iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** V **Month of publication:** May 2026

DOI: https://doi.org/10.22214/ijraset.2026.82981

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

Vision-Guided Quadcopter with Onboard Face Recognition and Autonomous Target Tracking

Mayank Patil[1], Deepraj Patil[2], Harshvardhan Patil[3], Milind Patil[4]
Department of Electronics and Telecommunication, Vishwakarma Institute of Technology, Pune, India

Abstract: This paper presents a low-cost vision-guided quadcopter capable of detecting, locking onto, and autonomously tracking a predefined target while transmitting its GPS coordinates in real time. The system integrates an ESP32-based custom flight controller with a Raspberry Pi 5 vision processing module to implement onboard face recognition and closed-loop visual servoing. Unlike conventional UAV platforms that rely on proprietary autopilot hardware and GPU-based processing systems, the proposed architecture introduces a fully embedded perception–action framework in which visual deviation directly influences flight control. Flight stabilization is achieved using cascaded PID control combined with complementary filter-based sensor fusion using an MPU6050 IMU. Face detection is performed using MediaPipe, and recognition is implemented using the LBPH algorithm. Experimental results demonstrate stable hover within ±3° angular deviation, recognition accuracy of 80–90% at 3–5 m distance, and control latency below 50 ms. The proposed system offers a cost effective and efficient
Keywords: ESP32, UAV, Vision-Based Tracking, Face Recognition, PID Control, GPS Localization, Embedded Systems, Raspberry Pi 5.

I. INTRODUCTION

The Unmanned Aerial Vehicles (UAVs) are growing in significance in surveillance, inspection, disaster management and security surveillance uses. They can be useful in smart monitoring systems due to their ability to be used in hazardous or inaccessible weather conditions. Object detection, recognition, and autonomous tracking tasks can now be performed by UAVs due to the recent improvements in computer vision. Nevertheless, the majority of the commercial drone systems are based on proprietary autopilot components and high-performance processors, which make the systems more expensive, increase power consumption, and complicate their architecture.

Most current UAV systems have flight stabilization and vision processing in separate modules. Although video could be recorded to be used in monitoring, at times, visual information does not serve a direct influence in real-time flight control. This division restricts the use of completely autonomous perception-based navigation. Also, real-time recognition systems often require use of graphics processing units, which do not fit into embedded applications that are low cost and energy efficient.

The proposed research is to create a lightweight vision guided quadcopter which will be able to identify a predefined face, follow it, track the target autonomously to prevent it, and send the GPS position of the target to the user. The system combines an ESP32-based flight controller and a Raspberry Pi 5 vision module in order to design a closed-loop perception action mechanism. Face detection is done in Media Pipe, recognition in LBPH algorithm and flight stability in cascaded PID control with IMU-based sensor fusion. When the target is locked, the GPS coordinates are captured and sent wirelessly to the UAV giving it an intelligent and cost-effective surveillance solution.

II. LITERATURE REVIEW

A number of researchers have been undertaking to perfect UAV stabilization and autonomous navigation. Bouabdallah et al. [1] introduced a quadrotor control system that operates on a PID control and showed that it is possible to maintain stable attitude control when sensor fusion is applied. In their work, they demonstrated that classical control approaches can still be used to stabilize the quadcopter with the right tuning. Their system however was primarily aimed at stabilizing and lacked real time vision based tracking. The UAV tracking based on vision has been studied in different works. Chen et al. [2] constructed an object tracking drone based on convolutional neural networks (CNNs), to detect and follow the objects. Their system was highly accurate in tracking though it needed the use of a processing hardware that was based on GPUs which made it costly and consumed a lot of energy. Equally, Sa et al. [3], applied the vision-based object following algorithm that uses deep learning algorithms on embedded systems with a high level of consistency but with high computation costs. These papers focus on the power of computer vision in UAV tracking but present weaknesses with regards to cost and the complexity of hardware.

Embedded systems like ESP32 and Raspberry Pi have recently become popular as low-cost systems to use in the development of intelligent systems. A study conducted by Ali and Hussain [4] revealed that ESP32 had the capability of doing real-time control tasks and global communication with a wireless connection, thus capable of the internet of things and robotics. Moreover, Bradski [5] demonstrated that algorithms that run on OpenCV can be made highly efficient on low-weight hardware. Such results confirm the possibility of onboard face detection and recognition utilization without the use of costly GPU systems.

