drones

MDPI

Article

Vision-Based Gesture-Driven Drone Control in a Metaverse-Inspired 3D Simulation Environment

Yaseen [1], Oh-Jin Kwon [1,*], Jaeho Kim [2], Jinhee Lee [1] and Faiz Ullah [1]

[1] Department of Electronics Engineering, Sejong University, Seoul 05006, Republic of Korea; yaseen@sju.ac.kr (Y.); jinee5025@sju.ac.kr (J.L.); faiz@sju.ac.kr (F.U.)
[2] Department of Electrical Engineering, Sejong University, Seoul 05006, Republic of Korea; kimjh@sejong.ac.kr
* Correspondence: ojkwon@sejong.ac.kr

Abstract: Unlike traditional remote control systems for controlling unmanned aerial vehicles (UAVs) and drones, active research is being carried out in the domain of vision-based hand gesture recognition systems for drone control. However, contrary to static and sensor based hand gesture recognition, recognizing dynamic hand gestures is challenging due to the complex nature of multi-dimensional hand gesture data, present in 2D images. In a real-time application scenario, performance and safety is crucial. Therefore we propose a hybrid lightweight dynamic hand gesture recognition system and a 3D simulator based drone control environment for live simulation. We used transfer learning-based computer vision techniques to detect dynamic hand gestures in real-time. The gestures are recognized, based on which predetermine commands are selected and sent to a drone simulation environment that operates on a different computer via socket connectivity. Without conventional input devices, hand gesture detection integrated with the virtual environment offers a user-friendly and immersive way to control drone motions, improving user interaction. Through a variety of test situations, the efficacy of this technique is illustrated, highlighting its potential uses in remote-control systems, gaming, and training. The system is tested and evaluated in real-time, outperforming state-of-the-art methods. The code utilized in this study are publicly accessible. Further details can be found in the "Data Availability Statement".

Keywords: dynamic hand gesture recognition; drone control; socket communication in real-time; drone control simulation; vision UAVs; temporal data classification; vision-based gesture recognition

Academic Editors: Angelo Trotta, Gokhan Secinti, Marco Di Felice and Zhangyu Guan

Received: 13 January 2025
Accepted: 18 January 2025
Published: 24 January 2025

Citation: Yaseen; Kwon, O.-J.; Kim, J.; Lee, J.; Ullah, F. Vision-Based Gesture-Driven Drone Control in a Metaverse-Inspired 3D Simulation Environment. *Drones* **2025**, *9*, 92. https://doi.org/10.3390/drones9020092

1. Introduction

The use of drone or unmanned aerial vehicles (UAVs), in aerial surveillance, monitoring, agricultural, videography, scientific research, transportation, rescue, surveying and mapping has grown in popularity in recent years [1]. Naturally, UAVs interaction has increased in frequency, particularly with the advancement of Artificial Intelligence (AI). As a substitute for AI, the Intelligent Human UAVs Interaction (HUI) technique is currently being explored [2]. Conventional methods uses controllers with a joystick. However, contrary to a joystick-based controller's approach, HUI technology enables inexperienced users to interact with highly trained and sophisticated interfaces, making it simple for people to use with the intention of creating an intuitive and natural interface [3,4].

The goal of HUI research is developing more inventive design and natural interfaces, which are broadly divided into four types: portable sensors, speech recognition, hand gesture recognition, and more user-friendly remote controls [1]. Table 1 provides a summary

of a typical HUI system's benefits and drawbacks [2,3]. Specifically, there is a greater focus on research in hand-gesture-based recognition (HGR) systems due to the intuitive and natural qualities of hand gestures methods for individuals to communicate with each other and exchange important information [4–6].

Table 1. Types of human UAVs interaction systems.

UAVs Interaction	Pros	Cons
Remote Controllers	Improved user experience Less training time required	Not so budget friendly
Wearable Sensors	Low computation More intuitive Accurate for motion capture	Can misinterprets motion capture Less intuitive Distance limitation
Speech Recognition	Low computation More natural Accurate for motion capture	Can be influenced by noise Effected by language differences and utterances
Gesture Recognition	Natural Offers great design flexibility No aided hardware needed	High computational cost Offer low control over the system

The fact that different types of data can be used in HGR systems "as being explored in the related work", they can be broadly classified into two main categories: sensor-based hand gesture recognition S-HGR, and vision-based hand gesture recognition V-HGR [1]. These methods can be further classified based on the data collection procedure, data type, training methods used and more [7]. The S-HGR uses one-dimensional raw data that is taken from wearable gloves, are equipped with several sensors, such flex, electromyogram (EMG), and inertial measuremnet unit (IMU) [8]. The S-HGR system is resistant to external factors, such as lighting conditions, and has little computation since it uses suitable data without the need for feature extraction [8]. But since the V-HGR system is less expensive and simpler to operate than the S-HGR system, most of the active research is being carried out in this domain [9]. In the V-HGR systems the main focus is to work and analyze 2D images or sequences of images [10]. Researchers mostly use two types of approaches in analyzing and training for V-HGR namely, hand-crafted feature based and deep feature-based methods [11].

Active research these days is more focused on deep feature-based algorithms due to their enhance feature extraction abilities. However, unlike hand crafted feature-based approaches i.e., Markov models [12,13], and support vector machines (SVM) [14], deep feature-based methods pose challenges in terms of computational complexity. Most of the deep feature-based methods involve the detection, identification, and interpretation of both static and dynamic gestures [7,15,16].

In this context, a static gesture is one in which the gesture has a fixed pose in the entire sequence of images [17]; a dynamic gesture, on the other hand, combines several movement postures over a number of frames [18]. Neural-network-centered techniques provide great performance in the static gesture recognition scenario [10,19], but in the case of dynamic hand gestures recognition, computational complexity is very high, and the accuracy is relatively low as compared to the former one [20].

Tracking algorithms for temporal data classification, are combined with deep-learning networks to solve the issue of computational complexity. For example Hu et al. [21] tried to combine two frameworks, DeepSORT, Open POSE for skeleton extraction, and human tracking respectively. Similarly, kalman filter was used by Kassab et al. [22] for tracking patterns in a given sequence of frames. These methods have also shown improved performance but with computational cost [1].

While these methods have shown state-of-the-art performance, they increases computational complexity. However, the HUI system is a real-time system that prioritizes performance and efficiency, but its objective of safe and straightforward design is not well served by sophisticated algorithms [23]. To solve this problem, several CNN and 3D CNN based approaches have also been proposed [24–26]. They offered great solutions, but again at a high computational cost [25].

Therefore, we propose a simulation-based system "a hybrid solution" to focus on real-time performance and safety for drone control. By changing the initial feature extractor module to SqueezeNet [27] in the model proposed by Hax et al. [10] and fine tuning, we proposed a light weight hybrid SqueezeNet-LSTM based network. The system offers dynamic hand gesture recognition with improved accuracy and reduced training parameters for our 3D drone control simulator. In our proposed method, we also designed and created the virtual simulation environment that is faster and more user-friendly compared to the system proposed by [1]. To sum up, in the case of V-HGR systems where pose estimation is challenging, we exploit the temporal aspects of the data for dynamic hand gesture classification. The vision-based gesture control commands in the proposed system are mapped with the pre-defined text commands that are selected based on the gesture type and are being sent to the 3D simulator environment via sockets to efficiently control the simulated drone.

