Paper ID #49611

Gesture-Based Drone Control: Enhancing Precision with Code Algorithms

Mathew Allen, Marshall University
Ben Taylor, Marshall University
Prof. Pingping Zhu, Marshall University

Prof. Pingping Zhu is an assistant professor in the Department of Computer Sciences and Electrical Engineering at Marshall University.

Preston K Sellards, Marshall University

Gesture-Based Drone Control: Enhancing Precision with Code Algorithms

Abstract

This research focuses on advancing the control of drones by utilizing hand gesture recognition through the drone's camera. By leveraging computer vision algorithms and machine learning techniques implemented in Python, this research aims to enhance the responsiveness and precision of drone operations in various environments. The core objectives involve developing and implementing gesture recognition mechanisms, emphasizing the need for drones to interpret and respond to human gestures accurately. Additionally, integrating real-time video processing to gather information about the operator's gestures will be crucial for effective control. Thorough testing and algorithm refinements are integral to this research to improve the efficiency and reliability of drone control via hand gestures.

Beyond technological advancements, the significance of this research lies in its potential applications across diverse sectors, such as search and rescue operations, entertainment, and human-computer interaction. The drone control strategies developed in this research hold promise for enhancing user interaction and contributing to more intuitive and accessible drone operations across various industries. These advancements signify progress towards establishing a more user-friendly and versatile autonomous drone landscape, highlighting the research's implications and practical applications.

1. Introduction

The growing demand for intuitive and efficient control systems in drone technology has spurred significant research into alternative methods for interacting with these devices. While traditional drone control methods, such as joysticks and remote controllers, have been effective, they often lack the flexibility and ease of use required for a seamless user experience, especially in changing or complex environments. As drones become more integrated into sectors, such as search and rescue, entertainment, and human-computer interaction, the need for more accessible and responsive control mechanisms has become more apparent. In particular, the ability to control drones through natural human gestures presents a promising solution to enhance user interaction and improve operational efficiency.

Recent advancements in computer vision and machine learning have opened new possibilities for gesture-based control systems. By using a drone's onboard camera to detect and interpret hand gestures, it is possible to create a more intuitive and versatile control interface. The integration of real-time video processing and gesture recognition algorithms can enable more precise and responsive drone operations, reducing the complexity of traditional control methods. Despite the promise of these technologies, challenges remain in developing systems that can accurately and reliably interpret gestures in complex environmental conditions.

This paper will explore the potential of hand gesture recognition as a method of controlling drones, focusing on the development and implementation of machine learning algorithms and computer vision techniques to enhance the precision and responsiveness of drone operations. By addressing key challenges such as real-time processing and gesture accuracy, this research aims to contribute to the creation of more accessible and effective drone control systems. Furthermore, the paper will examine the broader implications of this technology, considering its potential applications across industries and its role in shaping the future of autonomous drone operations.

2. Methodology

This research uses a hand gesture recognition system to control a drone, using computer vision and machine learning techniques implemented in Python. The main methodology involves detecting hand gestures using MediaPipe Hands, calculating the angles between finger lines, and associating specific gestures with corresponding drone commands.

2.1MediaPipe Hands

The hand gesture recognition system begins with the MediaPipe Hands module, which is an advanced real-time hand tracking solution developed by Google. MediaPipe provides a set of 21 landmarks for each detected hand, which are used to define key points on the hand's surface. These landmarks represent the positions of specific joints and tips on the hand, such as the tips of the fingers, knuckles, and wrist.

2.2 Angle and Distance Calculations for Gesture Recognition

A key aspect of this research is the development of a function, util.py, which calculates the angles between lines formed by specific hand landmarks. This function is used to determine the relative orientation and position of each finger. The primary metric for recognizing gestures is the angle between these lines, rather than distances between individual points so that the system does not depend on the size of the hand or the distance between the hand and the camera.

To calculate the angle between any two lines formed by three or more landmarks, the util.py function computes the angle formed by vectors based on the 3D coordinates of the landmarks. This process allows the system to determine precise finger movements and configurations, which are needed for differentiating between various hand gestures.

While angles are the main focus, the util.py function also calculates distances between relevant landmarks. These distances assist in detecting specific hand configurations, such as whether a finger is fully extended or bent. These calculations are secondary to the angle-based recognition to avoid issues related to hand size or distance.

2.3 Gesture Recognition and Drone Commands

Once the angles between the fingers are calculated, the main function compares these values with predefined gesture patterns stored in the system. Each hand gesture corresponds to a specific command, such as "take off," "land," "move forward," or "turn left." The gesture recognition function evaluates the current hand configuration and activates the corresponding drone action based on the recognized gesture.

Each gesture is defined by a set of conditions for the angles between specific fingers. For example, a gesture for "thumbs up" might involve specific angles between the thumb, index finger, and palm, while a "peace sign" could be recognized by the relative angles of the index and middle fingers. The system compares real-time values from the MediaPipe Hands output with these predefined angle thresholds to identify the gesture.

