Project Proposal: Controlling UAV with ESP32 S3 and IMU Sensor

COMPSCI 528: Mobile and Ubiquitous Computing

Team Members: Noshitha Padma Pratyusha Juttu, Janet Reshma Jeyasingh, Sri Harsha Vardhan Prasad Jella

Overview:

Gesture Recognition has emerged as a crucial research area in health and human-machine interaction, particularly for applications requiring high sensitivity and precision. Unlike static recognition, real-time Gesture Recognition involves the continuous analysis of dynamic gestures, posing unique challenges in accuracy and responsiveness. Machine learning methods have gained attention for addressing these challenges, but many existing solutions struggle to deliver reliable real-time performance.

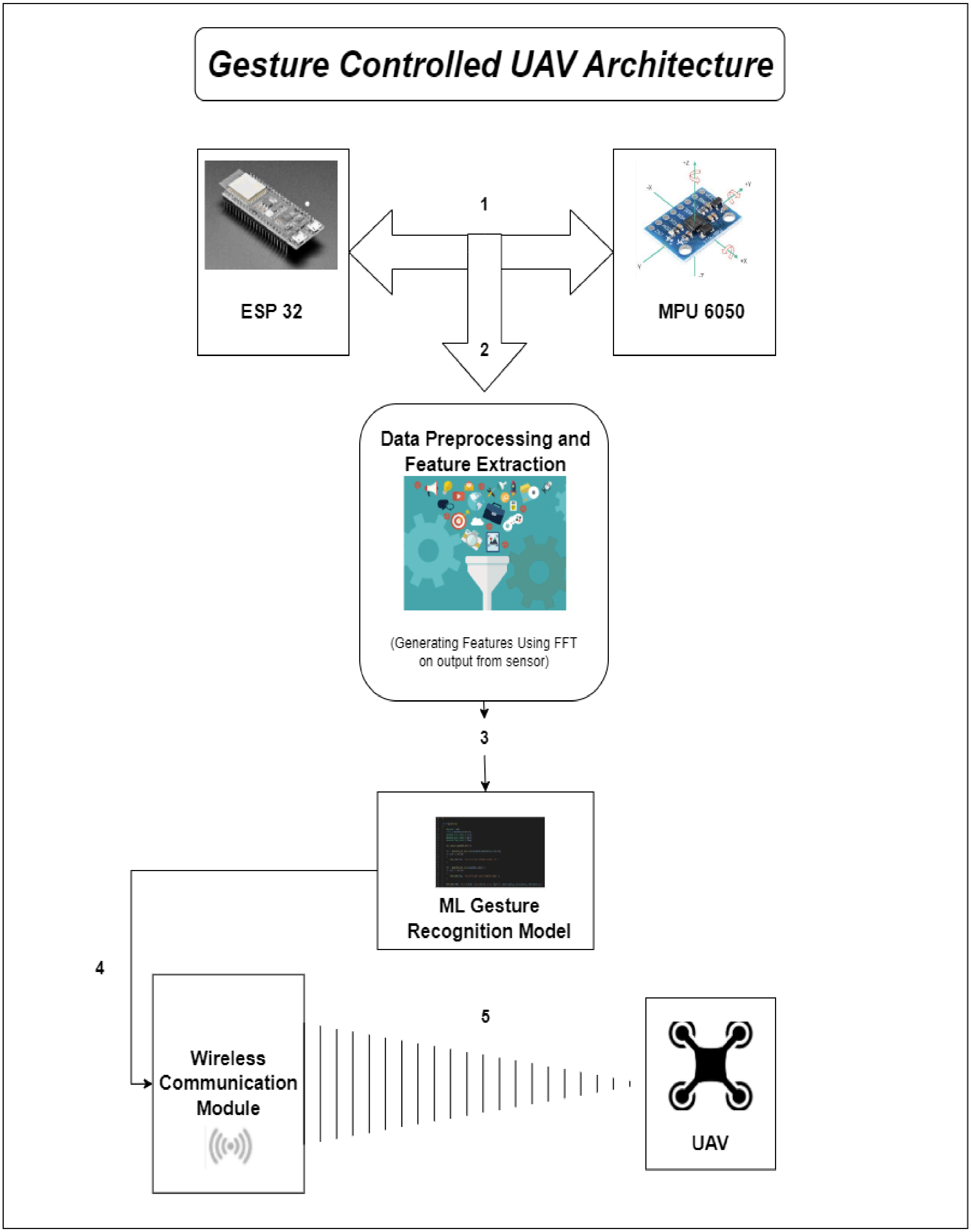
Our project focuses on developing a gesture-controlled UAV (Unmanned Aerial Vehicle) using the ESP32 S3 microcontroller and an MPU6050 IMU (Inertial Measurement Unit) sensor. The MPU6050 sensor will capture motion data from hand gestures, which will be processed and translated into commands for controlling the UAV.