Overall, our novel approach exploits a streamlined set of gestures and a 3D simulation environment for remotely operating drones. With the add of socket communication, it serves as a virtual simulation-based system, paving the path for more advance options. Comparing the proposed system's usability and efficiency against those of standard joystick controllers and the S-HGR systems, our proposed system is faster, more reliable and user friendly. We highlight our contributions in this work as follows;

- Programmable V-HGR drone control via distributed simulation.
- State of the art method is being used to classify dynamic hand gesture recognition.
- Designed compact set of commands for controlling drone.

Furthermore, in the following sections, hand gesture recognition based related work has been discussed in Section 2, our proposed method has been discussed in Section 3. Experiments and results are presented in Section 4 with necessary discussion in 5. In Section 6, we conclude our work with future directions and planing. In summary, it is anticipated that the proposed method will offer an HUI interface that is faster, secure and easy for non-experts to use.

2. Related Work

Numerous methods have been proposed on gesture-based UAVs interaction, with different types of data used for improving recognition, control and interaction in real time. Sensors based systems use data from IMUs, LiDAR and infrared, to capture and localize hand moments precisely [28]. Vision-based gesture control systems exploit RGB data while some systems also incorporate hybrid models, combining data from all the sources (Sensors, RGB-depth) to enhance accuracy and robustness in the case of real time applications. In all these scenarios, the gesture classification is broadly classified as; sensors-based hand gesture recognition (S-HGR), and vision-based hand gesture recognition (V-HGR).

2.1. Sensor Based Hand Gesture Recognition

While several approaches are being used in S-HGR based methods, they can be broadly classified mainly as data-based and machine learning-based [8,18]. The data-based approach uses accelerometer data that are taken from hand-mounted IMU sensors. They provide useful details on the angular posture of the finger joints and the postures of the

hands, helping to identify movements [29,30]. The authors of the study [31] presented a method for implementing a similar approach based on hand motion tracking and identification. S-HGR data-driven based methods have advantages in terms of computational complexity for hand gesture recognition [12,13]. Nevertheless, there drawback is that it is cumbersome since users must wear sensors devices.

The ML-based S-HGR method is an approach of employing machine learning classifiers, and there are several classification strategies for gesture recognition algorithms, such as artificial neural networks (ANNs) [32], decision trees (DT), artificial neural networks (ANN) [32,33], support vector machine (SVM) [9,9], and K-nearest neighbors (KNN) [34]. Muezzinoglu et al. [35] evaluated results for sensors-based data classification using machine learning classifiers such as, DT, SVM, and KNN for S-HGR. While machine learning-based methods offer promising results with regards to accuracy, they have increased computational cost and reduced accuracy in the case of data obtained from untrained individuals [1,10]. Current research trends also focus on data based on electromyogram sensors (EMGs), IMU sensors, and flex sensors [8]. For example, Mardiyanto et al. [36] proposed IMU and flex sensors mounted on the wrist, elbow, and forearm to control the remote-operated underwater vehicle. Kim et al. [30] proposed a real-time hybrid system that combines data obtained from the EMG and IMU sensors. The suggested method uses the EMGs to measure hand force, while the IMU records arm motion. The mapping of time-varying signals is a challenging task in these systems as these depend on human physical conditions [29].

2.2. Vision Based Hand Gesture Recognition

In the context of computer vision, hand gesture recognition can be viewed as an object detection challenge [1,10], hence they are divided into two types: deep feature-based and handcrafted feature-based. In the handcrafted feature-based algorithms, the training data is manually extracted and labeled by humans. In the deep feature-based approach, data is labelled at pre-processing stage, followed by feature extraction, and then training by a classifier [11,17,19]. Although the handcrafted feature-based methods show limited performance in comparison to deep feature-based, the computational complexity is lower [35].

Deep feature-based methods, such as long short-term memory (LSTM) networks and convolutional neural networks (CNN), have demonstrated remarkable performance in hand gesture detection recently [19,37]. Numerous researches are being carried out in this domain, however there are still several issues with the deep feature-based approaches in gesture recognition that need to be resolved for dynamic V-HGR systems. For example in the case of dynamic V-HGR, Kassab et al. [22] proposed that any three moving body parts of the interacting person be detected in the frame, and then tracked using Kalman filter (KF). Another method was proposed by Liu et al. in [29] using skeleton extraction by OpenPOse and tracking is carried out with DeepSORT. Several other techniques have been proposed using 3D CNNs and hybrid models [29–31].

In summary, several methods and approaches have been proposed recently in the domain of UAVs. To advance state of the art and real time application-based solutions, we propose a vison-based V-HGR simulation system used for drone control in a 3D environment. The drone is virtually controlled through commands sent via sockets to a remote PC performing real time interaction, commands transmission and dynamic hand gesture recognition. Therefore, in this paper, we propose an easy-to-use interface that can be used even by non-professionals to control UAVs in a virtual environment. The system is further discussed in the proposed methodology.

3. The Proposed Architecture

This paper proposes an integrated 3D virtual environment system for UAVs flight that communicates with a V-HGR module via sockets. The two modules, V-HGR and 3D virtual environment are connected through IP protocols that enable an intuitive and real-time drone control experience through hand gestures as shown in Figure 1.

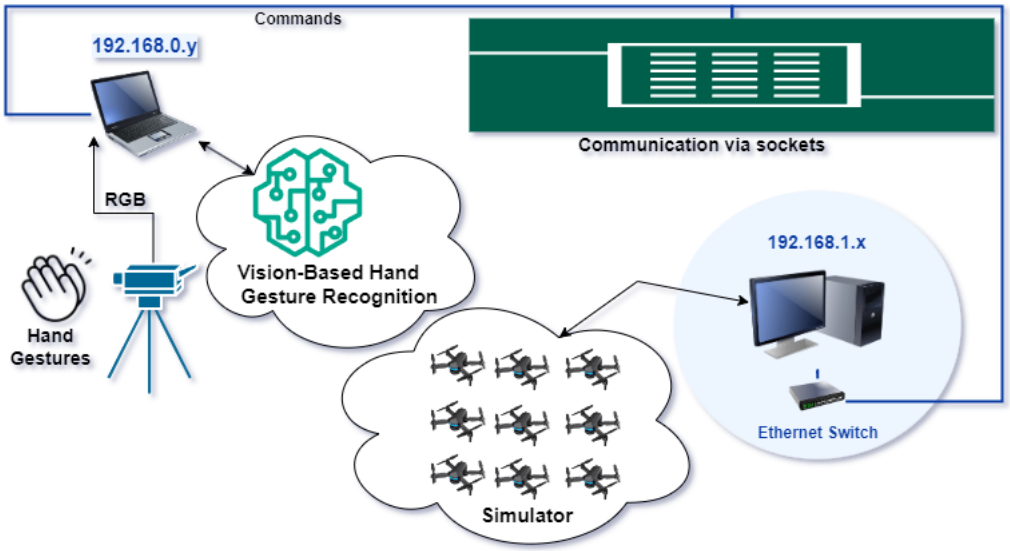

Figure 1. Gesture-Based UAVs Control: Demonstrating Human Gesture Recognition for Drone Navigation in a Simulated Environment via Sockets.