Once a gesture is identified, the corresponding drone command is triggered by the main function. This step is performed using a drone command in the Bitcraze library. A laptop webcam is currently being used for this system for stability and accuracy, a camera from one of the drones is planned to be used in the future.

2.4 Gesture Recognition Testing

The gesture recognition system was tested using a variety of hand gestures under different environmental conditions to evaluate its performance and robustness. These tests involved varying the hand's distance from the camera, lighting conditions, and background complexity to assess the system's ability to reliably recognize gestures. Adjustments were made to the angle and distance thresholds in the gesture defining functions based on experimental results to improve accuracy across different settings.

2.5 Video Processing and Feedback

The entire system is designed to function in real-time, utilizing a camera to capture video and process gestures. The camera feeds are continuously analyzed to detect hand landmarks, calculate angles, and recognize gestures. Feedback from the gesture recognition system is immediately implemented in the drone control, allowing for responsive command execution.

3. Results

The implementation of the hand gesture recognition system for drone control has shown promising results, successfully allowing us to control the drone through hand gestures. The system, utilizing MediaPipe Hands for gesture detection and custom Python functions for gesture recognition, has demonstrated the ability to execute a variety of drone commands in response to specific hand gestures.

3.1 Gesture Recognition Performance

The gesture recognition system performs well under controlled conditions, particularly when the hand is positioned perpendicular to the camera's view. In these ideal scenarios, the recognition accuracy is high, with gestures being detected and mapped to corresponding drone commands with minimal delay. The system is responsive, and the gesture-to-command mapping is intuitive, allowing for smooth and effective control of the drone. The speed of the system is good, with real-time processing and quick execution of commands, making the interaction natural and seamless.

3.2 Challenges with Hand Orientation and Camera Movement

The primary challenge encountered during testing relates to the orientation of the hand. The system requires the hand to be nearly perpendicular to the direction the camera is facing in order to achieve the highest accuracy in gesture recognition. When the hand is not perpendicular to the camera direction the accuracy of gesture detection decreases. This issue is compounded when the camera is in motion, as the hand's position relative to the camera changes dynamically, making it harder to maintain precise gesture recognition.

To mitigate this issue, the current setup relies on a fixed camera, specifically a laptop webcam, to achieve the best results. While this approach provides more stable hand detection, it limits the flexibility of the system, particularly in scenarios where the camera is mounted on the drone itself. The ability to use a moving camera without sacrificing accuracy remains a key challenge and will be a focus for future development. The system needs to be robust enough to handle variable camera angles and dynamic hand positions to make it viable for real-world applications where the camera cannot remain stationary.

3.3 Connectivity and Range Limitations

Another challenge observed during testing is related to the communication range between the laptop and the drone. The current radio being used limits the effective range of the system, meaning the system works reliably only within a few meters. This constraint impacts the practicality of the gesture control system, particularly in larger or more expansive environments where greater distances between the operator and the drone may be required. Enhancing the strength of the radio connection is necessary to ensure that the system can function reliably over longer distances. That being said, this method can be translated to a system with a more powerful drone and a longer ranged radio and still function in the same way.

3.4 Overall System Performance

Despite these challenges, the gesture recognition system has proven to be highly functional, with only a few limitations. The accuracy of the gesture detection is good when the hand is oriented correctly, and the system shows robust performance in terms of speed and responsiveness. The gestures themselves are intuitive, with the direction and movement of the hand aligning closely with the desired drone actions, making the system user-friendly and easy to operate.

4. Conclusion

This research demonstrates the potential of hand gesture recognition systems to improve drone control by offering a more intuitive and accessible user interface. Using MediaPipe Hands and custom Python algorithms, the system successfully recognizes hand gestures and translates them into real-time drone commands, providing a seamless and responsive experience. The ability to control drones through natural hand movements opens up new possibilities for more intuitive interactions in various fields, such as search and rescue, entertainment, and human-computer interaction.

While the system shows significant promise, challenges remain in improving accuracy, particularly with hand orientation and camera movement. The current reliance on a fixed camera setup limits the flexibility of the system, and ensuring robust performance with a moving camera remains a key area for future research. Additionally, the communication range between the laptop and drone is currently constrained, highlighting the need for stronger connectivity solutions to support reliable control over greater distances.

Despite these challenges, the speed, responsiveness, and intuitive nature of the system suggest that with further refinement, hand gesture recognition could become a highly effective and user-friendly method for drone control. As advancements in camera technology, gesture recognition algorithms, and communication protocols continue to evolve, this approach has the potential to significantly enhance drone operations, contributing to the development of more versatile and accessible autonomous drone systems in a variety of industries.

5. References

[1] Google, "MediaPipe Hands: Real-time hand tracking," GitHub, 2020. [Online]. Available: https://github.com/google/mediapipe.

[2] Oudah, M., Al-Naji, A., & Chahl, J. (2020). Hand gesture recognition based on computer vision: a review of techniques. *journal of Imaging*, *6*(8), 73.

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

[4] K. Nivi, "Tello-gesture-control: Gesture control for Tello drone," GitHub, 2020. [Online]. Available: https://github.com/kinivi/tello-gesture-control.