

Article

Wearable Drone Controller: Machine Learning-Based Hand Gesture Recognition and Vibrotactile Feedback

Ji-Won Lee ¹ and Kee-Ho Yu ^{2,3,*}¹ KEPCO Research Institute, Daejeon 34056, Republic of Korea² Department of Aerospace Engineering, Jeonbuk National University, Jeonju 54896, Republic of Korea³ Future Air Mobility Research Center, Jeonbuk National University, Jeonju 54896, Republic of Korea

* Correspondence: yu@jbnu.ac.kr

Abstract: We proposed a wearable drone controller with hand gesture recognition and vibrotactile feedback. The intended hand motions of the user are sensed by an inertial measurement unit (IMU) placed on the back of the hand, and the signals are analyzed and classified using machine learning models. The recognized hand gestures control the drone, and the obstacle information in the heading direction of the drone is fed back to the user by activating the vibration motor attached to the wrist. Simulation experiments for drone operation were performed, and the participants' subjective evaluations regarding the controller's convenience and effectiveness were investigated. Finally, experiments with a real drone were conducted and discussed to validate the proposed controller.

Keywords: human–drone interface; wearable device; hand gesture recognition; machine learning; vibrotactile feedback



Citation: Lee, J.-W.; Yu, K.-H. Wearable Drone Controller: Machine Learning-Based Hand Gesture Recognition and Vibrotactile Feedback. *Sensors* **2023**, *23*, 2666. <https://doi.org/10.3390/s23052666>

Academic Editors: Enrico Vezzetti, Andrea Luigi Guerra, Gabriele Baronio, Domenico Speranza and Luca Ulrich

Received: 10 January 2023

Revised: 14 February 2023

Accepted: 23 February 2023

Published: 28 February 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Nowadays, multicopter drones have been widely used because of their simple mechanism, control convenience, and hovering feature [1]. Drones are important in surveillance and reconnaissance, aerial photography and measurement, search and rescue missions, communication relay, and environmental monitoring [2,3]. To complete such applications and missions, highly sophisticated and dexterous drone control is required. Autonomous control has been partly used in their applications, as in waypoint following and programmed flight and mission, because of limited autonomy [4–6]. However, in the autonomous flight of drones, sometimes the autopilot is switched to manual control by a human operator according to the flight phase, such as landing, and in unexpected circumstances. The human role is necessary in the control loop when the system cannot fully reach an autonomous state.

Therefore, natural user interfaces for human operators have been studied extensively. An early study reported a novel user interface for manual control based on gestures, haptics, and PDA [7]. Moreover, multimodal natural user interfaces, such as speech, gestures, and vision for human–drone interaction, were introduced [8,9]. Recently, hand gesture-based interfaces and interactions using machine learning models have been proposed [10–15]. Hand gestures are a natural way to express human intent, and their effectiveness for applications of human–machine/computer interface/interaction has been reported in previous works. Some of the applications were focused on the control of drones based on deep learning models [16–19]. They used vision, optical sensors with infrared lights, and an inertial measurement unit (IMU) to capture the motion of the hand. The IMU attached to the user's hand senses the motion of the hand robustly compared to conventional vision systems, which are easily affected by light conditions and require tedious calibrations.

In contrast, to interact with a machine/computer, tactile stimulation has been adopted as a feedback means to humans for a long time [20–22]. Tactile stimulation is an additional channel to visual information for providing necessary information to humans. In

particular, tactile feedback is important for visually impaired persons in terms of representing surrounding circumstances and the environment instead of having to rely on visual information [23,24]. Even for sighted people, tactile feedback helps improve their understanding of the environment when visual information is insufficient or blocked. Some previous work related to tactile and force feedback for drone control was reported [25–27]. Tsykunov et al. [25] studied a tactile pattern-based control glove for cluster flight of nano-drones. Cluster flights were intuitive and safe, in which the user's position had to move continuously to guide the drone. Rognon et al. [26] addressed the flyjacket to control the drone with torso movements using IMU. The cable-driven haptic feedback system was mounted on the elbows of the user, who navigated the direction of waypoints. The flyjacket was demonstrated to control drones as much as joysticks and tablets, while the torso of the user was restricted. Duan et al. [27] developed a human–computer interaction system for improving realistic VR/AR. The tactile feedback device was used for estimating environments and objects. The gesture-based control was implemented with Leap Motion and Neural Network, which tend to be affected by light conditions.

We studied and implemented a wearable drone controller with hand gesture recognition and vibrotactile feedback, and the basic idea of the interface was presented as a preliminary version of this paper [28]. The hand gestures are sensed by the IMU, having robust features for motion acquisition compared to conventional vision placed on the back of the hand. The dominant parameters of the motion are extracted through sensitive analysis, and then machine learning-based classifications are performed. The classified and recognized hand gestures are commanded to the drone, and the distance to obstacles in the heading direction of the drone measured by the ultrasonic sensor is represented by activating the vibration motor attached to the user's wrist. The IMU-based hand motion capture is relatively free in the distance between the drone and the user, and the drone does not need to make the pose to acquire the hand motion visually. Vibrotactile feedback is an effective way to obtain obstacle information around a drone, especially when the visual information is limited or blocked during operation of the drone.

The remainder of this paper is organized as follows. The system configuration of the wearable drone controller is presented in Section 2, and hand gesture recognition based on machine learning models is analyzed in Section 3. The simulation experiment with subjective evaluation of participants is performed in Section 4, and a real drone experiment for validation is presented in Section 5. Finally, Section 6 presents the conclusion.

2. System Configuration

The controller developed here comprised vibrotactile feedback and hand gesture-based drone control. An ultrasonic sensor was mounted on the drone head to detect obstacles. A vibrator in the controller was stimulated according to the distance between the drone and its obstacles, as shown in Figure 1a. When the operator recognizes the vibration, the operator commands the gesture to avoid its obstacle (Figure 1b). The gestures were sensed by an IMU attached to the back of the hand. The gestures were categorized into two groups to control the drone appropriately. One of the control methods, called the direct mode, was defined to control drones directly for movements such as roll, pitch, and up/down. The cruise velocity of the drone was determined from the inclination of the IMU attached to the hand. Another control method, called the gesture mode, was used to control the drone with hand gestures more easily but may not perform quantitatively, as in the direct mode. The patterns of hand gestures used in the gesture mode were defined by imitating hand motions for helicopter guidance. Gesture pattern analysis and classification were conducted through machine learning to recognize these gestures, which are more sizable than the direct mode.

All sensors were connected to the Raspberry Pi 3b (Raspberry Pi Foundation, Cambridge, UK), as shown in Figure 2. The signal of the IMU was processed at 50 Hz and transmitted to the PC for gesture classification using MATLAB/Simulink. The classified gestures were also delivered to the AR drone control module of LabVIEW to command the

drone. The ultrasonic sensor measured the distance, with the signal at 40 kHz. The distance information is transmitted wirelessly to the Raspberry Pi of the controller and converted to vibration intensity. The overall shape of the controller was fabricated using 3D printing.

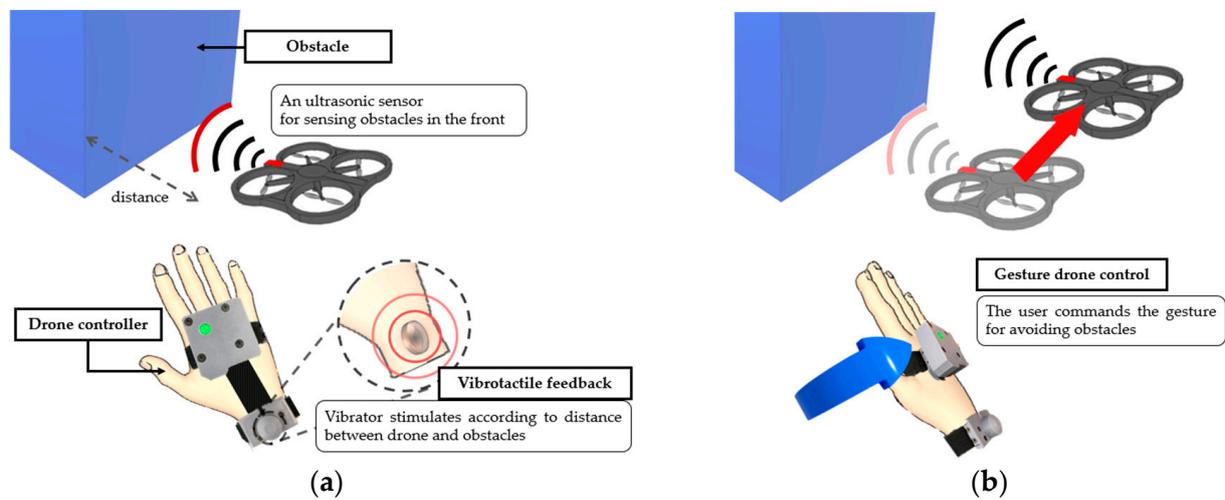


Figure 1. Concept of the proposed controller with vibrotactile feedback and gesture-based drone control. (a) Obstacle is detected in front of the drone, and the vibrator is stimulated. (b) Gesture of right direction is performed to avoid obstacle using gesture control.

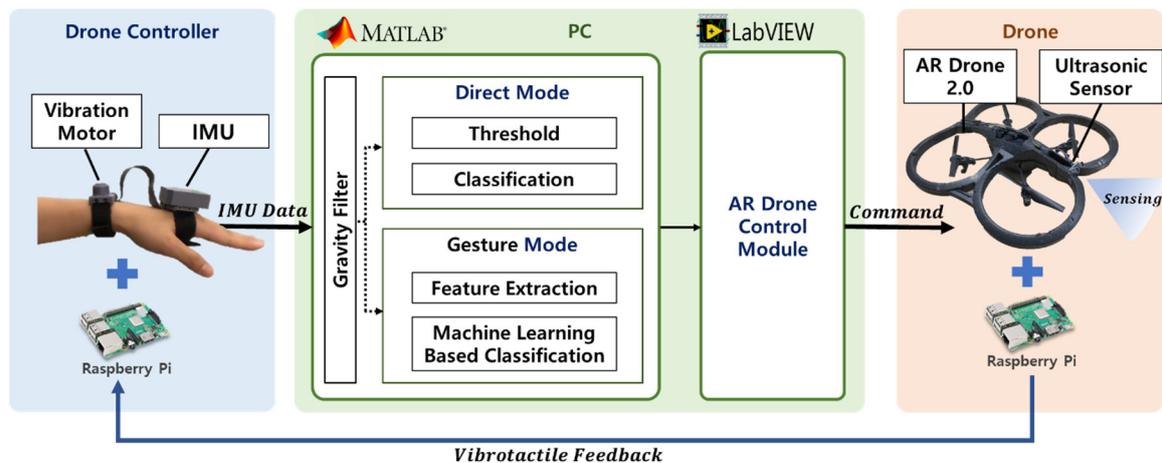


Figure 2. System configuration of wearable drone interface comprises a drone controller, hand gesture recognition and drone control module in PC, and drone.