Next, the overall structure of the proposed method is discussed as follows.

3.1. Hand Gesture Recognition Module

The pipeline for hand gesture recognition module includes input dataset for training, prepossessing of data, model training and model inference.

3.1.1. Nature of Data and Input Method

The pipeline of V-HGR module uses RGB-hand gesture clips contained in the dataset discussed in Section 3.4. The data is labled according to the list of gestures given in Table 2. The model is trained on the dataset using the proposed model discussed in Section 3.1.2. At the inference stage, the V-HGR module takes input from an RGB camera. We use input from a desktop PC's webcam. The gestures are performed by both hands, and continuous RGB frames were used for gesture clips. Each gesture is a sequence of frames. The sequences of frames are converted to stack of features and are classified by the trained SqueezeNet-LSTM based algorithm. Figure 2 shows the nature of the input data to the system.

Figure 2. RGB-camera takes hand gestures as input from the frames.

Table 2. List of commands used for controlling drone.

Command	Description	Command	Description
Move backward	both hands thumbs backward	One	left thumb, right index finger
Move Left	both hands thumbs move left	Two	left thumb, right index, middle finger
Move right	both hands thumbs move right	Three	left thumb, right index, middle and ring finger
Move forward	both hands thumbs move forward	Rotate	left thumb, right index finger rotate clockwise
Move down	both hands thumbs move downward	Drone selection	both hands index finger up
Takeoff	left thumb and right open palm move up	Land	left thumb right open palm down
All clear	Arms down "V"	Have command	Circle arm overhead
Hover	Arm straight side wise	Landing direction	left thumb right open palm down
Not clear	Arms up together	Slow down	Arms horizontal motion
Wave off	Arms circle downward		

3.1.2. Dynamic Gesture Recognition

In this section, we discuss some background details and dynamic V-HGR module of our proposed system.

Gesture Types

As discussed in Section 1, gesture are of two types; static and dynamic. The static gestures have constant pose in a series of frames, while dynamic gestures have varying hand posture over a series of frames, as shown in the following Figure 3.

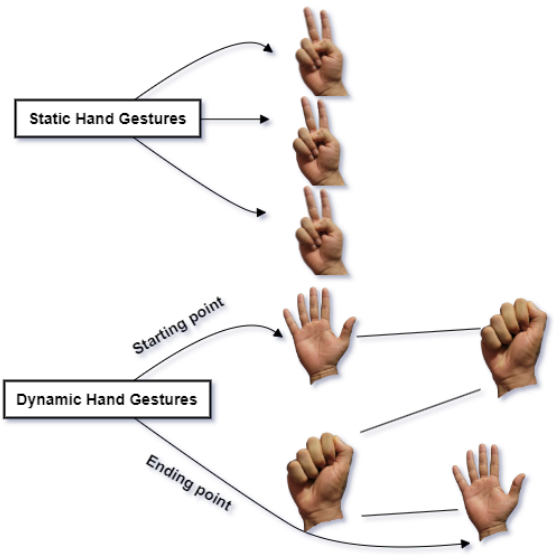

Figure 3. Static vs dynamic hand gestures.

List of Selected Gestures

As we know that models trained with biased datasets have poor performance on test data [38]. To avoid biases in the dataset, rather than using single hand gesture, we selected combination of both hands for each gesture. The list of gestures and their descriptions are given in the following Table 2.

Gesture Classification

Our proposed V-HGR module is a hybrid deep learning model that exploits the use of transfer learning. In order to be lightweight and more robust for dynamic gestures recognition, we used SqueezeNet [27] as a feature extractor for spatial data (2D images). The feature vectors obtained from the last layer of SqueezeNet was input to LSTM module. This LSTM module is used as a final classifier in the proposed method.

At the input stage, in the SqueezeNet module, the input convolutional layer is followed by max pooling layer. Then there are fire modules with max pooling layers in between. For improved gradient flow and faster convergence, we used complex bypassing in the fire modules, i.e. fire2 was directly connected to fire3 and fire4 through bypassed connections. Similar procedure was followed for the following layers of the fire5-fire9 modules, as shown in Figure 4. This complex bypassing allows a rich feature set across the model and thus allow the data pattern to bypass bottlenecks as used in the residual networks [39]. At the end, a convolutional layer passes the information to the global average pooling layer, which is input to the LSTM layer for temporal data classification. Initially our input to the network is a sequences of frames represented by $w \times h \times n$. Where n is the number of images in a gesture clip multiplied by resolution of frames. The length of the output feature vector is $512 \times n$, which is input to the LSTM layer. In order to improve the final accuracy of the entire model, we followed the layer patterns in the LSTM model according to the state-of-the-art methods [10,37]. The overall structure of the proposed V-HGR module for dynamic hand gesture recognition is shown in Figure 4.

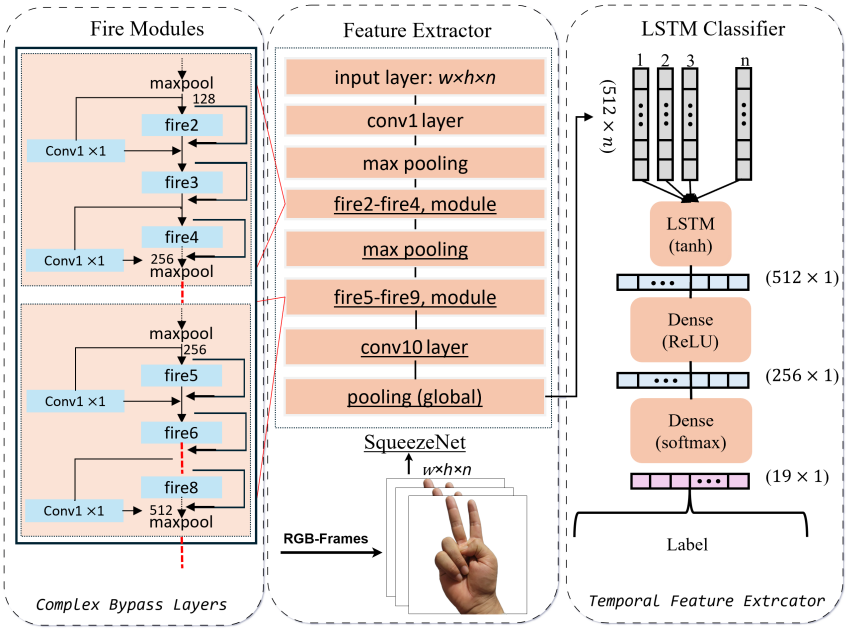

Figure 4. The architecture of the V-HGR model.