Embedded vision systems have heavily made use of face recognition methods like the Local Binary Patterns Histogram (LBPH) method since they are less computationally demanding. Ahonen et al. [6] established that LBPH offers good facial recognition with different lighting conditions and minimum processing power. It has also been demonstrated that media pipe based detection systems can offer efficient face localization in real-time that can be used in mobile and embedded devices [7]. Such techniques can be used to implement face-based tracking in resource-constrained UAV platforms. Localization GPS in UAVs has been researched widely in terms of navigation and surveillance. Kaplan and Hegarty [8] have talked of the application of GPS modules to real-time positioning in mobile systems where the accuracy is limited in open and obstructed setups. Recent studies [9] were able to show the merging of GPS information with UAV tracking platforms to offer geo-tagged coverage of detected targets. Nevertheless, most of these applications base the implementations upon the use of commercial autopilot boards as opposed to low-cost controllers that are self-made. In recent years, vision-based navigation and control of UAVs is studied as well. Kamel et al. [10] introduced a vision-based navigation and control framework of unmanned aerial vehicles and showed the possibility of using visual feedback during flight control in the autonomous regulation of motion. Their contribution focused on the value of perception and control that enhanced autonomy of UAV. Nevertheless, the suggested system was concerned mainly with the navigation and control performance based on the superior frameworks, but it did not take into account the low-cost implementation based on the lightweight embedded platforms. This has led to the necessity to have inexpensive architectures, which combine vision-based tracking with geo-localization in a single embedded platform.

III. SYSTEM ARCHITECTURE

A. Hardware Architecture

The proposed UAV is constructed using a compact quadcopter frame designed for lightweight and manoeuvrable operation, with a total take-off weight of approximately 900 g including onboard components. The propulsion system consists of four 2300 KV brushless DC motors with 5.1 × 5.2 inch tri-blade propellers, providing rapid thrust response suitable for dynamic tracking operations. The total thrust produced exceeds 2.5 times the system weight, resulting in a thrust-to-weight ratio greater than 2.5:1, which enables stable take-off and responsive control during aggressive manoeuvres. The flight controller is implemented using an ESP32 WROOM-32U microcontroller with a 240 MHz dual core processor, while attitude estimation is performed using the MPU6050 inertial measurement unit integrating a 3-axis accelerometer and gyroscope through the I2C bus.

Vision processing is carried out on a Raspberry Pi 5 for real time image analysis. A 5 MP camera module mounted on the UAV captures live video frames for face detection and recognition. Localization is achieved using a NEO-6M GPS module connected via UART to provide latitude and longitude information. The system is powered by a 3-cell 11.1 V 3300 mAh 45C Li-Po battery, providing an average flight time of about 5 minutes during tracking operations. Communication between the Raspberry Pi 5 vision module and the ESP32 flight controller is established through the SPI (Serial Peripheral Interface) protocol, enabling high-speed transmission of target position and tracking error information for real-time flight control adjustments.

B. Software Architecture

Flight control application software is developed on Arduino IDE on the ESP32 platform. The roll, pitch and yaw dynamics are controlled by a cascaded PID control system to ensure the stability of the attitude control during manoeuvring and flight in a hover. The control loop has a sampling rate of approximately 150 Hz and it can reject disturbances quickly and only smooth out, it cannot stabilize. A complementary filter that combines the data of gyroscopes and accelerators is utilized to do sensor fusion to remove the high frequency noise and long term drift of changing orientation. Onboard vision processing was made in Python with the OpenCV libraries on the Raspberry Pi 5. The face detection is done by Media Pipe and Local Binary Patterns Histogram (LBPH) algorithm is used to identify the face to offer the real time performance. The frames per second of the face video are capturing at about 25 frames/second of the video in order to do the processing real time. The recognition confidence threshold is experimentally set to 80 in order to minimize false detections. On a successful recognition, the pixel difference between the centroid of the face occurring (xf, yf) and centroid of the image frame (x c, y c) would be calculated and converted into proportional adjustment controls. These control references are also transmitted to the ESP32 and the PID setpoints are dynamically updated to make sure that the target is always centered and tracked.