2.1. Vibrotactile Feedback

In general, manual drone control is based on visual feedback. However, obstacles in the operating environment might interfere with visual feedback, which occasionally causes drone accidents. Hence, other sensory feedback, such as tactile and/or audio, is required to transfer obstacle information effectively.

In this study, vibrotactile feedback was adopted to represent obstacle information. An ultrasonic sensor (HC-SR04, SparkFun Electronics, Colorado, USA) was mounted on the drone head with a Raspberry Pi to operate the sensor so as to detect obstacles in front. The reliable detection region of the sensor was determined to be 0.03 to 2 m within $\pm 15^\circ$. The measured distance is transmitted to the Raspberry Pi of the controller to generate vibrotactile stimulation.

A coin-type vibration motor (MB-1004V, Motorbank Inc., Seoul, Korea) was used as the actuator for tactile stimulation. The motor's size (diameter \times height) was 9.0×3.4 mm, and its small size is suitable for wearable devices. The motor is attached to the wrist, which

allows free hand motion for the intended gesture without any movement restriction. To effectively deliver vibration to the user, a tactile stimulator was designed as a bracelet, which comprised a spring and hemispherical plastic under the vibration motor, as shown in Figure 3.

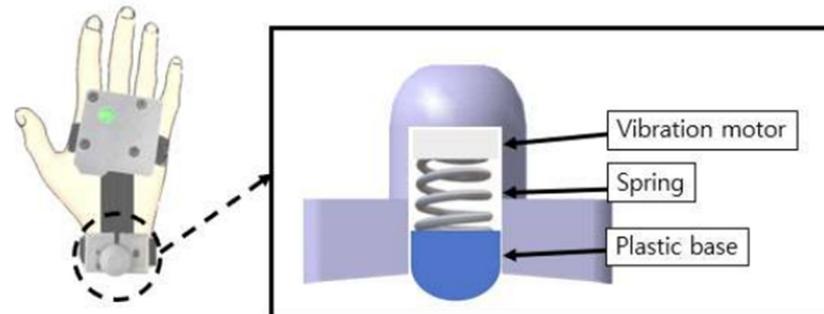


Figure 3. Structure of vibrotactile stimulator.

Based on previous related studies [29–31], an amplitude of 100 μm and a frequency of 96.2 Hz were fixed as the stimulation conditions to satisfy the vibrotactile threshold of detection. The vibration intensity changes according to the distance between the drone and the obstacle. A preliminary experiment was conducted to determine appropriate stimulation conditions for the vibration motor. The vibration intensity was controlled by modulating the pulse width (PWM) applied to the vibration motor. The duty rate was changed by two levels with a step change because a minimum stimulation difference is required for human tactile perception [20]. At first, the duty rate of 50% is given as an initial vibration intensity when the measured distance to an obstacle is within 1.0 m, and then the duty rate is raised to 100% for strong vibration intensity when the measured distance is within 0.5 m and the drone is getting closer to the obstacle.

2.2. Gesture-Based Drone Control

Gestures facilitate the delivery of intentions, emotions, thoughts, and nonverbal communication. In particular, hand gestures have been widely used as effective and easy communication methods [32,33]. Hand gesture-based control provides intuitive and portable drone control. There have been studies on hand gesture recognition using vision-based sensors, such as RGB and depth cameras, and Kinect sensors [34,35]. However, these systems have limitations in terms of light, angle, and position of the environment. To overcome this problem, we propose a wearable sensor-based gesture recognition system. The wearable sensor-based method includes electromyography (EMG) and IMU, etc. [36,37]. We used an IMU (SEN-14001, SparkFun Electronics, Colorado, USA) which measures the 3-dimensional angle, angular velocity, and linear acceleration.

As mentioned above, gestures were categorized into two groups corresponding to control purposes. The direct mode was defined to match drone movements such as roll, pitch, and up/down, as shown in Figure 4. The roll, pitch, and z-axis acceleration of the user's hand were calculated using an inertial sensor and mapped to the drone's posture and speed. The available hand motion range and its rate were considered to avoid the recognition of unintended hand gestures. A roll and pitch motion of less than $\pm 30^\circ$ and linear acceleration of less than ± 10 m/s are disregarded for the stable recognition of hand gestures.

The gesture mode shown in Figure 5 is defined by imitating the hand signal used for helicopter guidance from the ground operator [38,39]. The gesture mode has difficulty controlling the drone quantitatively, as in the direct mode, but can be used more easily to control flight direction. These hand gestures are generated naturally with individual deviations; therefore, the patterns should be analyzed and classified for accurate recognition. This study adopted machine learning to learn and classify hand gestures.

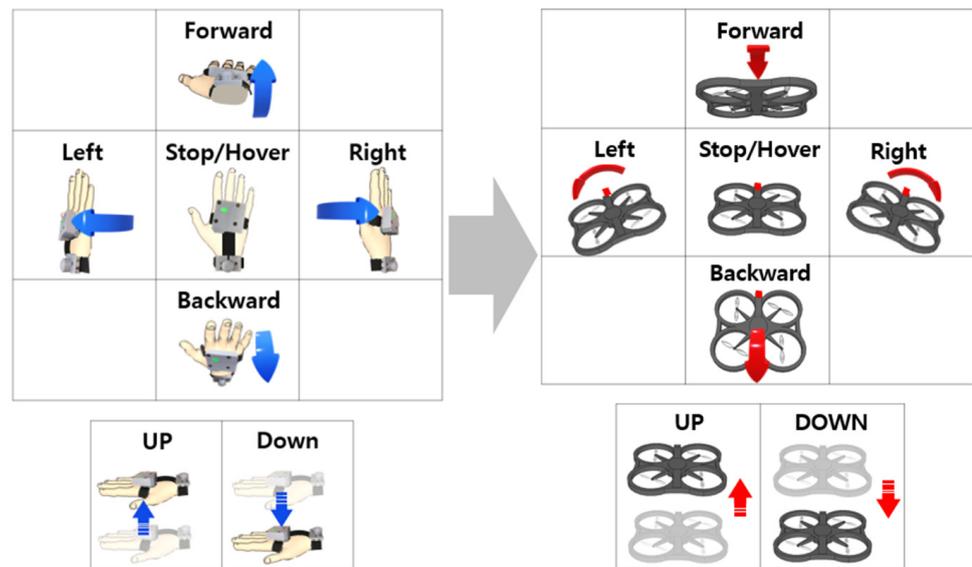


Figure 4. Definition of direct mode. It can directly control the drone’s posture and speed according to inclination and up/down of hand.

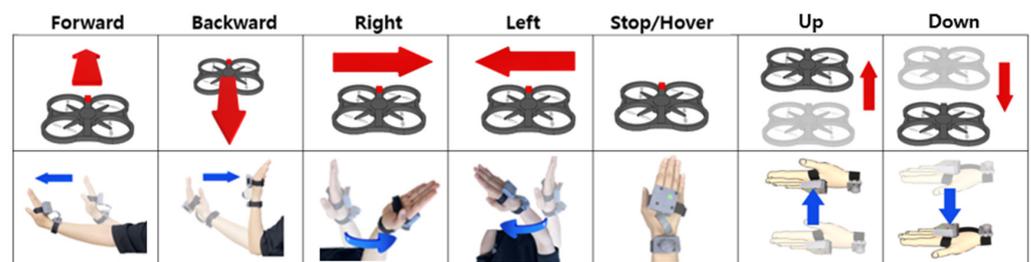


Figure 5. Definition of gesture mode. Gesture mode can control the direction of drone with natural hand motion.

3. Hand Gesture Recognition

The signal processing scheme in hand gesture mode is illustrated in Figure 6.

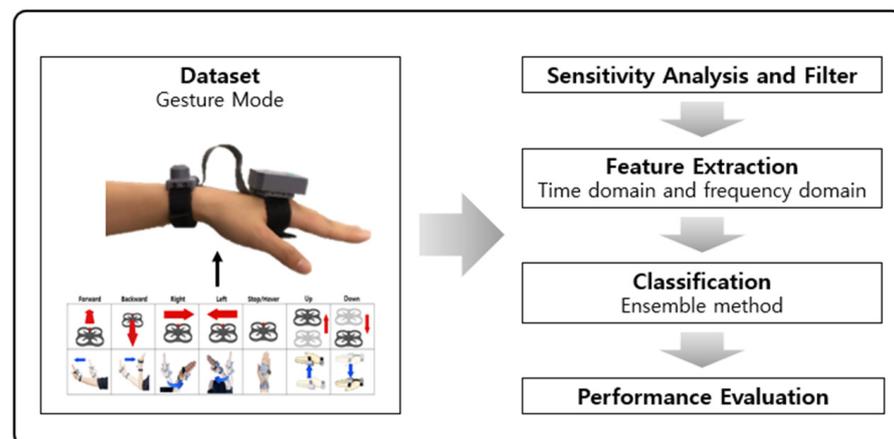


Figure 6. Scheme of signal processing for hand gesture recognition.

In Figure 6, the signals of the hand movements are obtained from the inertial sensor of the controller. Subsequently, the key factors of gestures were determined using sensitivity analysis to reduce the calculation time of signal processing. Based on the analysis, key signals were segmented using sliding windows and filtered using a low-pass filter to eliminate the gravity component of the accelerometer. The features were extracted from the

processed data, such as the mean, RMS, autocorrelation, power spectral density (PSD), and spectral peak. These features were input into the machine learning algorithm. Classification performance was evaluated based on accuracy, precision, recall, and the F-1 score.