3.2. Drone Control in the Simulator

The proposed method's simulation environment is built on Unity Engine. A glimpse of this 3D environment is provided in Figure 5 as screenshots. This simulator provides a real-time user experience for drone control. The simulator is run on a separate remote PC and receives commands from the V-HGR module via sockets as shown in Figure 6. The simulator module operates on a separate desktop connected by IP address. Constant feedback is received by the V-HGR module for each command that is being executed. Drones' maneuvers are accurately performed based on the commands such as move left, move right, drone selection etc. (the list of commands are mentioned in Table 2). To mimic

real world flight dynamics, the Unity's physics engine is used for inertia, drifting and force vectors. Each of these functionalities are implemented based on the nature of command used. Figure 5 provides screen shot of the simulator environment showing some scenarios. We have designed a compact set of commands to perform various tasks, such as number of drones can be selected(as there are numbering and selection commands), the selected drones can be given commands for maneuverability, takeoff and landing etc. Figure 5 shows that single as well as multiple drones can be selected for flight operation.

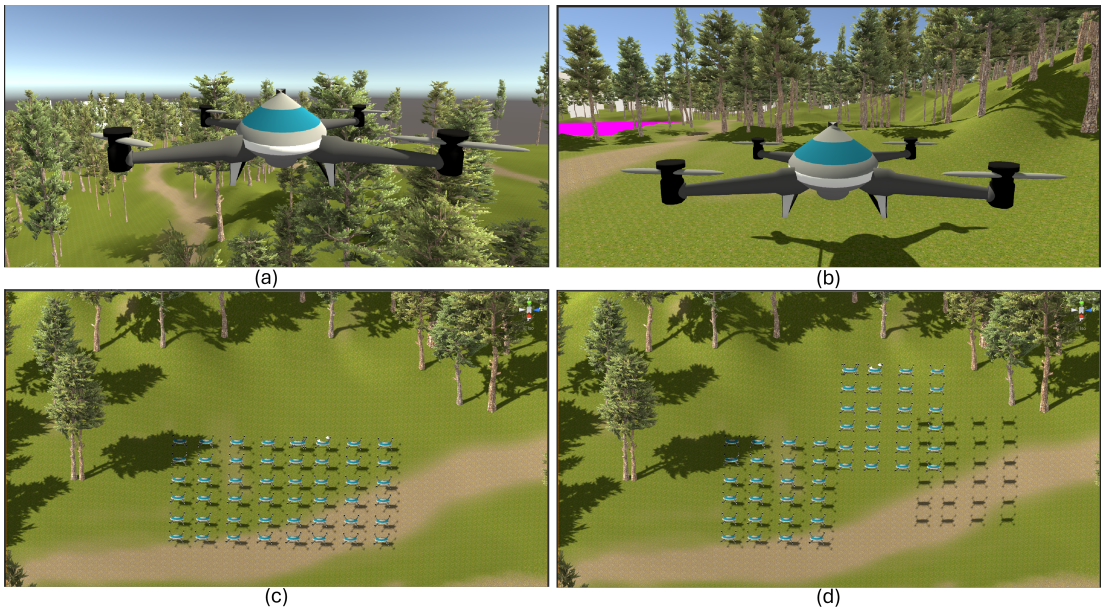

Figure 5. shows drones in a 3D environment, being controlled by commands received from the client side. (**a**) shows a single drone ready to take off. (**b**) shows the drone takeoff after a command was received. (**c**) shows swarm of drones waiting for command. (**d**) shows a scenario where a series of commands were received and executed; 1. drone selection, followed by 2. Two (to select half of the drones) and then followed by 3. Takeoff (selected drones took off).

3.3. Socket Communication

In our proposed system the 3D simulator acted as a server and V-HGR module as client as shown in Figure 6. For each command that has to be sent, a request is made and a connection is established. The messaging protocol along with key-factors are mentioned in Table 3. Upon the receiving commands from V-HGR module, the server send acknowledgment, and forward the command to the UAV's for operation.

Table 3. Overview of the server-client data exchange scenarios and key factors.

Module	Description	Key Factors
Socket Type	TCP (Transmission Control Protocol)	Ensures reliable delivery
Server-Client Setup	The server (simulator) actively listens commands from client (HGR system)	Multiple connections are being handled
Port Number	The communication channel	Both devices are configured
Data encoding	Commands are sent as byte streams	Encoding format: JASON format
Message size	Data stream size	Size is within buffer limits
Commands latency	Time taken in sending, processing and receiving the command	Acknowledge messages in TCP
Connection handling	Establishing, maintaining and terminating socket connection	Re-establish connection in case of failure

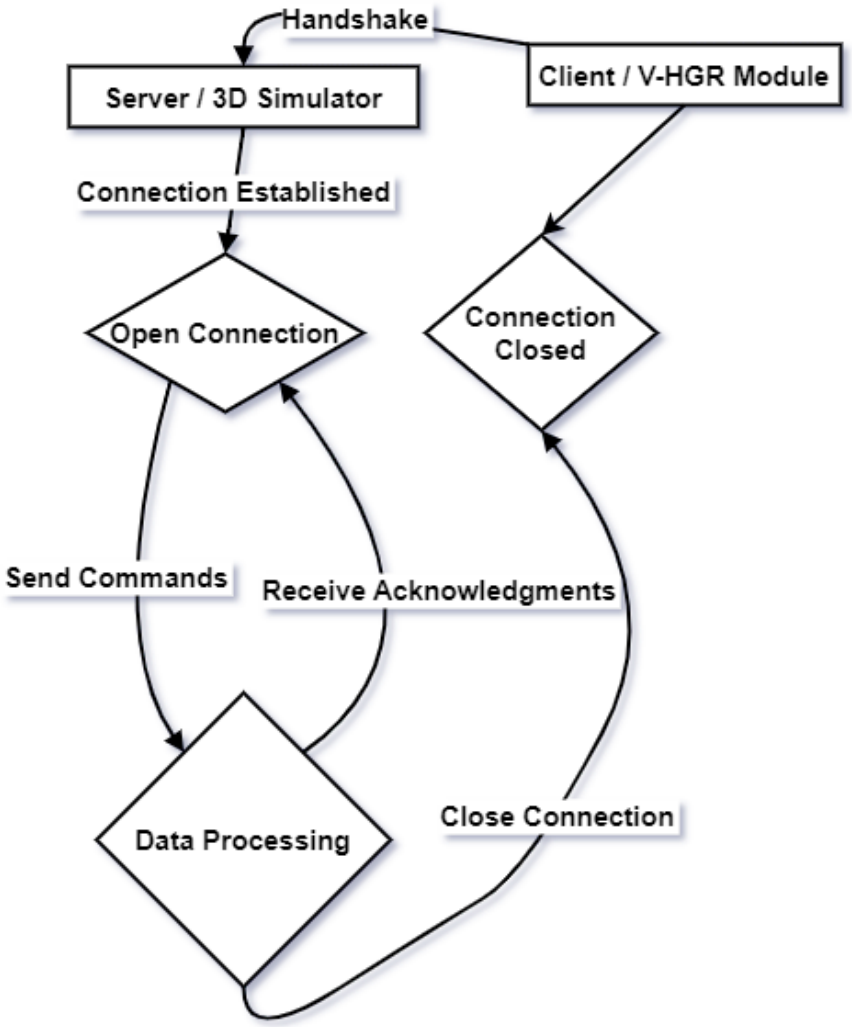

Figure 6. Communication between V-HGR and the 3D simulator module via sockets.

