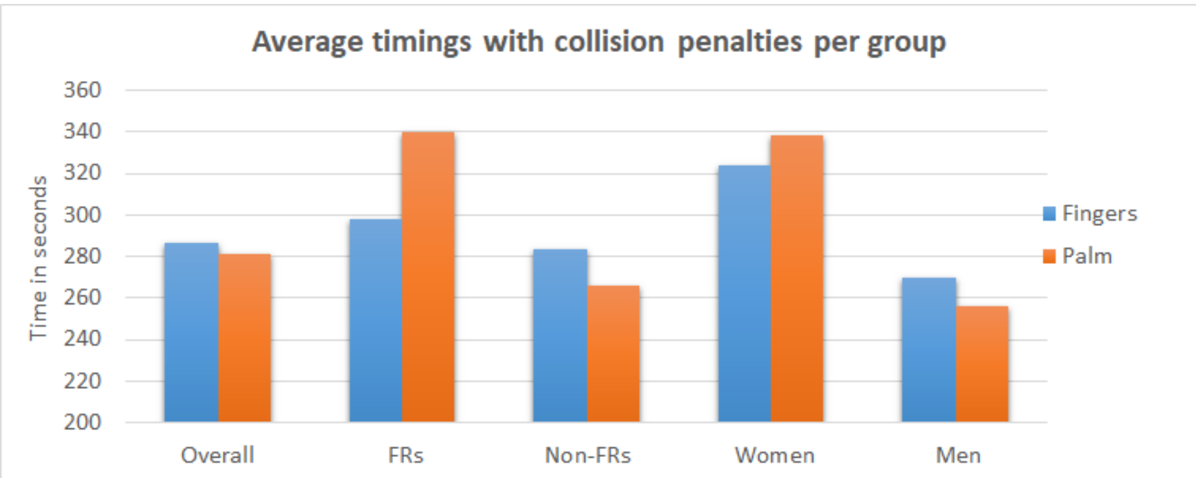


**Figure 12. Average timings with a 10 second penalty applied for each collision: Overall among all participants, and separately for particular groups.**

It is possible that the users' overall preference for the palm-based controls is at least partly due to this problem. This problem is attributed to the sensor hardware, and it is possible that with future versions or firmware updates it might be alleviated.

## CONCLUSIONS AND FUTURE WORK

In this paper the authors consider the task of providing efficient and intuitive UAV gesture control for first responders on the field. A set of user and technical requirements was formulated and two modes of control were designed and evaluated, with one (based on discrete gestures) restricting UAV movement combinations and thus providing more precision and the other (based on the orientation of the palm) allowing the full range of complex moves.

A preliminary user study was carried out to identify the most comfortable hand gestures and their perceived associations with UAV controls. Then the two different modes of control were evaluated by users in a simulated environment in a final user study. Both members of the general populace and actual first responders took part in this use study, which included both objective performance metrics and subjective comfort and usability polling in the form of a questionnaire.

The results of the user study were presented and discussed, noting overall preference as well as preference by role (FRs vs. the general populace) and gender. The palm-based controls emerged as more comfortable, easy to learn, and accurate subjectively, with objective metrics corroborating this conclusion. At the same time, problems and complications were identified, including both hardware limitations and implementation oversights.

As this is a work in progress, the authors plan to improve and expand on this research in several ways:

Due to time constraints, in this early stage only right-handed gestures were recognized, which naturally made it difficult for left-handed users. **Left hand support** will be straightforward to implement, and is within the authors' plans for the immediate future.

Though the orientation of the palm was detected consistently and accurately, the recognition of individual gestures of the finger-based control set was not so robust. This proved to be a major point of frustration among some users, and doubtlessly impacted on that control set's less positive evaluation. **Improving the robustness** of gesture recognition will be an important part of future research.

As mentioned before, in the implementation used for this user study gestures were recognized with LeapMotion, an infrared camera based hand tracking peripheral. The authors plan to **explore more hand tracking hardware and modalities** and identify one which will increase recognition robustness while satisfying to the greatest possible extent operational concerns such as portability, low power consumption, and low cost. Towards this end, AvatarVR's upcoming SDK update will be tested, perhaps having solved some finger tracking problems that were encountered. An interesting path of research, and one the authors intend to pursue, will be purely vision-based hand tracking, such as *Google AI Blog* (2020). This can both reduce cost, as it will be compatible with any camera, and possibly prove more robust and versatile.

In this work the authors have focused on static hand pose or hand state gestures, rather than gestures as a movement of the hand in time. This allowed a finer control of the intensity of each command, as well as automatic braking when the hand state stopped corresponding with the predefined gestures; both of the above are important in UAV control, especially in crisis situations. However, **gestures with a time component** could be incorporated and corresponded to more complex UAV commands, such as returning to the operator or circling around a point in space.