3.1. Dataset

Large-scale motion data are required to classify hand gestures using a machine learning algorithm. Hence, the participants (male = 10, female = 2; right-hand dominant = 12; 26.6 ± 6 , 24–30 years old) were asked to perform the motions of gesture mode (forward, backward, right, left, stop, up, and down). The signal was recorded at 50 Hz to obtain quaternion, acceleration, and angular velocity along the three axes. To prevent imbalanced data, the number of each gesture data was 3300, and the overall number of data was 23,100.

3.2. Sensitivity Analysis and Preprocess

A sensitivity analysis was conducted with reduced processing time to ensure that key parameters dominantly influence hand gestures. The results of the sensitivity analysis are shown in Figure 7. Based on the results, the acceleration elements of x, y, z, and the angular velocity of y and z are commonly influenced as key parameters of hand gestures.

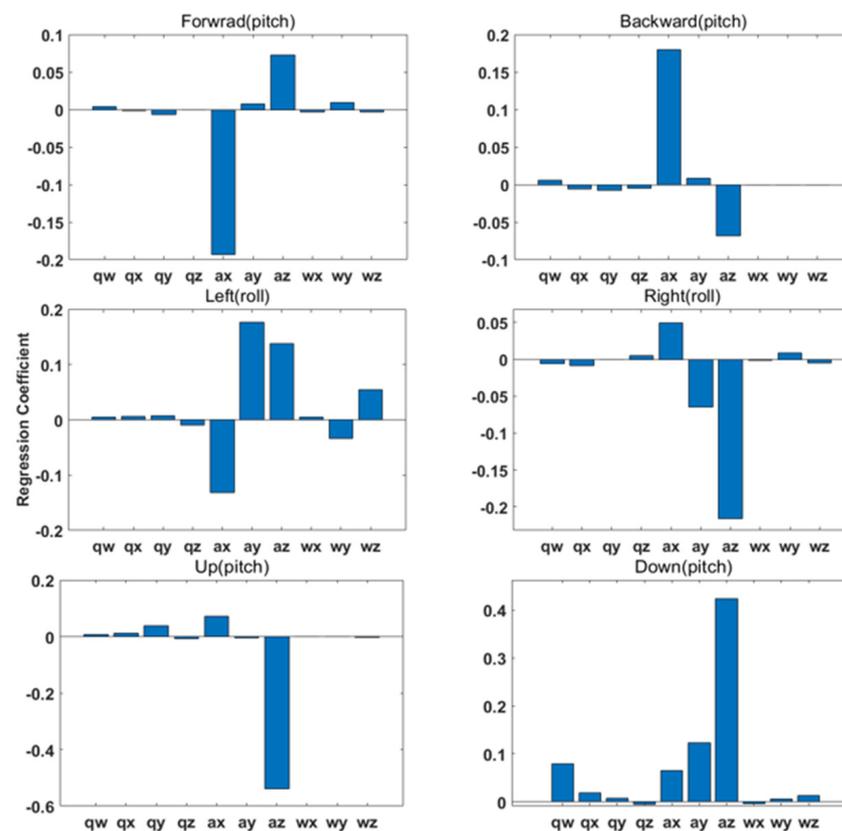


Figure 7. Result of sensitivity analysis on gesture mode movements. In particular, gestures such as forward, backward, up, and down were found to perform numerous pitch motions. Left and right gestures were known to perform roll actions primarily. According to the results, acceleration of x, y, z and angular velocity of y and z were determined to be key parameters in common.

Raw signals were also processed to remove the gravity component forced on the accelerometer using a low-pass filter. These signals were segmented with fixed sliding windows of 2.56 s and a 50% overlap (128 readings/window) [40–43].

3.3. Feature Extraction

Feature extraction is generally conducted to improve efficiency of the algorithm computation. The features are determined as time-domain and frequency-domain features [40–45].

We computed features such as the mean, RMS, autocorrelation, power spectral density (PSD), and spectral peak. The mean, RMS, and autocorrelation r_k are included in the time-domain features. The characteristics are as follows:

$$mean = \frac{1}{N} \sum_{i=1}^N a_i, \quad (1)$$

$$RMS = \sqrt{\frac{1}{N} (a_1^2 + a_2^2 + \dots + a_N^2)} \quad (2)$$

$$r_k = \frac{\sum_{i=k+1}^N (a_i - \bar{a})(a_{i-k} - \bar{a})}{\sum_{i=1}^N (a_i - \bar{a})^2} \quad (3)$$

where a_i denotes the components of the acceleration and angular velocity, and N denotes the window length. Equation 3 computes autocorrelation, which is the correlation between a_i and the delayed value a_{i+k} , where $k = 0, \dots, N$, \bar{a} denotes the mean of a_i . PSD and spectral peaks were included in the frequency-domain features. PSD was calculated as the squared sum of the spectral coefficients normalized by sliding window length. PSD is described as follows [44,45]:

$$PSD = \frac{1}{N} \sum_{i=0}^{N-1} x_i^2 + y_i^2 \quad (4)$$

with $x_i = z_i \cos\left(\frac{2\pi fi}{N}\right)$ and $y_i = z_i \sin\left(\frac{2\pi fi}{N}\right)$, where z denotes the discrete data for the frequency spectrum, and f represents the Fourier coefficient in the frequency domain. The spectral peak was calculated as the height and position of PSD [44,45].

3.4. Classification Model

To employ an appropriate classification algorithm, we compared the classification results of the ensemble, SVM, KNN, naive Bayes, and trees. The classifiers were used in MATLAB, and the training and test data were used in 9 to 1.

The performance of the learned model was evaluated based on the following expressions:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (5)$$

$$Precision = \frac{TP}{TP + FP} \quad (6)$$

$$Recall = \frac{TP}{TP + FN} \quad (7)$$

$$F1 - score = 2 * \frac{recall * precision}{recall + precision} \quad (8)$$

where TP is a true positive, TN is a true negative, FP is a false positive, and FN is a false negative. Figure 8a illustrates the comparison results of performance. Figure 8b shows the detailed results of each classifier's accuracy. According to Table 1, the ensemble method exhibited adequate performance in terms of classification accuracy.

Table 1. Comparison result of classifiers (%).

| | Ensemble | SVM | KNN | Naive Bayes | Trees |
|-----------|----------|------|------|-------------|-------|
| accuracy | 97.9 | 95.1 | 92.8 | 84.0 | 83.6 |
| precision | 97.9 | 95.1 | 93.0 | 84.1 | 83.8 |
| recall | 97.9 | 95.1 | 92.9 | 84.1 | 83.8 |
| F1-score | 97.9 | 95.1 | 92.9 | 84.1 | 83.8 |

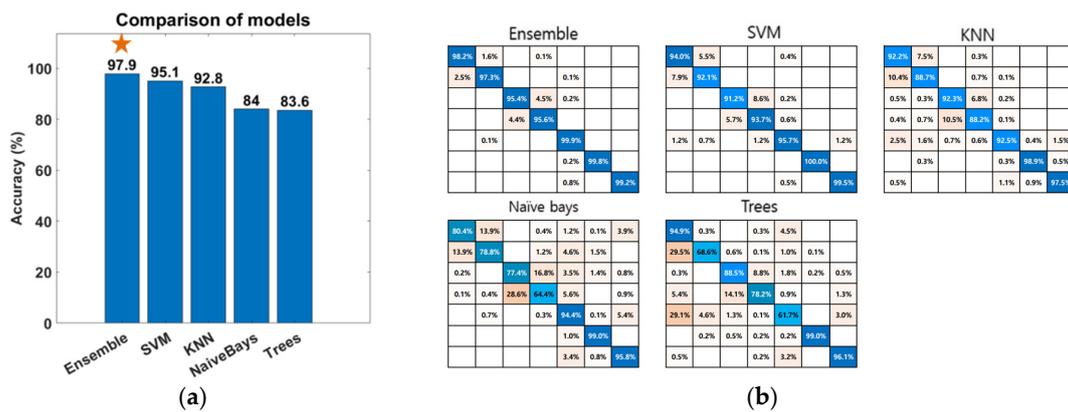


Figure 8. Result of gesture mode classification: (a) Comparison accuracy of each classification model. (b) Confusion matrix of ensemble, SVM, KNN, naive Bayes, and trees. From top left, the hand motions of forward, backward, right, left, stop (hover), up, and down are presented in order.

4. Simulation Experiment

4.1. Gesture-Based Control

4.1.1. Simulation Setup

To assess the effectiveness of the proposed gesture control, the participants (male = 10, female = 2; right-hand dominant = 12; 26.6 ± 6 , 24–30 years old) performed a virtual flight simulation with hand gestures while equipped with a wearable device. The participants did not have experience flying drones or were beginners. A schematic of the simulation mission is shown in Figure 9. The participants executed the appropriate hand gesture to maneuver the drone through the 12 waypoints described by the passing windows. These waypoints were placed according to the yellow guideline to pass, and one waypoint was required to adjust the flight altitude of the drone. The position of the drone was calculated at 50 Hz through quadrotor dynamics from MATLAB/Simulink. The simulation experiment comprised three preliminary sessions and the main experiment with a given mission. All participants were given sufficient time to complete a flight mission with direct and gesture modes.

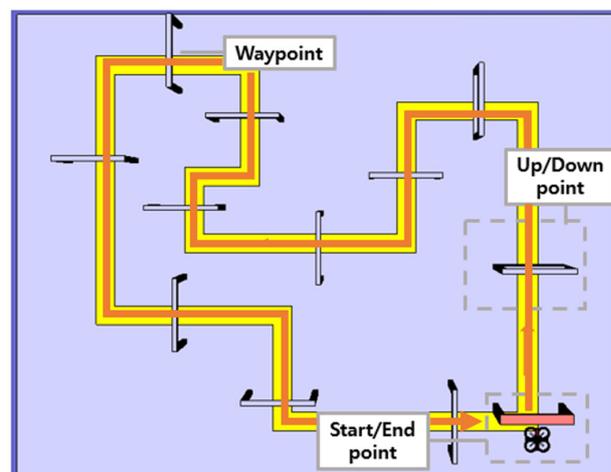


Figure 9. Mission schematic for drone flight simulation with gesture-based drone control.

Average trial duration, gesture accuracy, and number of gesture repetitions were calculated to evaluate the learning effect and adaptability of gesture-based drone control. Average trial duration, meaning the required average time for task completion, was used to evaluate the learning effect in the subjects. The average gesture accuracy was used to evaluate the performance of the classification of hand gestures. The average number of gesture repetitions, counting the frequency of motion changes, was used to evaluate adaptability.