3.4. Dataset

Data hungry deep-learning models requires enormous amount of labeled data for training [10,22,29,34]. As we have designed a compact set of gestures for carrying out the UAV's basic operation, a creation of dataset was needed. Therefore, considering different factors, such as lighting condition, variation in distance from the camera, indoor outdoors scenes, we collected a total of 240 × 12 gesture clips using more than 30 subjects. To make sure model's improved performance, we merged an existing similar dataset specifically designed for dynamic gesture recognition called, "UAV-Gesture dataset" [40] with our newly created dataset. This resulted in larger training data for the model's improved performance. The merged dataset has a total of 19 gesture classes with 232,474 frames in total. The class distribution is shown in Figure 7. The list of class labels are reflected in the Table 2. Using five fold cross validation, we allocated 80% and 20% of the data for training and validation respectively.

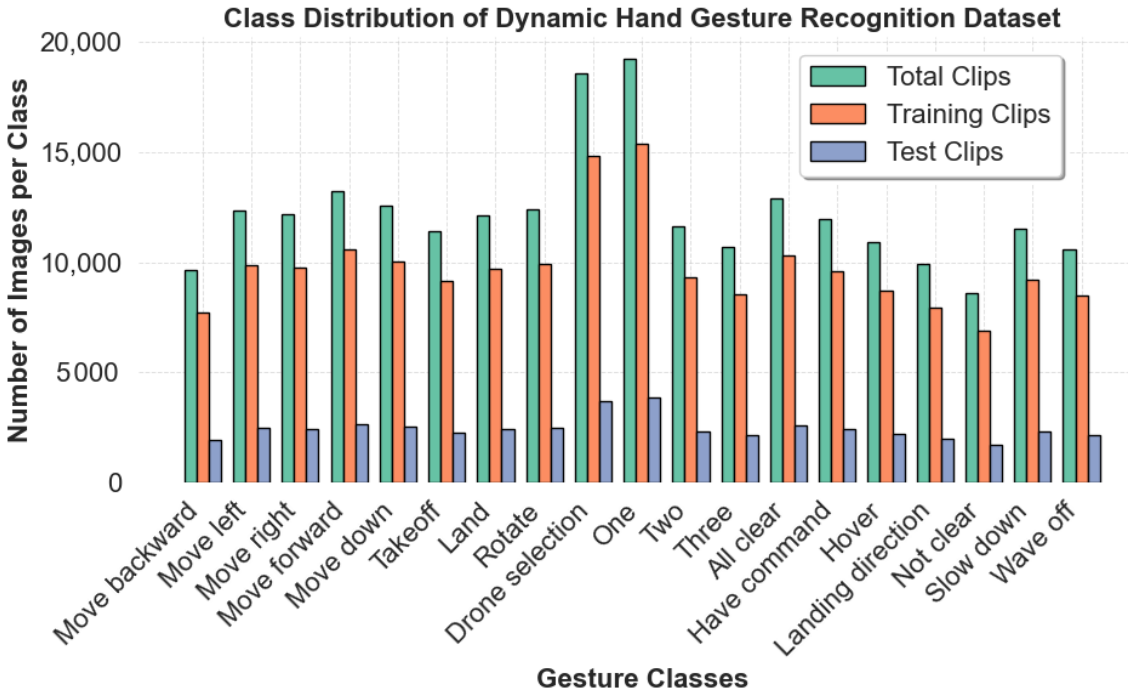

Figure 7. Dataset's class distribution. Class labels are shown on the x-axis, while number of images in each category are shown on the y-axis.

4. Experiments & Results

To evaluate the performance of the proposed method, we assessed the functionality of each module and the overall system using performance evaluation metrics, computational complexity and lap time.

4.1. Experimental Setup

HP desktop computers made by HP Inc. (Hong Kong), were used in our experiments. The model names are Pavilion Gaming Desktop TG01-1xxx. Equipped with GPU of NVIDIA GeForce RTX 2060 SUPER and RAM 32 GB. Having 8 cores with clock speed of 2.9 GHz.

We trained the model with 350 epochs, with early stopping option enabled. For training and validation losses the categorical cross-entropy cost function was used with an adam-optimizer for optimal weight adjustments. In the case of activation fucntion for LSTM network, tangent hyperbolic function (tanh) and for final output layer softmax function was used.

Ten subjects volunteered to test the proposed system. The V-HGR module was installed on one machine, and the 3D simulator was installed on another machine in the same room. The two machines were connected through IP addresses, as shown in the Figure 1. There mechanism is shown in Figure 6 and briefly described in Section 3.3. To test the proposed method in a real world scenario, the same process was repeated with replacing the 3D simulator based drone with a real world small drone. The E99 drone pro model was used for this purpose. The interfacing was done with the system via micro Arduino chip "ATmega32U4 microcontroller". Experimental as well as subjective test results were recorded for both scenarios. Comparison has been made against the existing methods and the results are reported in Tables 4, 5 and 6. Each subject carefully performed the list of gestures mentioned in Table 2. As the gestures recognition is vision-based, the subjects performed them with ease, whereas in the case of wearable gloves (and aided sensors), a constant distance has to be maintained and proper training is needed before operation.

4.2. Experimental Results

Experiments on the proposed system was carried out in terms of evaluation of; gesture classification with performance evaluation metrics, computational cost, and lap time in comparison with traditional baseline methods.

Evaluation of Vision-Based Gesture Classification

We performed subjective testing to assess the performance of dynamic gesture recognition system. In accordance with the active research, the assessment is usually carried out based on confusion matrix [9,22,32]. Figure 8 shows confusion matrix based on test dataset. Further results are presented based on the test data evaluation for the V-HGR module as shown in Table 4. The confusion matrix clearly shows that the trained model was able to correctly identify true positives, true negative as actual positives and actual negatives. Similarly, the performance evaluation metrics, when compared to the baseline results, show that the model performed well.

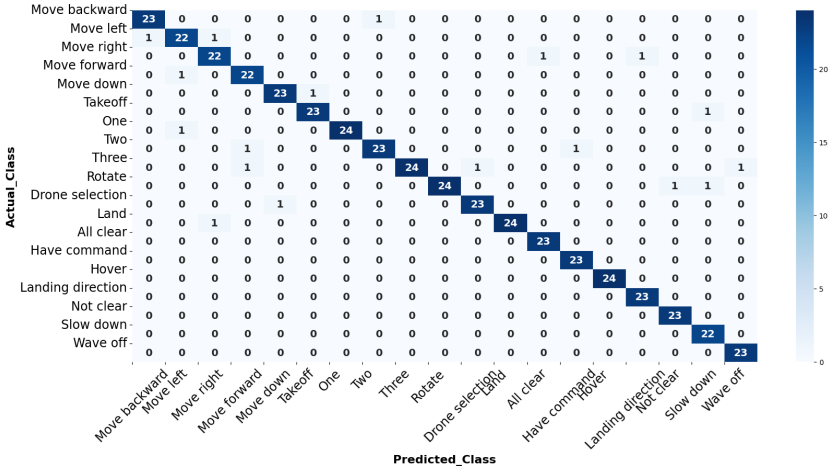

Figure 8. Confusion matrix for vision-based dynamic gesture recognition trained model based on the test data.