IV. METHODOLOGY

A. Theory

1) Flight Control and Stabilization

A cascaded Proportional-Integral-Derivative (PID) control structure applied on the ESP32 microcontroller is used to stabilize the flight. A MPU6050 inertial measurement unit gives both accelerator and gyro sensor data used in real-time orientation determination. In order to get quality attitude information, a complementary filter is used to combine data of gyroscopes and accelerators and thus reduce drift and high frequency noise. Angular velocity is regulated by the inner loop through the control of angular velocity, whereas the outer PID loop controls roll, pitch, and yaw angles. The control loop frequency is over 100 Hz, and this is used to maintain the stable hover level and maneuverability. The resulting computed control signals are then converted to pulse-width modulation (PWM) outputs to control the speed of the electronic speed controllers and ensure a dynamic balance

2) Vision-Based Target Detection and Recognition

Video frames are received by the Raspberry Pi module and the board displays the frames at a rate of about 20-30 frames per second. The Media-Pipe framework is utilized to complete face detection and gives effective and lightweight bounding box localization that is capable of being utilized by embedded systems. Once a face is detected, it is extracted by the Local Binary Patterns Histogram (LBPH) algorithm. The obtained feature vectors are matched with a pre-trained facial database composed of several pictures of the target under varying orientations and lighting conditions so as to enhance the robustness of the recognition. Detection is not made until the confidence value is over a pre-set value on a series of frames, which lowers false detection and enhances accuracy in dynamic scenes. The recognition confidence given by LBPH classifier varies depending on among others quality of the training set, the lighting condition, facial orientation, distance between the camera and the target, and the confidence level set by the classifier. In the experiment, 80% confidence was set empirically to trade off between detecting and false positives. Experimental results also indicate the recognition confidence values of 80-90 percent in both indoor lighting at 3-5 meter distance, the results produced are 89 percent and 84 percent respectively, indicating the strong feature similarity during the real-time tracking.

3) Autonomous Target Locking and Tracking

Once the target has been recognized successfully, the UAV goes into target locking. The centroid points of the face detected are compared with centroid of the picture frame to detect the horizontal and vertical deviations. The deviations are also mapped to flight control changes where horizontal error affects the yaw correction and Vertical changes affect the altitude changes. Also relative distance is estimated by way of changes in the size of the face bounding box, which allows control in the forward or backward direction. These calculated values of deviation are sent to the ESP32 which adjusts the reference inputs to the PID controller respectively. This servoing mechanism is a visual servoing system that functions in a closed-loop to enable the UAV to autonomously follow and keep the target in the center of the frame.

4) GPS-Based Geo-Localization and Reporting

The flight controller has a GPS module that is used to give real time data on latitude and longitude. Upon keeping the target within a predefined range of tolerance over a certain period of time, the system records and captures the geographic coordinates of the UAV. These are the coordinates that are sent out in a package with recognition confirmation and time stamp data and transferred wirelessly to the ground station or user interface. This feature facilitates geo-tagged real-time reporting on the identified target, which increases the applicability of the system in surveillance and monitoring activities

5) Workflow and Safety Mechanisms

There are three major modes of the UAV, namely, manual mode, search mode, and target lock mode. Manual mode The user controls take-off control and positioning control. In search mode, the drone automatically scouts around the area of the target identified under the search mode by using constant vision processes. Upon success of the recognition, the system enters into the target lock mode allowing the tracking and GPS reporting to be turned on. Fail-safe functions also come along to enhance the reliability of the functioning like activation of a search mode in case of target loss, hover when there is a loss of communications and controlled landing when the battery is low. By these safety measures, it is possible to operate safely and continuously in the actual deployment.

B. Mathematical model

1) Dynamic Model of the Quadcopter

The quadcopter is modeled as a rigid body with six degrees of freedom, consisting of translational motion along the ,,$z$axes and rotational motion about roll (ϕ), pitch (θ), and yaw (ψ)axes. The motion of the UAV is governed by the Newton–Euler formulation. The translational dynamics are expressed as

$$m\ddot{r} = RT - mg$$

where m represents the mass of the quadcopter, $\ddot{r}$denotes the linear acceleration vector, R is the rotation matrix transforming body coordinates to the inertial frame, T is the total thrust vector generated by the propellers, and g represents gravitational acceleration. The total thrust produced by the four propellers is given by

$$T = k_f(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2)$$

where $kf$is the thrust coefficient and ωirepresents the angular velocity of the ithmotor. The combined thrust controls the vertical motion and contributes to translational movement.