The core idea is to enable real-time UAV control through hand gestures (e.g., UP, DOWN, LEFT, RIGHT). By using the ESP IDF (Espressif IoT Development Framework), the ESP32 S3 will be programmed to interpret the IMU sensor's accelerometer and gyroscope data. This data will be analyzed to recognize specific hand movements, which will then be mapped to flight maneuvers for the UAV.

This project combines embedded systems, mobile computing, and gesture recognition technologies, offering a practical exploration of sensor-based control in the context of UAV navigation.

Key components of the project include:

- **IMU Sensor Integration**: The MPU6050 IMU sensor will be used to capture motion data.
- **Real-time Data Processing**: Collected data will be processed on the ESP32 S3 to recognize gestures. The ESP32 is a powerful SoC microcontroller with Wi-Fi, Bluetooth 4.2, dual cores up to 240 MHz, 36 GPIO pins, 16 PWM channels, and 4MB flash memory, making it a major upgrade from the ESP8266.
- **UAV Control**: Hand gestures will be used to control the direction and movement of the UAV.


Gesture Controlled UAV Architecture
ESP 32
MPU 6050
1
2
Data Preprocessing and Feature Extraction
(Generating Features Using FFT on output from sensor)
3
ML Gesture Recognition Model
4
Wireless Communication Module
5
UAV

Steps Overview:

1. **Collection of Data:**

   The IMU sensor captures the dynamic acceleration and angular velocity of the hand in 3D space while performing gestures. This data is transmitted to the ESP32 microcontroller.

2. **Data Preprocessing and Feature Extraction:**

   The raw sensor data is preprocessed for consistency and reliability. The relevant frequency domain features are extracted from the preprocessed data using Fast Fourier Transformations (FFT).

3. **Feeding the preprocessed data:**

   The preprocessed data is then passed to the gesture classification model for gesture prediction.

4. **Gesture Classification:**

   A machine learning model is trained using the extracted features and corresponding gesture labels. During real-time prediction, the model processes input features and identifies the gesture being performed.

5. **Communication between the Gesture system and UAV:**

   After the gesture is classified, the corresponding control signal is transmitted wirelessly (e.g., via Bluetooth) from the ESP32 to the UAV.

Why is this project exciting for you and your group?

This project excites us for several reasons:

1. **Interdisciplinary Learning**: It brings together various technical domains such as sensor data acquisition, embedded systems programming, wireless communication, and UAV dynamics, offering a rich interdisciplinary learning experience.
2. **Real-world Applications**: Gesture control of UAVs has immense practical potential. Such systems could be used for intuitive human-machine interfaces in drone-based delivery systems, search and rescue operations, or even gaming and entertainment.
3. **Exploring Emerging Technologies**: Working with ESP32 S3 and gesture-based control is a hands-on approach to mastering emerging technologies. Understanding how to capture, interpret, and act upon sensor data using microcontrollers is critical for mobile and ubiquitous computing, which is the essence of this course.

4. **Challenge and Innovation**: This project allows us to push the boundaries of what is possible with low-cost hardware while solving technical challenges related to real-time data processing and UAV control.

What new hardware do you need besides what has already been provided?

Though we have access to the ESP32 S3 and MPU6050 IMU sensor, additional hardware will be required to extend the functionality of our project:

- **Quadcopter Frame**: A small, lightweight bluetooth enabled quadcopter frame to build the UAV prototype and test flight control systems.
- **Brushless Motors and ESC (Electronic Speed Controllers)**: Motors to drive the propellers and ESCs to regulate the speed and direction of the motors based on the commands from the ESP32 S3.
- **Propellers**: Four propellers will be needed to ensure stable flight and maneuverability.
- **Battery Pack**: A high-capacity battery will power the quadcopter and the electronics, ensuring adequate flight time.
- **(Optional)Wireless Communication Module**: If the project scope expands, we may need a Bluetooth or Wi-Fi module to enable remote control from a smartphone or computer in addition to the gesture-based control system.