Further evaluation of the proposed system was conducted based on subjective testing. A total of 10 subjects participated in the test for operating the drone via hand gestures in front of the RGB camera, they were given a little training on the spot for hand gesture activity. Each of the subjects performed gestures based on the labels listed in Table 3. Different test scenarios were considered for testing, such as clear environment with proper lighting conditions, with varying hand speed, in low and bright lighting conditions and with varying hand distance from the camera. The accuracy was assessed based on the number of true positives, true negatives, false positives and false negatives. The results showed that the gestures were accurately recognized without the influence of the subject being used. Further more, accuracy and computational complexity for the V-HGR module was also calculated for comparison with baseline-methods as shown in Table 4. Kassab et al. [22] proposed a simplified Tiny-YOLOv2 for dynamic hand gesture recognition. Chen, B. [34] proposed dynamic gesture recognition model using graph neural network (GCN). Liu et al. [29] developed a convolutional neural network-based algorithm for dynamic gesture recognition. Based on the data presented in Table 5, it can be seen clearly that the proposed method outperformed all the baseline methods and took less time in comparison with the baseline methods. Thus, data presented in these Tables 4 and 5, verifies that the V-HGR module in our system is well suited for controlling drone.

Table 4. Accuracy based on each classs data for which the proposed model was trained.

	Gesture Classes	Accuracy
Vision-Based HGR	Move backward	97.8%
	Move left	96.5%
	Move right	98.1%
	Move forward	95.2%
	Move down	96.5%
	Takeoff	96.2%
	One	97.7%
	Two	95.6%
	Three	97.4%
	Rotate	96.3%
	Drone selection	97.0%
	Land	95.5%
	All clear	96.8%
	Have command	98.4%
	Hover	97.1%
	Landing direction	96.3%
	Not clear	97.7%
	Slow down	98.0%
	Wave off	97.9%

Table 5. Comparison of the proposed method for dynamic hand gesture recognition with the base line algorithms.

Algorithm	Proposed Systems	# of Classes	Accuracy	Computational Cost (ms)
Tiny YOLOv2 [22]	HGR for UAVs	10	90%	42.70
GNN [34]		6	81%	45.00
CNN [29]		2	80%	20.00
Micro-IMU [1]		8	93%	0.089
Hax et al. [10]		12	84.5%	15.80
UAV-Gesture [40]		13	91.9%	>20
Ours		12	96.9%	0.027

4.3. Evaluation Based on Time Lap

One of the flight performance metrics in drone control is the lap time [8,16]. The purpose of the proposed system is to build a natural, safe and intuitive interface for easy drone control by inexperienced users, hence the test was performed using lap time. A total of 10 subjects participated in the experiment. Lap times were calculated for the proposed systems virtually as well as in real world scenario, for comparison with baseline methods. The time it took to perform control commands using all gestures (mentioned in Table 2) by each individual subject was recorded as lap time. The results showed that our system was more suitable for use by non-experts (who had no prior experience of drone control). Reflecting on the improved user experience of our system. The proposed system's performance was also compared with existing methods and the results are listed in below Table 6. All of the systems uses vision-based gesture recognition modules, comparing their detection time, accuracy and lap time, our proposed method outperformed the existing solutions.

Table 6. Comparison results with the existing dynamic gestures based UAVs systems.

Proposed Methods	Machine	Data Used	Total Gestures	Detection Time (ms)	Lap Time (ms)	Speed (fps)
HUI system [41]	UAV	RGB	3	47.61	333.27	21
H-DTV Interaction [42]	TV	Depth + RGB	6	35.21	246.47	28
Intelligent HUI System [34]	UAV	Skeleton + RGB	6	57.03	399.21	26
Dynamic HUI System [22]	UAV	RGB	10	42.92	300.44	25
Ours (Virtual)	UAV	RGB	19	12.34	86.38	29
Ours (Real World)	UAV	RGB	19	15.14	105.98	29

5. Discussion

This research presents a vision-based gesture recognition system for remotely controlling drones operation in a virtual environment. The compact gesture set and quick system response time ensure intuitive control and at the same time, with minimal cognitive load of operators. This is accomplished by designing a small yet compact set of easily memorable gestures (a total of 19), thus guaranteeing both factually attainable functionality, and reliability when the goals are set on operation safety in harsh scenarios. Unlike with more complex gesture systems, or the very basic ones which may overwhelm users with memorizing tens of gestures thus eventually reducing effectiveness in practical scenarios.

In contrast to other existing solutions for gesture recognition, our proposed system is simple and flexible. It is also essential to remember that the gesture set was designed to have as few basic commands as possible while maximizing the functionality. The fact that the basic commands can be expanded based on the use case scenario, showcases the uniqueness of our proposed system. Furthermore, the use of socket communication for remote operation and control in a metaverse-like environment further highlight the novelty of the system. The classification goals were achieved using dynamic hand gesture recognition system by exploiting a hybrid light weight model which is another novel aspect of our proposed system. The hybrid design of the model with complex bypassing fire modules not only make the model lightweight but also highly efficient in recognizing patterns in temporal data, with an overwhelmingly quick response time as shown in Table 6. Also, the model's performance showcases strong capacity for temporal data processing which can be easily adopted for similar vision based tasks.

Limitations

In spite of its advantages, our system has some limitations at present. Unlike in case with most of the datasets, we have intentionally kept our set of gestures minimal, 19 base gestures (or classes) so as to ensure easy usability of the system by the users. This may seem as a limiting factor at first glance, but it complies with the principle of having less cognitive load and fewer operation risks. Datasets with more complex set of gestures have been proven to be less effective and impractical in real-world scenarios [43,44].

Additionally, the current implementation details in this study are restricted only to simulation based operation (although we have tested the system with real drone and provided the results in Table 6, implementation of hardware details are beyond the scope of this study). Additionally, the hardware testing results completely aligned with that of testing in virtual system as can be seen in the Table 6. Applying the system for controlling real physical drones would require further optimization techniques in interfacing the hardware and fine tuning the use case scenarios. Also, the gestures set may further requires expansion in case, swarm of drone control in real world scenario is required, this is because of two reasons; further derived commands will be needed, and the hardware interfacing for multiple drones will need optimization and development.

6. Conclusions

We propose a vision-based real-time human-UAV interaction system in a virtual environment. Rather than using aided wearable sensor based or a standard joystick-based control system that demands a lot of training time and effort, it focuses to mimic a natural and intuitive interface that is simple enough for non-experts to use. Currently active research is being carried out on vision-based HGR systems.

However, to recognize dynamic gestures, a vision-based system comes with complex computational costs. For drone control in real world scenarios, a hybrid system was used for dynamic hand gesture recognition in a virtual environment. Each gesture was assigned a unique command for controlling the drone in a 3D environment. With improved accuracy and precision of the proposed algorithm, the creation of a natural and intuitive interface based system was possible.