The rotational dynamics of the quadcopter are described by

$$I\dot{\omega} = \tau - \omega \times (I\omega)$$

where $I$is the inertia matrix of the UAV, ω is the angular velocity vector, and τ represents the torque vector generated by differential motor thrust. These torques control the roll, pitch, and yaw motions of the quadcopter.

2) Pid Control Model

Flight stabilization is achieved using a cascaded PID controller. The control law for each axis is defined as

$$u(t) = K_p e(t) + K_i \int e(t)dt + K_d \frac{de(t)}{dt}$$

The attitude error between desired and measured angles. The proportional term corrects instantaneous error, the integral term compensates accumulated steady-state error, and the derivative term improves system damping and response speed.

3) Sensor Fusion Model

To obtain stable attitude estimation, a complementary filter combines gyroscope and accelerometer measurements. The estimated orientation angle is calculated as

$$\theta_{est} = \alpha(\theta_{prev} + \omega_{gyro}\Delta t) + (1 - \alpha)\theta_{acc}$$

where $\alpha$is the filter weighting coefficient, $\omega gyro$represents angular velocity from the gyroscope, Δt is the sampling interval, and $\theta acc$is the angle derived from accelerometer measurements. This method balances short-term gyroscope accuracy with long-term accelerometer stability.

4) Visual Tracking Error Model

The visual tracking system computes the deviation between the detected face center and the image frame center. The horizontal and vertical tracking errors are defined as

$$e_x = x_f - x_c$$
$$e_y = y_f - y_c$$

where (xf ,yf) represents the detected face centroid and (xc , yc)denotes the image center. These errors are mapped into control inputs for UAV orientation using proportional control

$$u_{yaw} = K_{tx} e_x$$
$$u_{altitude} = K_{ty} e_y$$

where $Ktx$and $Kty$are visual tracking gains. This mapping ensures that the UAV adjusts its yaw and altitude to keep the detected face centered in the image frame.

$$e_d = A_{ref} - A_{current}$$

The relative distance to the target is approximated using the bounding box area variation where $Aref$ represents the reference face area and $Acurrent$ is the currently detected area. This error is used to adjust pitch motion for forward or backward movement.

5) GPS Localization Model

The UAV position is determined using latitude (ϕ) and longitude (λ) obtained from the GPS module. The relative displacement between two positions is approximated by

$$d = R\sqrt{(\Delta\phi)^2 + (\cos\,\phi \cdot \Delta\lambda)^2}$$

where R represents the Earth's radius and $\Delta\phi$,$\Delta\lambda$ denote changes in latitude and longitude in radians. This model enables approximate distance estimation for location tagging of the tracked target.