Ultimately, of course, the authors' goal is to move beyond simulation and **test finger-based control on real UAVs**, integrating with the various built-in features included in individual models, such as obstacle avoidance and more complex navigation commands. Real UAV testing will start within the next few months, and more options and improvements will be incorporated over time.

## ACKNOWLEDGMENTS

This research has been partially supported by the European Commission funded program FASTER, under H2020 Grant Agreement 833507. The authors would like to thank the Hellenic Rescue Team, branch of Thessaloniki (*Hellenic Rescue Team* 2020), for taking part in the user study and providing valuable feedback.

**REFERENCES**

*AvatarVR* (2020). URL: https://avatarvr.es/ (visited on 01/23/2020).

Chan, T. K., Yu, Y. K., Kam, H. C., and Wong, K. H. (2018). "Robust hand gesture input using computer vision, inertial measurement unit (IMU) and flex sensors". In: *2018 IEEE International Conference on Mechatronics, Robotics and Automation (ICMRA)*. IEEE, pp. 95–99.

Chandarana, M., Meszaros, E. L., Tmjillo, A., and Allen, B. D. (2017). "Analysis of a gesture-based interface for uav flight path generation". In: *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, pp. 36–45.

Chang, A., Dai, A., Funkhouser, T., Halber, M., Niessner, M., Savva, M., Song, S., Zeng, A., and Zhang, Y. (2017). "Matterport3d: Learning from rgb-d data in indoor environments". In: *arXiv preprint arXiv:1709.06158*.

Forte, F., Naldi, R., Macchelli, A., and Marconi, L. (2012). "Impedance control of an aerial manipulator". In: *2012 American Control Conference (ACC)*. IEEE, pp. 3839–3844.

Ge, L., Liang, H., Yuan, J., and Thalmann, D. (2016). "Robust 3d hand pose estimation in single depth images: from single-view cnn to multi-view cnns". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3593–3601.

Ge, L., Ren, Z., Li, Y., Xue, Z., Wang, Y., Cai, J., and Yuan, J. (2019). "3d hand shape and pose estimation from a single rgb image". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 10833–10842.

*Google AI Blog* (2020). URL: https://ai.googleblog.com/2019/08/on-device-real-time-hand-tracking-with.html (visited on 01/23/2020).

*Hellenic Rescue Team* (2020). URL: http://www.hrt.org.gr/hellenic-rescue-team.en.aspx (visited on 01/23/2020).

Iba, S., Weghe, J. M. V., Paredis, C. J., and Khosla, P. K. (1999). "An architecture for gesture-based control of mobile robots". In: *Proceedings 1999 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human and Environment Friendly Robots with High Intelligence and Emotional Quotients (Cat. No. 99CH36289)*. Vol. 2. IEEE, pp. 851–857.

Konstantinidis, D., Dimitropoulos, K., and Daras, P. (2018). "A deep learning approach for analyzing video and skeletal features in sign language recognition". In: *2018 IEEE International Conference on Imaging Systems and Techniques (IST)*. IEEE, pp. 1–6.

*Leap Motion Developer* (2020). URL: https://developer.leapmotion.com/ (visited on 01/23/2020).

Loianno, G., Thomas, J., and Kumar, V. (2015). "Cooperative localization and mapping of MAVs using RGB-D sensors". In: *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 4021–4028.

Michael, N., Shen, S., Mohta, K., Mulgaonkar, Y., Kumar, V., Nagatani, K., Okada, Y., Kiribayashi, S., Otake, K., Yoshida, K., et al. (2012). "Collaborative mapping of an earthquake-damaged building via ground and aerial robots". In: *Journal of Field Robotics* 29.5, pp. 832–841.

Molchanov, P., Yang, X., Gupta, S., Kim, K., Tyree, S., and Kautz, J. (2016). "Online detection and classification of dynamic hand gestures with recurrent 3d convolutional neural network". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4207–4215.

Nagi, J., Giusti, A., Gambardella, L. M., and Di Caro, G. A. (2014). "Human-swarm interaction using spatial gestures". In: *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 3834–3841.

*NASA TLX* (2020). URL: https://humansystems.arc.nasa.gov/groups/tlx/ (visited on 01/23/2020).

Natarajan, K., Nguyen, T.-H. D., and Mete, M. (2018). "Hand gesture controlled drones: An open source library". In: *2018 1st International Conference on Data Intelligence and Security (ICDIS)*. IEEE, pp. 168–175.

Neto, P., Pires, J. N., and Moreira, A. (2009). "High-level programming for industrial robotics: using gestures, speech and force control". In: *IEEE International Conference on Robotics and Automation, Kobe*.

Ng, W. S. and Sharlin, E. (2011). "Collocated interaction with flying robots. In 2011 RO-MAN". In: *Atlanta, GA*, pp. 143–149.