Participants were also asked to respond to a questionnaire regarding convenience of the control operation and physical discomfort after the experiment. Questions were answered in the form of a 7-point Likert scale.

4.1.2. Results

The results of the first and last trials were analyzed to investigate the controller's efficiency and the learning effect. Figures 10 and 11 indicate the drone trajectory and the frequency of motion changes during the first and last attempts using direct mode control. Compared with the first trial, the trajectory stabilized in the last trial. In addition, the frequency of motion changes decreased because of the adaptation of direct mode control. Average accuracy also increased as the experiment progressed, and there was a larger reduction in trial duration and repetition, as shown in Figure 12.

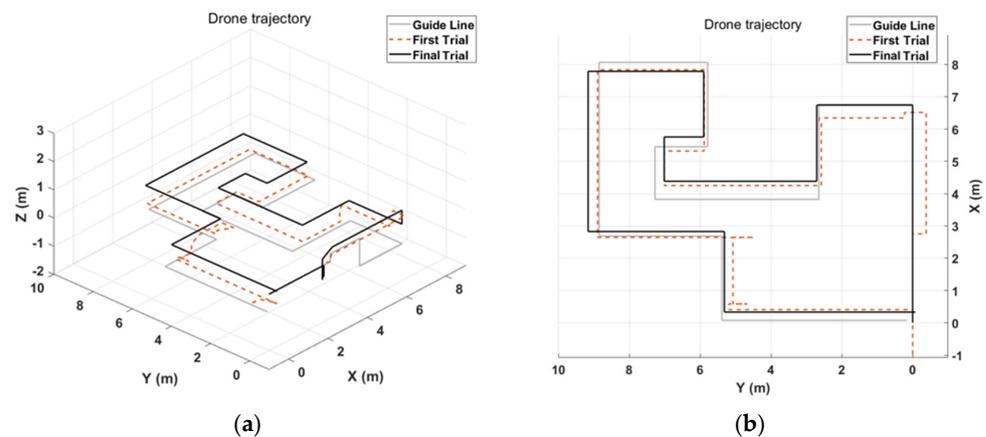


Figure 10. Example of drone flight trajectory with direct mode. Gray line is guideline, dotted line is trajectory of first trial, and black line is trajectory of last trial. Comparing the trajectory between first and last trials, the last one is a more stable shape. (a) Isometric view of drone trajectory can be found by adjusting drone altitude. (b) Top view of drone trajectory can be determined by behavior of drone according to the user on the x-y dimension.

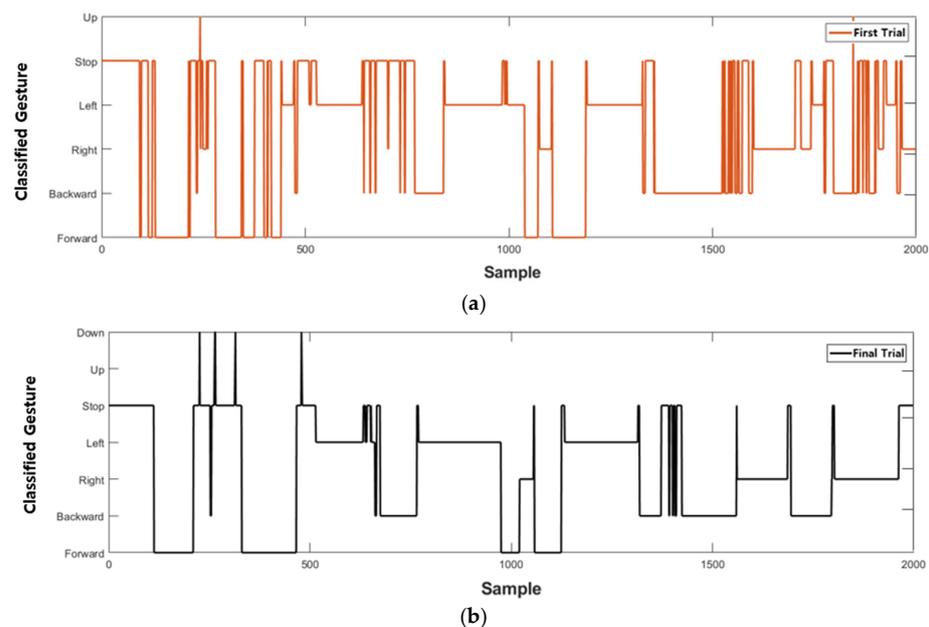
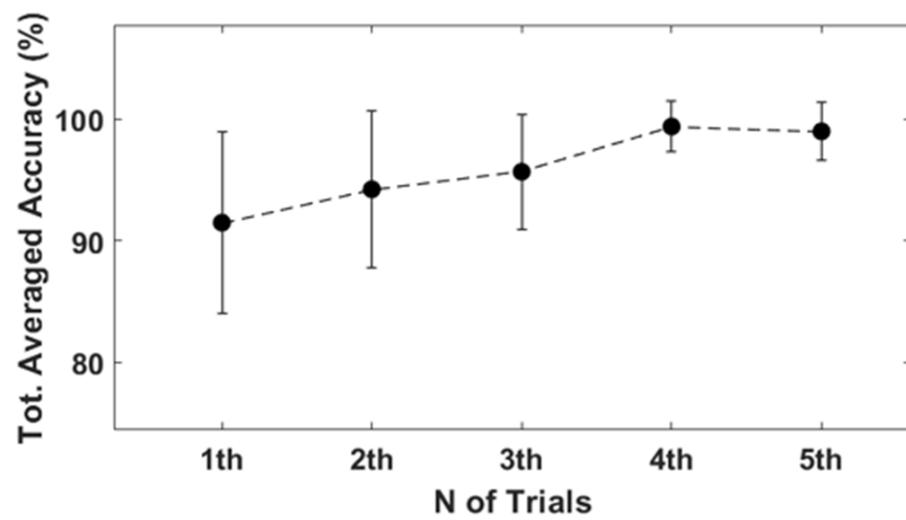
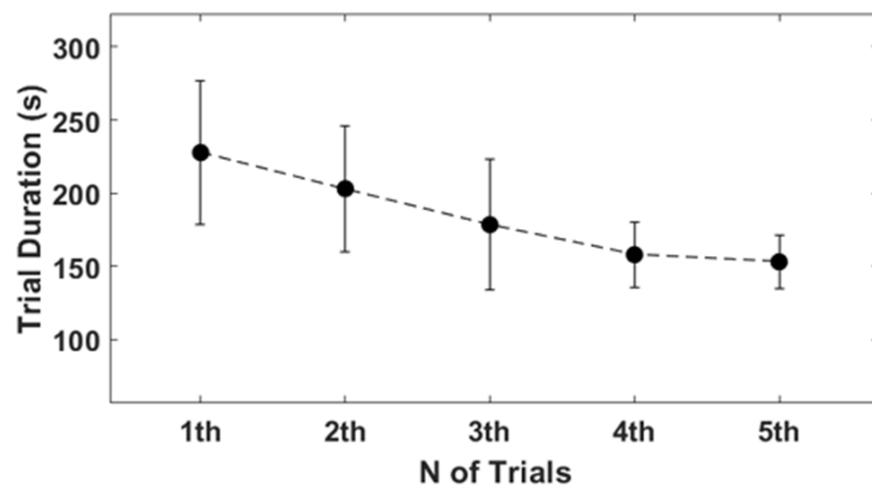


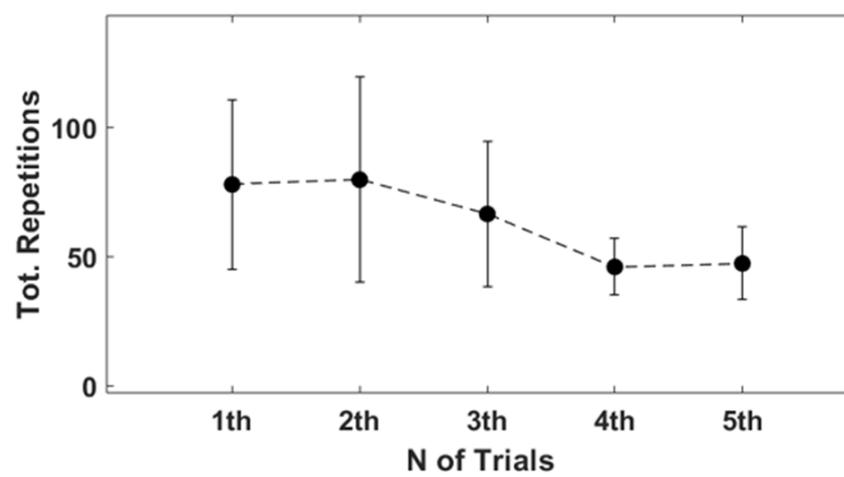
Figure 11. Frequency of gesture changes during trial duration with direct mode. (a) Classified gesture changes at first trial. (b) Classified gesture changes at last trial, which shows less change than first trial.



(a)



(b)



(c)

Figure 12. Results of direct mode simulation (mean \pm SD). (a) Total average accuracy of each trial. First trial is $91.5 \pm 7.4\%$, and last trial is $99 \pm 2.4\%$. (b) Average trial duration. First trial is 227.7 ± 48.7 s, and last trial is 153.0 ± 18.1 s. (c) Total repetitions of gestures. First trial is 77.8 ± 32.8 , and last trial is 47.2 ± 14.0 .

Similarly, Figures 13 and 14 show drone trajectory and frequency of motion changes in the first and last trials with gesture mode control. In the first trial, the drone trajectory was unstable, especially when the user executed the gesture in the left direction. The frequency of motion changes also decreased compared with the first trial. Across all trials, average accuracy tended to increase, as did a larger reduction in trial duration and repetition, as shown in Figure 15. These results indicate that gesture-based control has learning effects and adaptability.