C. Block Diagram

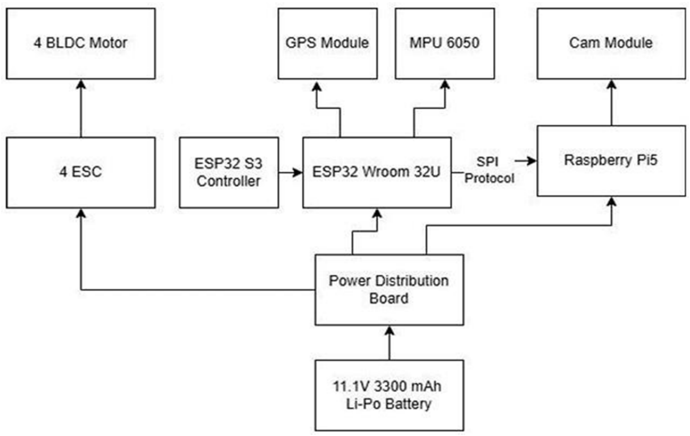

Fig.1 Block Diagram

D. Flowchart

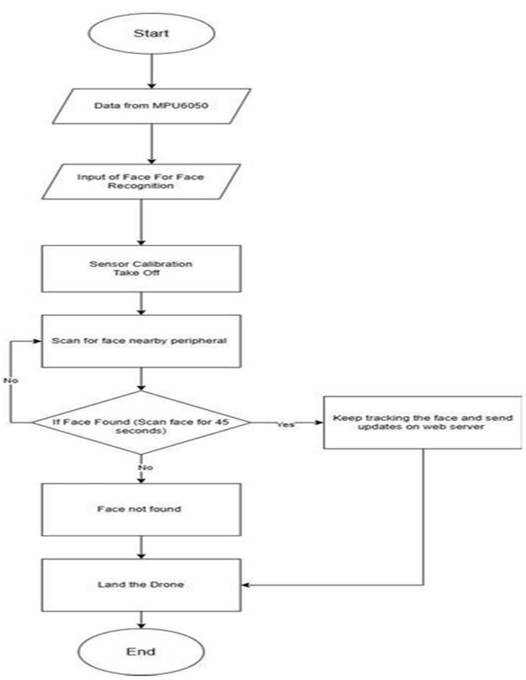

Fig.2 Flowchart

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

V. SIMULATION

A. Circuit Diagram

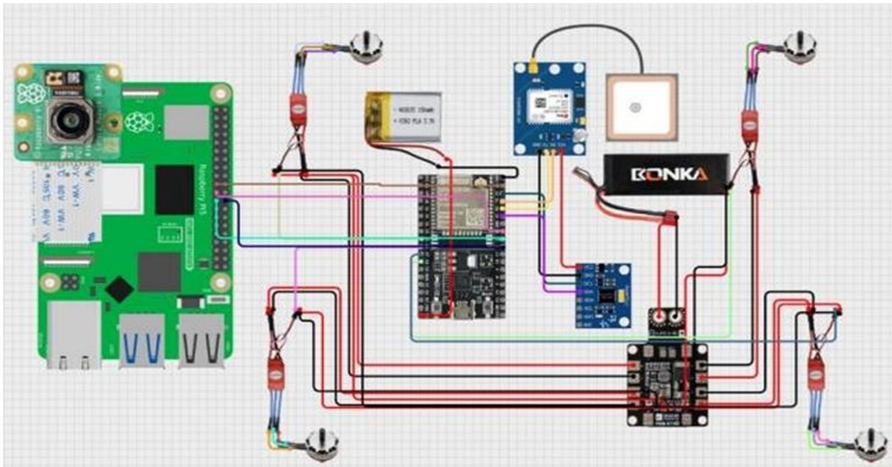

Fig.3 Circuit diagram of the UAV hardware integration.

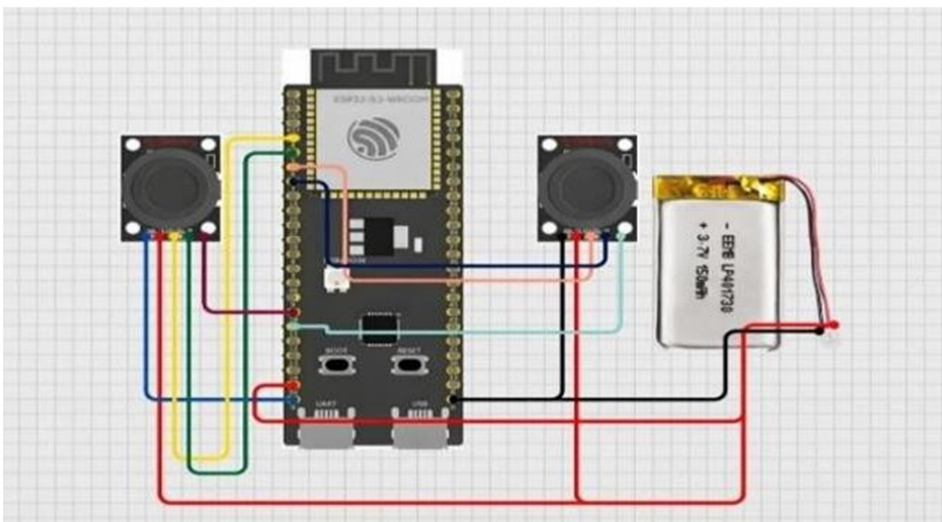

Fig.4 Wireless joystick controller architecture.

Figure 3 illustrates the hardware integration and wireless control architecture of the proposed UAV system. The ESP32 WROOM-32U functions as the main flight controller interfacing with the MPU6050 IMU sensor, NEO-6M GPS module, ESCs, and BLDC motors for stabilization and propulsion control. The Raspberry Pi 5, connected to a camera module, performs real-time vision processing and transmits target tracking information to the ESP32 through the SPI communication protocol. Power is supplied by an 11.1 V Li-Po battery through a power distribution board, which distributes regulated power to all onboard components. The MPU6050 communicates with the ESP32 via the I²C interface for attitude sensing, while the GPS module provides real-time position data through UART communication. The ESCs receive PWM signals from the flight controller to regulate the speed of the BLDC motors, enabling stable flight and autonomous target tracking.