Team members, and their roles:

These are the tentative roles and responsibilities initially,we collaborate with each other to get the best for every step.

Sri Harsha Vardhan Prasad Jella – Lead Hardware Engineer

- **Role**: Responsible for interfacing the hardware components, including wiring the ESP32 S3 with the IMU sensor and ensuring smooth communication through I2C protocol. This member will also handle the assembly of the UAV prototype by integrating motors, ESCs, and power distribution boards.Once ML model is trained, they will integrate the model with the ESP32 firmware and ensure that the predicted gestures are correctly translated into UAV flight commands.

Janet Reshma Jeyasingh – Embedded Systems Developer

- **Role**: Focuses on programming and firmware development using the ESP-IDF framework. This member will write the code necessary to collect real-time IMU sensor data (accelerometer and gyroscope readings) and ensure proper pre-processing, such as filtering and data cleaning. They will be responsible for passing the processed data to the machine learning model. Additionally, they will program the ESP32 S3 to send control signals to the UAV based on predicted gestures.

Noshitha Juttu – Machine Learning Model Developer

- **Role**: This member is in charge of designing, training, and implementing the machine learning model for gesture recognition. They will pre-process the IMU data (using techniques like FFT or filtering), extract relevant features, and develop a gesture classification model. They will be responsible for training the model, testing it with various gesture inputs, and optimizing it for real-time prediction.

Milestones and deliverables:

The project will be completed in a series of carefully planned Bi-weekly milestones, ensuring incremental progress and timely completion.

Milestone 1 (Week 2):
ESP32 Setup & Sensor Integration
Set up the ESP32 S3 and connect it to the MPU6050 IMU sensor. Collect initial accelerometer and gyroscope data.
Deliverable: Basic hardware setup and data collection code.

Milestone 2 (Week 4):
Gesture Data Collection & Pre-Processing
Collect and label gesture data (UP, DOWN, LEFT, RIGHT). Apply FFT or other preprocessing techniques.
Deliverable: Labeled gesture dataset and initial preprocessing code.

Milestone 3 (Week 6):
Gesture Recognition & UAV Control Mapping
Develop algorithms for real-time gesture recognition and map gestures to UAV movement commands.
Deliverable: Real-time gesture recognition code and initial UAV control.

Milestone 4 (Week 8):
System Integration & Testing
Finalize UAV control, test real-time gesture recognition, and verify the UAV's response in different conditions.
Deliverable: Fully functioning gesture-controlled UAV prototype.

Final Deliverable (End of Term):
Submit final project including:

- **Codebase:** ESP32 firmware, gesture recognition, UAV control.
- **Data:** Labeled gesture data used for training and testing the system.
- **Report:** Comprehensive documentation of project development, results, and future work.

Project's GitHub link:

The project's development will be managed using GitHub for version control and collaboration. Our project repository will be located at:
[**GitHub Repository**]

This repository will contain:

- All the source code for the ESP32 S3 firmware.
- Data collected from the IMU sensor.
- Scripts used for gesture analysis.
- Documentation and final project report.

References:

1. Kim, M., Cho, J., Lee, S., & Jung, Y. (2019). IMU Sensor-Based Hand Gesture Recognition for Human-Machine Interfaces. *Sensors (Basel, Switzerland)*, *19*(18), 3827. https://doi.org/10.3390/s19183827
2. Babiuch, Marek & Foltynek, Petr & Smutný, Pavel. (2019). Using the ESP32 Microcontroller for Data Processing. 1-6. 10.1109/CarpathianCC.2019.8765944.
3. Djemal, Achraf & Hellara, Hiba & Barioul, Rim & Ben Atitallah, Bilel & Ramalingame, Rajarajan & Fricke, Ellen & Kanoun, Olfa. (2022). Real-Time Model for Dynamic Hand Gestures Classification based on Inertial Sensor. 10.1109/CIVEMSA53371.2022.9853648.