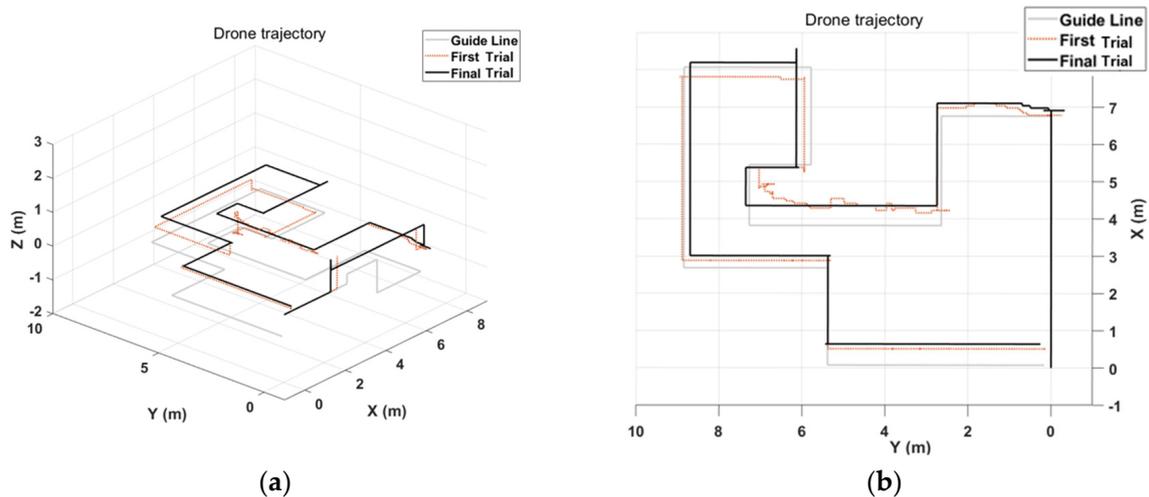


Figure 13. Example of drone flight trajectory with gesture mode. Gray line is guideline, dotted line represents drone trajectory of first trial, and black line is last trial. (a) Isometric view can be found altitude control using gestures. (b) Top view of drone trajectory can be figured out drone movements on x-y dimension. While first trial is low-accuracy and the trajectory is unstable, last trial is high-accuracy and confirms that the trajectory appeared similar to the guideline.

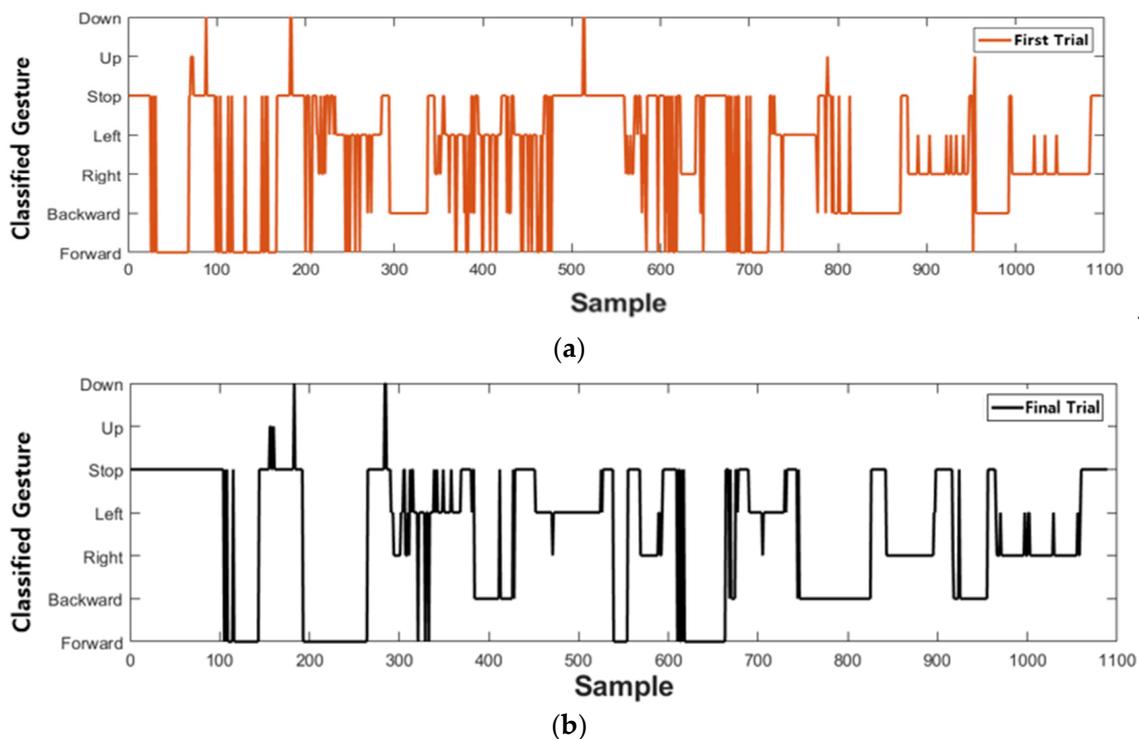
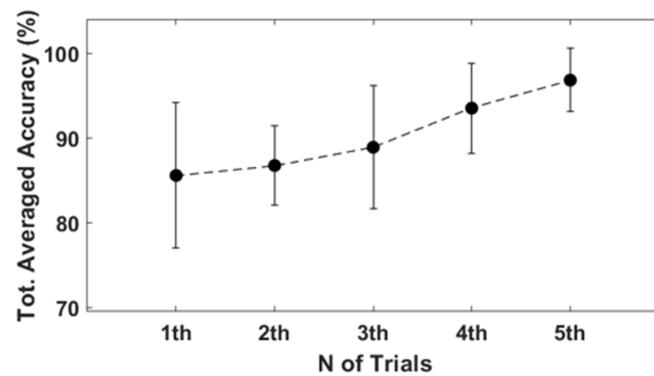
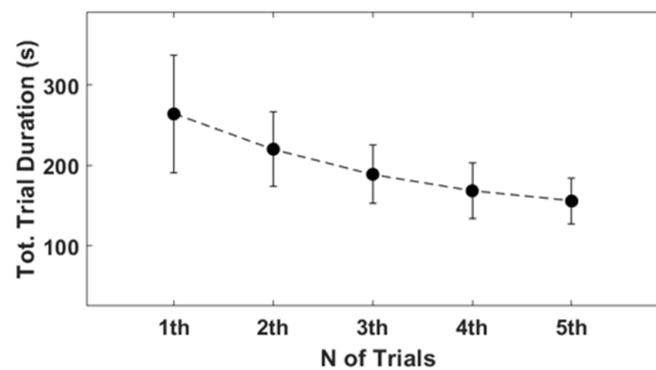


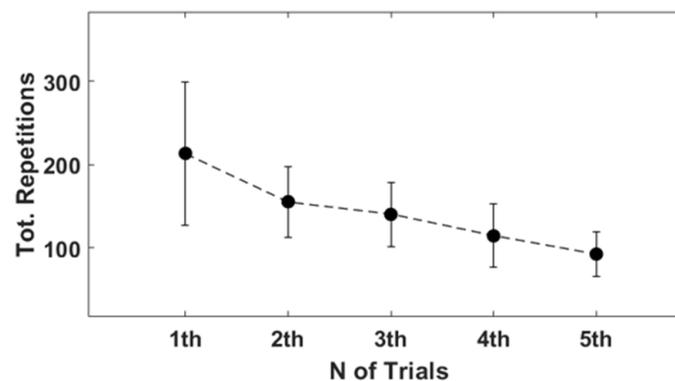
Figure 14. Frequency of gesture changes during trial duration with gesture mode. (a) Classified gesture changes at first mission. (b) Classified gesture changes at last trial, which shows less change than first trial.



(a)



(b)



(c)

Figure 15. Results of gesture mode simulation (mean \pm SD). (a) Average accuracy of first trial was $85.9 \pm 8.4\%$, and final trial was $96.8 \pm 3.9\%$. (b) First trial duration was 264 ± 73.3 s, and last duration was 155.7 ± 28.9 s, which shows decreased duration. (c) Difference of total repetitions between first and last trial is 138.9 ± 59.1 .

4.2. Vibrotactile Feedback

4.2.1. Simulation Setup

An obstacle avoidance simulation experiment was conducted to demonstrate the performance of vibrotactile feedback. The participants, who were the same as mentioned above, were asked to avoid obstacles distributed in front of the drone using gesture control. All the participants performed the simulation experiments three times with a fixed field of view, as shown in Figure 16a. In the first trial, participants controlled the drone without vibrotactile conditions. In the other trials, the drone was controlled using vibrotactile feedback. To assess the performance of the vibrotactile stimulation, the evaluator measured the collision status through the top view of the flight environment, as shown in Figure 16b.

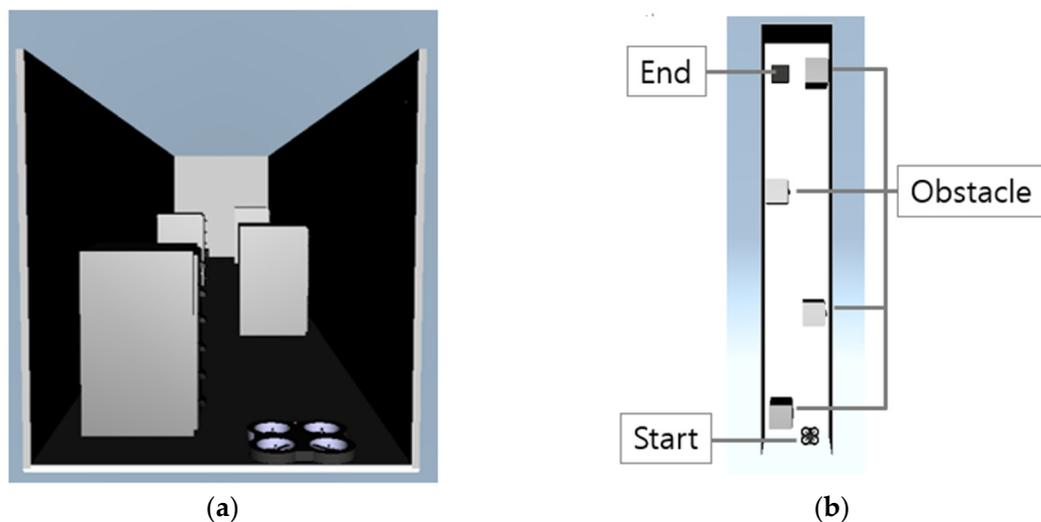


Figure 16. Mission schematic for vibrotactile feedback simulation experiment. (a) Fixed view from the participants who were asked to control the drone flight avoiding obstacles. (b) Top view of drone flight environment to evaluate the control performance with vibrotactile feedback.