To assess the performance and effectiveness of our proposed system, subjective evaluation was conducted in terms of classification metrics and lap time. A comparison of the proposed method was made with the baseline algorithms. The classification performance for dynamic hand gesture recognition of 95.6% mAP was recorded and the effectiveness of the gesture detection module was confirmed by subjective testing. The latency of the proposed method was also compared with baseline methods which showed that the operation was completed more quickly compared to the aided sensor-based systems. Hence, it was confirmed that the proposed method offers an intuitive and safer way for human-UAV control system. Also, according to the subjects, the performance with respect to classification accuracy remains steady, showing that the system can be operated by non-experts without any intense training needed.

Although the proposed system was intended for a 3D virtual environment, its basic functionality was tested via a real drone in an indoor environment, for advance drone operation in an outdoor environment certain points need to be considered; i.e., environmental conditions, interfacing hardware optimizations and subjects training. Future work will also include expanding use case scenarios, deriving more complex set of gestures commands (deriving from the basic set of gesture proposed in this study). And at last, system optimization for swarm of hardware drone control for complex scenarios like aerial art and more.

Author Contributions: Conceptualization, Y. and O.-J.K.; data curation, Y.; formal analysis, Y.; funding acquisition, O.-J.K., J.K. and J.L.; project administration, J.L.; resources, J.L.; software, Y., J.L. and F.U.; supervision, O.-J.K.; visualization, Y; writing—original draft, Y.; writing—review and editing, Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Unmanned Vehicles Core Technology Research and Development Program through the National Research Foundation of Korea (NRF) and Unmanned Vehicle Advanced Research Center (UVARC) funded by the Ministry of Science and ICT, the Republic of Korea (2023M3C1C1A01098414). This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2024-2021-0-01816) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation).

Data Availability Statement: The code generated and analyzed during the current study is available at: https://github.com/yaseen21khan/HGR-based-drone-control-in-Unity (accesssed on 17 January 2025).

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

Abbreviations

The following abbreviations are used in this manuscript:

UAVs	Unmanned aerial vehicles
HUI	Intelligent human UAVs interaction
S-HGR	Sensor-based hand gesture recognition
V-HGR	Vision-based hand gesture recognition
LiDAR	Light Detection and Ranging

References

1. Yoo, M.; Na, Y.; Song, H.; Kim, G.; Yun, J.; Kim, S.; Moon, C.; Jo, K. Motion estimation and hand gesture recognition-based human–UAV interaction approach in real time. *Sensors* **2022**, *22*, 2513. [CrossRef]
2. Zhang, X.; Zhang, H.; Sun, K.; Long, K.; Li, Y. Human-Centric Irregular RIS-Assisted Multi-UAV Networks with Resource Allocation and Reflecting Design for Metaverse. *IEEE J. Sel. Areas Commun.* **2024**, *42*, 603–615. [CrossRef]
3. Sun, L.; Liu, Z.; Ning, Z.; Wang, J.; Fu, X. Multi-Agent Q-Net Enhanced Coevolutionary Algorithm for Resource Allocation in Emergency Human-Machine Fusion UAV-MEC System. *IEEE Trans. Autom. Sci. Eng.* **2024**. [CrossRef]
4. Javed, S.; Hassan, A.; Ahmad, R.; Ahmed, W.; Ahmed, R.; Saadat, A.; Guizani, M. State-of-the-art and future research challenges in uav swarms. *IEEE Internet Things J.* **2024**, *11*, 19023–19045. [CrossRef]
5. Sun, L.; Wang, J.; Wan, L.; Li, K.; Wang, X.; Lin, Y. Human-UAV Interaction Assisted Heterogeneous UAV Swarm Scheduling for Target Searching in Communication Denial Environment. *IEEE Trans. Autom. Sci. Eng.* **2024**. [CrossRef]
6. Wang, W.; Xu, X.; Bilal, M.; Khan, M.; Xing, Y. Uav-assisted content caching for human-centric consumer applications in iov. *IEEE Trans. Consum. Electron.* **2024**, *70*, 927–938. [CrossRef]
7. HN, N.K.; Prasad, G.; Chandrappa, S.; Gujjar, P. Integration of Computer Vision Techniques, UAV, and Metaverse Analysis to Uplift Healthcare Services. In *Ubiquitous Computing and Technological Innovation for Universal Healthcare*; IGI Global: Hershey, PA, USA, 2024; pp. 175–200.
8. Sharmila, P.; Maheswaran, M.; Mohanraj, T.; Verma, R.; Malviya, B.; SubbaRao, S. A Way of Safe Wireless Networks using IMU/UWB/Vision through Sensor Networks. In Proceedings of the 2024 4th International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India, 14–15 May 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 266–271.
9. Jing, X.; Liu, F.; Masouros, C.; Zeng, Y. ISAC from the sky: UAV trajectory design for joint communication and target localization. *IEEE Trans. Wirel. Commun.* **2024**, *23*, 12857–12872. [CrossRef]
10. Hax, D.R.T.; Penava, P.; Krodel, S.; Razova, L.; Buettner, R. A Novel Hybrid Deep Learning Architecture for Dynamic Hand Gesture Recognition. *IEEE Access* **2024**, *12*, 28761–28774. [CrossRef]
11. Noh, D.; Yoon, H.; Lee, D. A Decade of Progress in Human Motion Recognition: A Comprehensive Survey From 2010 to 2020. *IEEE Access* **2024**, *12*, 5684–5707. [CrossRef]
12. Saha, A.; Rajak, S.; Saha, J.; Chowdhury, C. A survey of machine learning and meta-heuristics approaches for sensor-based human activity recognition systems. *J. Ambient. Intell. Humaniz. Comput.* **2024**, *15*, 29–56. [CrossRef]
13. Sok, P.; Xiao, T.; Azeze, Y.; Jayaraman, A.; Albert, M.V. Activity recognition for incomplete spinal cord injury subjects using hidden Markov models. *IEEE Sens. J.* **2018**, *18*, 6369–6374. [CrossRef]
14. Abidine, B.M.; Fergani, L.; Fergani, B.; Oussalah, M. The joint use of sequence features combination and modified weighted SVM for improving daily activity recognition. *Pattern Anal. Appl.* **2018**, *21*, 119–138. [CrossRef]
15. Madavan, M.; Kumar, A.; Ramakrishnan, A.B.; Manikandan, R.; Magesh, S. Incorporation of Computer Vision and Metaverse Analysis Using UAV Communications for Healthcare Applications. In *Ubiquitous Computing and Technological Innovation for Universal Healthcare*; IGI Global: Hershey, PA, USA, 2024; pp. 252–273.
16. Kang, J.; Chen, J.; Xu, M.; Xiong, Z.; Jiao, Y.; Han, L.; Niyato, D.; Tong, Y.; Xie, S. Uav-assisted dynamic avatar task migration for vehicular metaverse services: A multi-agent deep reinforcement learning approach. *IEEE/CAA J. Autom. Sin.* **2024**, *11*, 430–445. [CrossRef]
17. Hu, Z.; Qiu, F.; Sun, H.; Zhang, W.; Ding, Y.; Lv, T.; Fan, C. Learning a compact embedding for fine-grained few-shot static gesture recognition. *Multimed. Tools Appl.* **2024**, *83*, 79009–79028. [CrossRef]
18. Sun, Y.; Huang, J.; Cheng, Y.; Zhang, J.; Shi, Y.; Pan, L. High-accuracy dynamic gesture recognition: A universal and self-adaptive deep-learning-assisted system leveraging high-performance ionogels-based strain sensors. *SmartMat* **2024**, *5*, e1269. [CrossRef]