Özaslan, T., Loianno, G., Keller, J., Taylor, C. J., Kumar, V., Wozencraft, J. M., and Hood, T. (2017). "Autonomous navigation and mapping for inspection of penstocks and tunnels with MAVs". In: *IEEE Robotics and Automation Letters* 2.3, pp. 1740–1747.

Pavlovic, V. I., Sharma, R., and Huang, T. S. (1997). "Visual interpretation of hand gestures for human-computer interaction: A review". In: *IEEE Transactions on pattern analysis and machine intelligence* 19.7, pp. 677–695.

Pfeil, K., Koh, S. L., and LaViola, J. (2013). "Exploring 3d gesture metaphors for interaction with unmanned aerial vehicles". In: *Proceedings of the 2013 international conference on Intelligent user interfaces*, pp. 257–266.

Sanders, B., Vincenzi, D., Holley, S., and Shen, Y. (2018). "Traditional vs gesture based UAV control". In: *International Conference on Applied Human Factors and Ergonomics*. Springer, pp. 15–23.

Shah, S., Dey, D., Lovett, C., and Kapoor, A. (2018). "Airsim: High-fidelity visual and physical simulation for autonomous vehicles". In: *Field and service robotics*. Springer, pp. 621–635.

Spurr, A., Song, J., Park, S., and Hilliges, O. (2018). "Cross-modal deep variational hand pose estimation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 89–98.

Tomic, T., Schmid, K., Lutz, P., Domel, A., Kassecker, M., Mair, E., Grixa, I. L., Ruess, F., Suppa, M., and Burschka, D. (2012). "Toward a fully autonomous UAV: Research platform for indoor and outdoor urban search and rescue". In: *IEEE robotics & automation magazine* 19.3, pp. 46–56.

*Ultraleap* (2020). URL: https://www.ultraleap.com/ (visited on 01/23/2020).

Waldherr, S., Romero, R., and Thrun, S. (2000). "A gesture based interface for human-robot interaction". In: *Autonomous Robots* 9.2, pp. 151–173.

Walker, M., Hedayati, H., Lee, J., and Szafir, D. (2018). "Communicating robot motion intent with augmented reality". In: *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, pp. 316–324.

Weichert, F., Bachmann, D., Rudak, B., and Fisseler, D. (2013). "Analysis of the accuracy and robustness of the leap motion controller". In: *Sensors* 13.5, pp. 6380–6393.

Yuan, L., Reardon, C., Warnell, G., and Loianno, G. (2019). "Human gaze-driven spatial tasking of an autonomous MAV". In: *IEEE Robotics and Automation Letters* 4.2, pp. 1343–1350.

Yuan, S., Garcia-Hernando, G., Stenger, B., Moon, G., Yong Chang, J., Mu Lee, K., Molchanov, P., Kautz, J., Honari, S., Ge, L., et al. (2018). "Depth-based 3d hand pose estimation: From current achievements to future goals". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2636–2645.

Zimmermann, C. and Brox, T. (2017). "Learning to estimate 3d hand pose from single rgb images". In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4903–4911.

## APPENDIX: USER EVALUATION QUESTIONNAIRE

*Participants were asked to complete the following questionnaire after completing using the UAV simulator with both control sets (finger-based and palm-based):*

### User info and demographics

Username: *(this was included to allow correlation with the objective metrics extracted by Unity)*

Gender:

- Female
- Male

Job role:

- First responder
- Other

Handedness:

- Left-handed
- Right-handed
- Ambidextrous

I consent to have my answers in this questionnaire and my performance metrics in the UAV recorded and used to extract anonymized statistics and evaluation results regarding gesture based UAV controls.

- Yes
- No

### Finger-based controls

Physical ease/comfort: *(range of 1-7)*

- 1: Not comfortable at all, very tiring
- 7: Very comfortable, not tiring at all

Ease to learn: *(range of 1-7)*

- 1: Hard to learn
- 7: Easy to learn

Usability: *(range of 1-7)*

- 1: Frustrating, I can't make the UAV move as I intend
- 7: It is very easy to move the UAV as I intend

#### Palm-based controls

Physical ease
comfort: *(range of 1-7)*

- 1: Not comfortable at all, very tiring
- 7: Very comfortable, not tiring at all

Ease to learn: *(range of 1-7)*

- 1: Hard to learn
- 7: Easy to learn

Usability: *(range of 1-7)*

- 1: Frustrating, I can't make the UAV move as I intend
- 7: It is very easy to move the UAV as I intend

#### Comparison

I would prefer to control the UAV using:

- finger-based gestures
- palm orientation

I would rather:

- always use the method I selected in the previous question
- have the option to use both, depending on the situation and the way I want the UAV to move