4.2.2. Result

Without vibrotactile feedback, the participants could not manipulate the drone appropriately because of a lack of visual feedback, as shown in Figure 16a. The drone successfully avoided all obstacles with vibrotactile feedback, as illustrated in Figure 17.

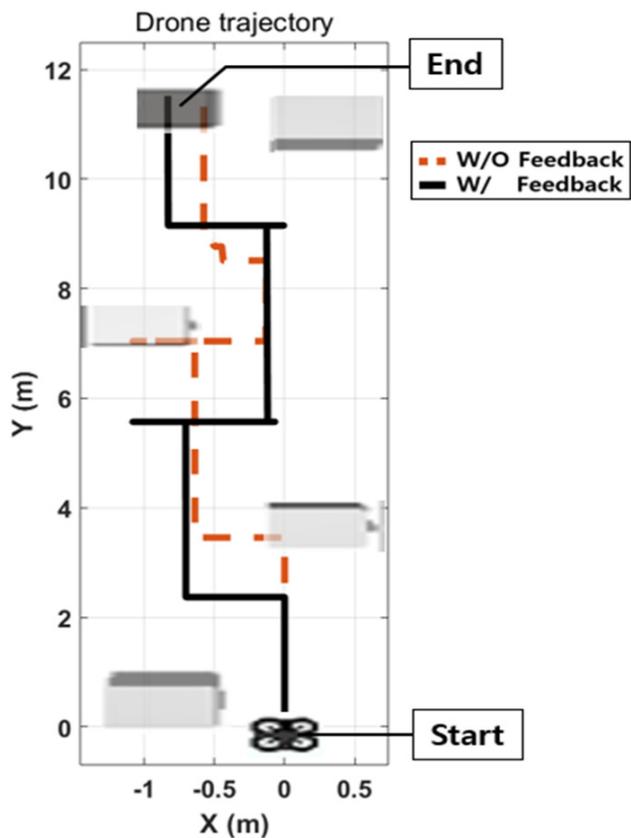


Figure 17. Example of drone flight trajectories with and without vibrotactile conditions. When the vibrotactile feedback did not work, the drone crashed into boxes because of lack of visual feedback (dashed line). With vibrotactile feedback, the drone achieved the mission of avoiding obstacles (black line).

The results of the vibrotactile feedback simulation are presented in Table 2. All the participants experienced crashes with obstacles at least once. When the vibrotactile stimulator was not used, the participants did not recognize the obstacles well with limited visibility. Therefore, the drone crashed an average of 2.2 times. However, with vibrotactile feedback, each participant detected obstacles via vibrotactile stimulation in the wrist and effectively commanded the drone to avoid obstacles.

Table 2. Results of obstacle avoidance using vibrotactile feedback. Without vibrotactile feedback, the drone failed to avoid obstacles; meanwhile, collision avoidance was completed successfully with vibrotactile feedback.

| Participant No. | First Trial (without Feedback) | Second Trial (with Feedback) | Third Trial (with Feedback) |
|-----------------|-----------------------------------|---------------------------------|--------------------------------|
| Participant 1 | 2/4 * | 4/4 | 4/4 |
| Participant 2 | 1/4 | 4/4 | 4/4 |
| Participant 3 | 1/4 | 4/4 | 4/4 |
| Participant 4 | 2/4 | 4/4 | 4/4 |
| Participant 5 | 1/4 | 4/4 | 4/4 |
| Participant 6 | 2/4 | 4/4 | 4/4 |
| Participant 7 | 2/4 | 4/4 | 4/4 |
| Participant 8 | 1/4 | 4/4 | 4/4 |
| Participant 9 | 2/4 | 4/4 | 4/4 |
| Participant 10 | 2/4 | 4/4 | 4/4 |
| Participant 11 | 2/4 | 4/4 | 4/4 |
| Participant 12 | 2/4 | 4/4 | 4/4 |

* Successful/avoidance trials.

4.3. Subjective Evaluation of Participants

Subjective evaluations were conducted to assess the user-friendliness and effectiveness of the controller after the experiment. Participants were asked to respond to the experience of use based on a 7-point Likert scale. The participants gave assessment points for the user-friendliness and effectiveness of the direct and gesture modes. Similarly, participants were asked to respond to the control with and without vibrotactile feedback. Table 3 shows the subjective evaluation of gesture-based drone control. The participants agreed to the convenience and effectiveness of the controller by more than 6 points and did not agree on discomfort and fatigue by less than 3 points. There are no significant differences between the direct and gesture modes, confirming that the proposed controller is easy to use and effective.

Table 3. Results of subjective evaluation by participants in gesture-based drone control.

| Question | Direct Mode | Gesture Mode |
|---|-------------|--------------|
| 1. The proposed gesture was natural for me. | 6.4 ± 0.6 | 6.0 ± 0.7 |
| 2. I felt physical discomfort while controlling. | 1.6 ± 0.6 | 2.0 ± 0.9 |
| 3. My hand and arm were tired while controlling. | 2.0 ± 0.6 | 2.4 ± 0.8 |
| 4. The proposed gesture was user-friendly. | 6.5 ± 0.9 | 6.3 ± 1.4 |
| 5. I felt the convenience of controlling a drone with one hand. | 6.5 ± 0.6 | 6.6 ± 0.5 |
| 6. It was interesting to fly a drone with a gesture. | 6.5 ± 1.0 | 6.9 ± 0.3 |

Table 4 shows the subjective evaluations of the vibrotactile feedback. All the participants responded that there were substantial differences between the control with vibrotactile feedback and the control relying on only limited visual information. They expected that vibrotactile feedback would be helpful for drone control in the field.

Table 4. Results of subjective evaluation by participants in vibrotactile feedback.

| Question | Mean \pm SD |
|--|---------------|
| 1. The vibrotactile feedback was helpful for obstacle avoidance. | 6.9 \pm 0.3 |
| 2. The vibration intensity was appropriate. | 6.6 \pm 0.6 |
| 3. My hand and wrist were tired by vibrotactile feedback. | 1.4 \pm 0.8 |
| 4. The obstacle avoidance was difficult without the vibrotactile feedback condition. | 6.5 \pm 0.7 |
| 5. If I flew a drone in real life, vibrotactile feedback would be helpful. | 6.5 \pm 0.3 |

5. Experimental Validation

5.1. Gesture-Based Drone Control

5.1.1. Setup

The implemented controller was tested to validate its performance. The flight test was conducted using a quadrotor (AR.Drone 2.0, Parrot, Paris, France). In the mission scenario, the user flew the drone through four gates using gesture control, as shown in Figure 18. The drone cruised at 0.4 m/s and hovered at a height of about 0.8 m. The user was positioned within gray dotted lines (Figure 18a) to ensure visibility for drone control. Three gates were placed on the guideline, and one was installed on the table to control up/down movements. An appropriate gesture was assigned for each trajectory, and the user's gesture was recognized and compared with the assigned gesture. To evaluate the performance, gesture accuracy was evaluated using drone states, classified gestures, and aerial videos during the experiment.

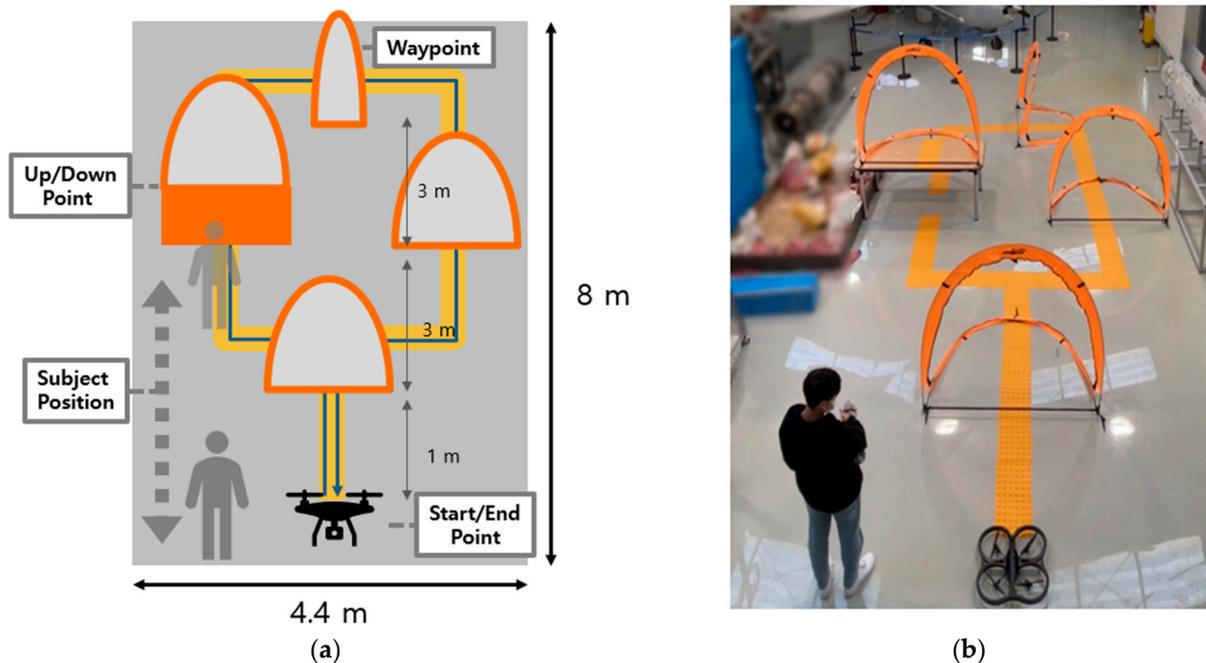


Figure 18. Experiment scenario with gesture-based drone control. (a) Schematic diagram for experiment. The yellow guideline was installed for reference of drone trajectory. The user walked within gray dotted lines to secure the view. (b) Experiment environment based on schematic was set indoors for safe operation.

5.1.2. Result

The entire recognition rate in the direct mode is shown in Figure 19a. The average accuracy of the direct mode was 96.4%, and the mission duration was 103 s. The classification rate in the gesture mode condition is shown in Figure 19b. The average accuracy of the gesture mode was 98.2%, and the mission duration was 119 s.