19. Kapitanov, A.; Kvanchiani, K.; Nagaev, A.; Kraynov, R.; Makhliarchuk, A. HaGRID–HAnd Gesture Recognition Image Dataset. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, Hi, USA, 3–8 January 2024; pp. 4572–4581.
20. Zhang, H.; Huang, S.L.; Kuruoglu, E.E. HGR Correlation Pooling Fusion Framework for Recognition and Classification in Multimodal Remote Sensing Data. *Remote Sens.* **2024**, *16*, 1708. [CrossRef]
21. Hu, Q. Enhancing American Sign Language Communication with Virtual Reality: A Gesture Recognition Application on Oculus Quest 2. Ph.D. Thesis, Carleton University, Ottawa, ON, Canada, 2024.
22. Kassab, M.A.; Ahmed, M.; Maher, A.; Zhang, B. Real-time human-UAV interaction: New dataset and two novel gesture-based interacting systems. *IEEE Access* **2020**, *8*, 195030–195045. [CrossRef]
23. She, X.; Ma, H.; Ren, H.; Li, H. Vision-based adaptive prescribed-time control of UAV for uncooperative target tracking with performance constraint. *J. Syst. Sci. Complex.* **2024**, *37*, 1956–1977. [CrossRef]
24. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
25. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
26. Al-Hammadi, M.; Muhammad, G.; Abdul, W.; Alsulaiman, M.; Bencherif, M.A.; Mekhtiche, M.A. Hand gesture recognition for sign language using 3DCNN. *IEEE Access* **2020**, *8*, 79491–79509. [CrossRef]
27. Koonce, B.; Koonce, B. SqueezeNet. In *Convolutional Neural Networks with Swift for Tensorflow: Image Recognition and Dataset Categorization*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 73–85.
28. Cruz, P.J.; Vásconez, J.P.; Romero, R.; Chico, A.; Benalcázar, M.E.; Álvarez, R.; Barona López, L.I.; Valdivieso Caraguay, Á.L. A Deep Q-Network based hand gesture recognition system for control of robotic platforms. *Sci. Rep.* **2023**, *13*, 7956. [CrossRef]
29. Liu, C.; Szirányi, T. Real-time human detection and gesture recognition for on-board UAV rescue. *Sensors* **2021**, *21*, 2180. [CrossRef] [PubMed]
30. Li, C.; Li, S.; Gao, Y.; Zhang, X.; Li, W. A two-stream neural network for pose-based hand gesture recognition. *IEEE Trans. Cogn. Dev. Syst.* **2021**, *14*, 1594–1603. [CrossRef]
31. Kim, J.H.; Thang, N.D.; Kim, T.S. 3-D hand motion tracking and gesture recognition using a data glove. In Proceedings of the 2009 IEEE International Symposium on Industrial Electronics, Seoul, Republic of Korea, 5–8 July 2009; IEEE: Piscataway, NJ, USA, 2009; pp. 1013–1018.
32. Abdullahi, S.B.; Bature, Z.A.; Gabralla, L.A.; Chiroma, H. Lie recognition with multi-modal spatial–temporal state transition patterns based on hybrid convolutional neural network–bidirectional long short-term memory. *Brain Sci.* **2023**, *13*, 555. [CrossRef]
33. Shan, J.; Jiang, W.; Huang, Y.; Yuan, D.; Liu, Y. Unmanned aerial vehicle (UAV)-Based pavement image stitching without occlusion, crack semantic segmentation, and quantification. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 17038–17053. [CrossRef]
34. Chen, B.; Hua, C.; Li, D.; He, Y.; Han, J. Intelligent Human–UAV interaction system with joint cross-validation over Action–Gesture recognition and scene understanding. *Appl. Sci.* **2019**, *9*, 3277. [CrossRef]
35. Müezzinoğlu, T.; Karaköse, M. An intelligent human–unmanned aerial vehicle interaction approach in real time based on machine learning using wearable gloves. *Sensors* **2021**, *21*, 1766. [CrossRef]
36. Mardiyanto, R.; Utomo, M.F.R.; Purwanto, D.; Suryoatmojo, H. Development of hand gesture recognition sensor based on accelerometer and gyroscope for controlling arm of underwater remotely operated robot. In Proceedings of the 2017 International Seminar on Intelligent Technology and Its Applications (ISITIA), Surabaya, Indonesia, 28–29 August 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 329–333.
37. López, L.I.B.; Ferri, F.M.; Zea, J.; Caraguay, Á.L.V.; Benalcázar, M.E. CNN-LSTM and post-processing for EMG-based hand gesture recognition. *Intell. Syst. Appl.* **2024**, *22*, 200352.
38. Fabbrizzi, S.; Papadopoulos, S.; Ntoutsi, E.; Kompatsiaris, I. A survey on bias in visual datasets. *Comput. Vis. Image Underst.* **2022**, *223*, 103552. [CrossRef]
39. Gao, K.; Zhang, H.; Liu, X.; Wang, X.; Xie, L.; Ji, B.; Yan, Y.; Yin, E. Challenges and solutions for vision-based hand gesture interpretation: A review. *Comput. Vis. Image Underst.* **2024**, *248*, 104095. [CrossRef]
40. Perera, A.G.; Wei Law, Y.; Chahl, J. UAV-GESTURE: A dataset for UAV control and gesture recognition. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops, Munich, Germany, 8–14 September 2018.
41. Monajjemi, M.; Mohaimenianpour, S.; Vaughan, R. UAV, come to me: End-to-end, multi-scale situated HRI with an uninstrumented human and a distant UAV. In Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, Republic of Korea, 9–14 October 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 4410–4417.
42. Zhang, S.; Zhang, S. A novel human-3DTV interaction system based on free hand gestures and a touch-based virtual interface. *IEEE Access* **2019**, *7*, 165961–165973. [CrossRef]

43. Dang, T.L.; Pham, T.H.; Dao, D.M.; Nguyen, H.V.; Dang, Q.M.; Nguyen, B.T.; Monet, N. DATE: A video dataset and benchmark for dynamic hand gesture recognition. *Neural Comput. Appl.* **2024**, *36*, 17311–17325. [CrossRef]
44. Nayan, N.; Ghosh, D.; Pradhan, P.M. A multi-modal framework for continuous and isolated hand gesture recognition utilizing movement epenthesis detection. *Mach. Vis. Appl.* **2024**, *35*, 86. [CrossRef]