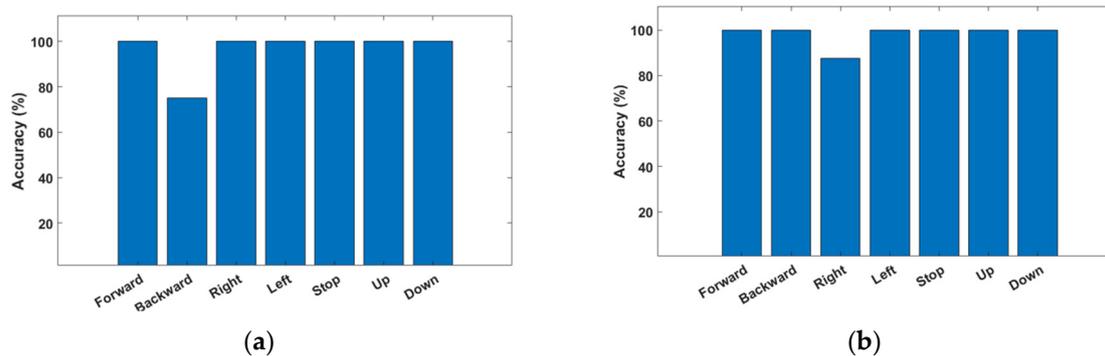


Figure 19. Accuracy of real drone control. (a) Average accuracy of direct mode is about 96%, which is about 2.6% lower than the simulation. (b) Average accuracy of gesture mode is about 98%, which is about 1.4% higher than the simulation.

5.2. Vibrotactile Feedback

5.2.1. Setup

Vibrotactile feedback was tested for efficiency in a real environment, as shown in Figure 20. The user controls the drone using gesture mode through gates to avoid obstacles. To demonstrate the effectiveness of vibrotactile feedback, the user's visibility was limited by the blinding of the panel, which blocked part of the obstacle and the gate to pass. The first trial was performed without a vibrotactile condition. The second trial was conducted with vibrotactile feedback, representing the distance between the drone and the obstacle.

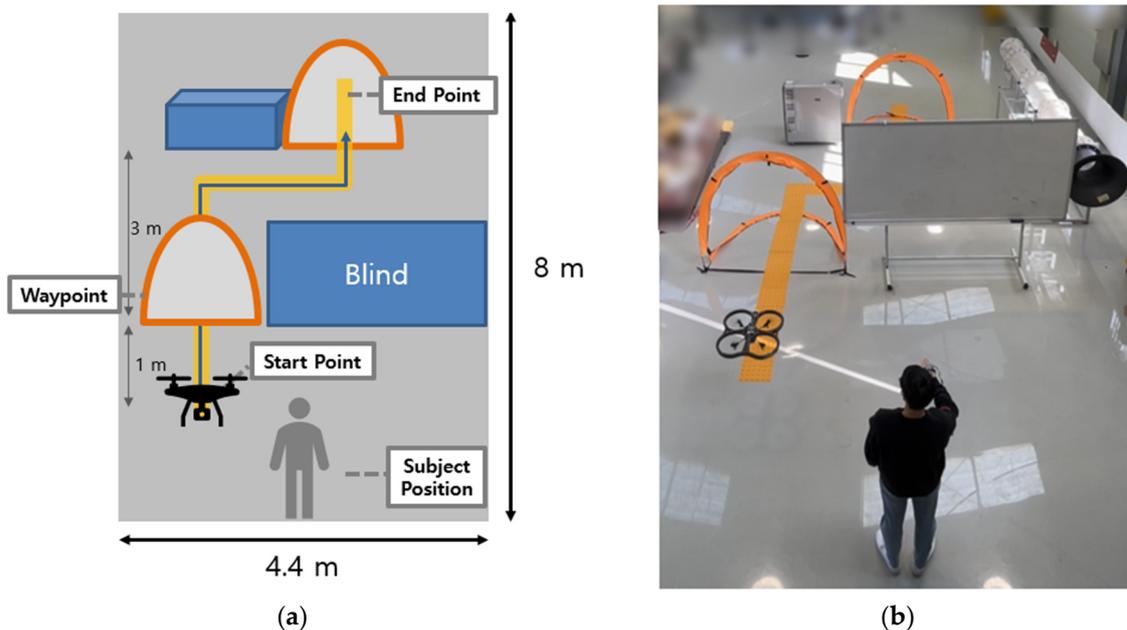


Figure 20. Flight experiment for vibrotactile feedback with gesture mode. (a) Schematic diagram comprising visual obstruction, drone, passing gates, obstacle, and guideline. (b) Based on the scheme, flight experiment was set up.

5.2.2. Result

The drone failed to avoid collisions without vibrotactile feedback, as shown in Figure 21a. The user cannot estimate the distance between the drone and the obstacle owing to limited visual information. However, collision avoidance was achieved successfully with vibrotactile feedback, as shown in Figure 21b. In addition, the distance to the obstacle in front of the drone was inferred through the intensity of the vibrotac-

tile stimulation, and the mission was carried out by maintaining a certain distance from the obstacle.

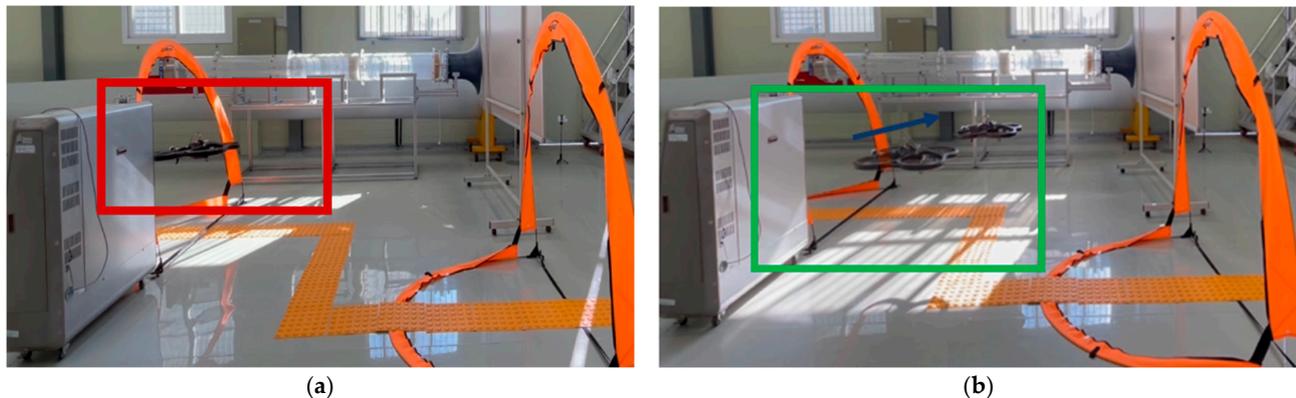


Figure 21. Result of vibrotactile feedback. (a) Drone could not avoid the collision in front. Owing to the lack of visual information for the obstacle, the mission failed. (b) Drone flew successfully through gates, and distance to the obstacle was maintained using vibrotactile feedback.

5.3. Discussions

The presented drone controller exhibited user-friendly and intuitive features, which can be used more easily by an inexperienced user for drone maneuvering. Through simulation experiments on drones, an effective interface for drone control was revealed by estimating control accuracy and mission duration. An experiment on a real drone was carried out to validate the effectiveness of the proposed controller. Indeed, vibrotactile feedback was helpful in detecting obstacles compared to using only visual information. The participants also felt more comfortable with and interested in the proposed controller.

In a further study, the addition of yawing control of the drone was considered to implement a supplementary sensor to measure the hand motion's orientation accurately. Furthermore, implementing additional ultrasonic sensors on a drone enables the omnidirectional distance measurement of obstacles. The aforementioned approaches are expected to complete a wearable drone control system with a natural user interface.

6. Conclusions

This study proposed a wearable drone controller with hand gesture recognition and vibrotactile feedback. The hand motions for drone control were sensed by an IMU attached to the back of the hand. The measured motions were classified based on machine learning using the ensemble method with a classification accuracy of 97.9%. Furthermore, the distance to obstacles in the heading direction of the drone was fed back to the user stimulating the vibration motor attached to the wrist, known as vibrotactile feedback. In the simulation experiment by the participant group, the hand gesture control showed good performance, and the vibrotactile feedback helped the user be aware of the operation environment of the drone, especially when limited visual information was available. A subjective evaluation of the participants was performed to assess the convenience and effectiveness of the proposed controller. Finally, an experiment with a real drone was conducted, confirming that the proposed controller could be applicable for drone operation as a natural interface.

Author Contributions: Conceptualization and methodology, K.-H.Y. and J.-W.L.; simulation and experiment, J.-W.L.; analysis and validation, K.-H.Y. and J.-W.L.; writing, K.-H.Y. and J.-W.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research has been supported by the National Research Foundation of Korea (NRF) under Grant 2021R1A2C1009127.

Acknowledgments: The authors would like to thank Byeong-Seop Sim for his technical support in fabricating the device, Kun-Jung Kim for his technical advice about the machine learning approach, and Min-Chang Kim for his support in performing the experiment.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bouabdallah, S.; Becker, M.; Siegwart, R. Autonomous miniature flying robots: Coming soon!—Research, Development, and Results. *IEEE Robot. Autom. Mag.* **2007**, *14*, 88–98. [[CrossRef](#)]
2. Shakhatreh, H.; Sawalmeh, A.H.; Al-fuqaha, A.; Dou, Z.; Almaita, E.; Khalil, I.; Othman, N.S.; Khreishah, A.; Guizani, A.M. Unmanned Aerial Vehicles (UAVs): A Survey on Civil Applications and Key Research Challenges. *IEEE Access* **2019**, *7*, 48572–48634. [[CrossRef](#)]
3. Zeng, Y.; Zhang, R.; Lim, T.J. Wireless Communications with Unmanned Aerial Vehicles: Opportunities and Challenges. *IEEE Commun. Mag.* **2016**, *54*, 36–42. [[CrossRef](#)]
4. Chen, J.Y.C.; Barnes, M.J.; Harper-Sciarni, M. Supervisory Control of Multiple Robots: Human-Performance Issues and User-Interface Design. *IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.)* **2011**, *41*, 435–454. [[CrossRef](#)]
5. Zhou, J.; Zhi, H.; Kim, M.; Cummings, M.L. The Impact of Different Levels of Autonomy and Training on Operators' Drone Control Strategies. *ACM Trans. Hum. -Robot Interact.* **2019**, *8*, 22. [[CrossRef](#)]
6. Smolyanskiy, N.; Gonzalez-Franco, M. Stereoscopic First Person View System for Drone Navigation. *Front. Robot. AI* **2017**, *4*, 11. [[CrossRef](#)]
7. Fong, T.; Conti, F.; Grange, S.; Baur, C. Novel interfaces for remote driving: Gestures, haptic and PDA. In Proceedings of the Society of Photo-Optical Instrumentation Engineers, San Diego, CA, USA, 29 July–3 August 2001; Volume 4195, pp. 300–311.
8. Fernandez, R.A.S.; Sanchez-Lopez, J.S.; Sampedro, C.; Balve, H.; Molina, M.; Campoy, P. Natural User Interface for Human-Drone Multi-Modal Interaction. In Proceedings of the 2016 International Conference on Unmanned Aircraft Systems, Arlington, VA, USA, 7–10 June 2016; pp. 1013–1022.
9. Tezza, D.; Andujar, M. The State-of-the Art of Human-Drone Interaction: A Survey. *IEEE Access* **2019**, *7*, 167438–167454. [[CrossRef](#)]
10. Hakim, N.L.; Shih, T.K.; Arachchi, S.P.K.; Aditya, W.; Chen, Y.-C.; Lin, C.-Y. Dynamic Hand Gesture Recognition Using 3DCNN and LSTM with FSM Context-Aware Model. *Sensors* **2019**, *19*, 5429. [[CrossRef](#)]
11. D'Eusanio, A.; Simoni, A.; Pini, S.; Borghi, G.; Vezzani, R.; Cucchiara, R. Multimodal Hand Gesture Classification for the Human-Car Interaction. *Informatics* **2020**, *7*, 31. [[CrossRef](#)]
12. Choi, J.-W.; Ryu, S.-J.; Kim, J.-W. Short-Range Radar Based Real-Time Hand Gesture Recognition Using LSTM Encoder. *IEEE Access* **2019**, *7*, 33610–33618. [[CrossRef](#)]
13. Han, H.; Yoon, S.W. Gyroscope-Based Continuous Human Hand Gesture Recognition for Multi-Modal Wearable Input Device for Human Machine Interaction. *Sensors* **2019**, *19*, 2562. [[CrossRef](#)] [[PubMed](#)]
14. Cifuentes, J.; Boulanger, P.; Pham, M.T.; Prieto, F.; Moreau, R. Gesture Classification Using LSTM Recurrent Neural Networks. In Proceedings of the 2019 International Conference on IEEE Engineering in Medicine and Biology Society, Berlin, Germany, 23–27 July 2019; pp. 6864–6867.
15. Zhao, H.; Ma, Y.; Wang, S.; Watson, A.; Zhou, G. MobiGesture: Mobility-aware hand gesture recognition for healthcare. *Smart Health* **2018**, *9–10*, 129–143. [[CrossRef](#)]
16. Liu, C.; Sziranyi, T. Real-Time Human Detection and Gesture Recognition for On-Board UAV Rescue. *Sensors* **2021**, *21*, 2180. [[CrossRef](#)]
17. Begum, T.; Haque, I.; Keselj, V. Deep Learning Models for Gesture-controlled Drone Operation. In Proceedings of the 2020 International Conference on Network and Service Management, Izmir, Turkey, 2–6 November 2020.
18. Hu, B.; Wang, J. Deep Learning Based Hand Gesture Recognition and UAV Flight Controls. *Int. J. Autom. Comput.* **2020**, *17*, 17–29. [[CrossRef](#)]
19. Shin, S.-Y.; Kang, Y.-W.; Kim, Y.-G. Hand Gesture-based Wearable Human-Drone Interface for Intuitive Movement Control. In Proceedings of the 2019 IEEE International Conference on Consumer Electronics, Las Vegas, NV, USA, 11–13 January 2019.
20. Burdea, G. *Force and Touch Feedback for Virtual Reality*; John Wiley & Sons, Inc.: New York, NY, USA, 2000.
21. Bach-y-Rita, P.; Kercel, S.W. Sensory substitution and the human-machine interface. *Trends Cogn. Sci.* **2003**, *7*, 541–546. [[CrossRef](#)] [[PubMed](#)]
22. Erp, J.B.F.V. Guidelines for the use of vibro-tactile displays in human computer interaction. In Proceedings of the Eurohaptics Conference, Edinburgh, UK, 8–10 July 2002; pp. 18–22.
23. Dakopoulos, D.; Bourbakis, N.G. Wearable Obstacle Avoidance Electronic Travel Aids for Blind: A Survey. *IEEE Trans. Syst. Man Cybern. C Appl. Rev.* **2010**, *40*, 25–35. [[CrossRef](#)]
24. Kim, D.; Kim, K.; Lee, S. Stereo camera based virtual cane system with identifiable distance tactile feedback for the blind. *Sensors* **2014**, *14*, 10412–10431. [[CrossRef](#)]
25. Tsykunov, E.; Labazanova, L.; Tleugazy, A.; Tsetserukou, D. SwamTouch: Tactile Interaction of Human with Impedance Controlled Swarm of Nano-Quadrotors. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems, Madrid, Spain, 1–5 October 2018.

26. Rognon, C.; Ramachandran, V.; Wu, A.R.; Ljspeert, A.J.; Floreano, D. Haptic Feedback Perception and Learning with Cable-Driven Guidance in Exosuit Teleoperation of a Simulated Drone. *IEEE Trans. Haptics* **2019**, *12*, 375–385. [[CrossRef](#)]
27. Duan, T.; Punpongsonan, P.; Iwaki, D.; Sato, K. FlyingHand: Extending the range of haptic feedback on virtual hand using drone-based object recognition. In Proceedings of the SIGGRAPH Asia 2018 Technical Briefs, Tokyo, Japan, 4–7 December 2018.
28. Lee, J.W.; Kim, K.J.; Yu, K.H. Implementation of a User-Friendly Drone Control Interface Using Hand Gestures and Vibrotactile Feedback. *J. Inst. Control Robot. Syst.* **2022**, *28*, 349–352. [[CrossRef](#)]
29. Lim, S.C.; Kim, S.C.; Kyung, K.U.; Kwon, D.S. Quantitative analysis of vibrotactile threshold and the effect of vibration frequency difference on tactile perception. In Proceedings of the SICE-ICASE International Joint Conference, Busan, Korea, 18–21 October 2006; pp. 1927–1932.
30. Yoon, M.J.; Yu, K.H. Psychophysical experiment of vibrotactile pattern perception by human fingertip. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2008**, *16*, 171–177. [[CrossRef](#)]
31. Jeong, G.-Y.; Yu, K.-H. Multi-Section Sensing and Vibrotactile Perception for Walking Guide of Visually Impaired Person. *Sensors* **2016**, *16*, 1070. [[CrossRef](#)] [[PubMed](#)]
32. Ahmed, M.A.; Zaidan, B.B.; Zaidan, A.A.; Salih, M.M.; Bin Lakulu, M.M. A Review on System-Based Sensory Gloves for Sign Language Recognition State of the Art between 2007 and 2017. *Sensors* **2018**, *18*, 2208. [[CrossRef](#)] [[PubMed](#)]
33. Avola, D.; Bernardi, M.; Cinque, L.; Foresti, G.L.; Massaroni, C. Exploiting Recurrent Neural Networks and Leap Motion Controller for the Recognition of Sign Language and Semaphoric Hand Gesture. *IEEE Trans. Multimed.* **2019**, *21*, 234–245. [[CrossRef](#)]
34. Nagi, J.; Giusti, A.; Di Caro, G.A.; Gambardella, L.M. Human Control of UAVs using Face Pose Estimates and Hand Gestures. In Proceedings of the 2014 ACM/IEEE International Conference on Human-Robot Interaction, Bielefeld, Germany, 3–6 March 2014.
35. Vadakkepat, P.; Chong, T.C.; Arokiasami, W.A.; Xu, W.N. Fuzzy Logic Controllers for Navigation and Control of AR. Drone using Microsoft Kinect. In Proceedings of the 2016 IEEE International Conference on Fuzzy Systems, Vancouver, BC, Canada, 24–29 July 2016.
36. DelPreto, J.; Rus, D. Plug-and-Play Gesture Control Using Muscle and Motion Sensors. In Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction, Cambridge, UK, 23–26 March 2020; pp. 439–448.
37. Gromov, B.; Guzzi, J.; Gambardella, L.M.; Alessandro, G. Intuitive 3D Control of a Quadrotor in User Proximity with Pointing Gestures. In Proceedings of the International Conference on Robotics and Automation, Paris, France, 31 May–31 August 2020; pp. 5964–5971.
38. *Visual Signals*; Army Publishing Directorate: Washington, DC, USA, 2017.
39. *Visual Aids Handbook*; Civil Aviation Authority: West Sussex, UK, 1997.
40. Anguita, D.; Ghio, A.; Oneto, L.; Parra, X.; Reyes-Ortiz, J.L. Human Activity Recognition on Smartphones Using a Multiclass Hardware-Friendly Support Vector Machine. In *Ambient Assisted Living and Home Care*; Bravo, J., Hervás, R., Rodríguez, M., Eds.; IWAAL 2012; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2012; Volume 7657.
41. Anguita, D.; Ghio, A.; Oneto, L.; Parra, X.; Reyes-Ortiz, J.L. A Public Domain Dataset for Human Activity Recognition Using Smartphones. In Proceedings of the European Symposium on Artificial Neural Networks, Bruges, Belgium, 24–26 April 2013; pp. 437–442.
42. Helou, A.E. Sensor HAR Recognition App. 2015. Available online: <https://www.mathworks.com/matlabcentral/fileexchange/54138-sensor-har-recognition-app> (accessed on 20 November 2021).
43. Helou, A.E. Sensor Data Analytics. 2015. Available online: <https://www.mathworks.com/matlabcentral/fileexchange/54139-sensor-data-analytics-french-webinar-code> (accessed on 20 November 2021).
44. Attal, F.; Mohammed, S.; Dedabrishvili, M.; Chamroukhi, F.; Oukhellou, L.; Amirat, Y. Physical Human Activity Recognition Using Wearable Sensors. *Sensors* **2015**, *15*, 31314–31338. [[CrossRef](#)] [[PubMed](#)]
45. Dehjangi, O.; Sahu, V. IMU-Based Robust Human Activity Recognition using Feature Analysis, Extraction, and Reduction. In Proceedings of the 2018 International Conference on Pattern Recognition, Beijing, China, 20–24 August 2018.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.