The system operates using a closed-loop perception and control mechanism in which visual feedback from the Raspberry Pi directly influences UAV motion. Real-time tracking errors generated from face detection are continuously transmitted to the ESP32 for dynamic PID-based flight corrections. This integration enables autonomous target locking, smooth maneuverability, and stable hovering during tracking operations. Additionally, the wireless controller shown in Figure 4 is built around an ESP32-S3 microcontroller connected to dual joystick modules for controlling throttle, pitch, roll, and yaw movements. The controller is powered by a 3.7 V lithium-polymer battery and transmits control commands wirelessly to the UAV, enabling real-time manual operation and seamless switching between manual and autonomous flight modes

VI. RESULT AND DISCUSSION

Table I: Experimental Performance Metrics

Parameter	Measured Value	Remarks
Total Takeoff Weight	900 g	Including battery and vision module
Motor Specification	2300 KV BLDC	5.1 × 5.2 inch tri- blade propellers
Thrust-to-Weight Ratio	> 2.5 :1	Ensures stable and responsive flight
Battery Capacity	3300 mAh, 45C	3S Li-Po (11.1 V)
Average Flight Time	5 ~7 minutes	Under tracking conditions
PID Control Frequency	150 Hz	Stable attitude control
Vision Processing Frame Rate	~25 FPS	Real-time detection
Face Detection & Recognition Accuracy	80–90%	At 3–5 m distance LBPH Similarity score
Recognition Confidence Threshold	80%	Experimentally tuned
Control Latency	< 50 ms	Vision-to-control response delay
Angular Stability	±3°	During hover and tracking
Detection Range	3–5 meters	Indoor testing conditions
GPS Module	NEO-6M	UART-based localization
GPS Position Accuracy	±2–3 m	Open-sky conditions
Communication Method	SPI Protocol	ESP32 ↔ Raspberry Pi 5

Table II: Comparison Table

Feature	Propos ed System	CNN-based UAV	Commercial Drone
Cost	Low	High	Very High
GPU Required	No	Yes	Yes
Embedded Closed Loop	Yes	Partial	No
Recognition Accuracy	85–88%	>95%	Proprietary
Power Consumption	Low	High	High

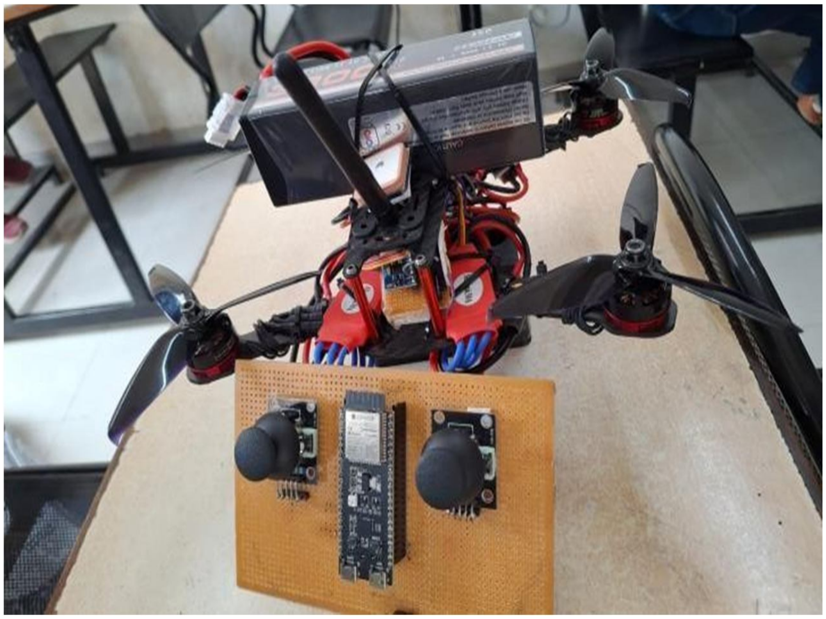

Fig. 5: Experimental UAV prototype used for testing.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

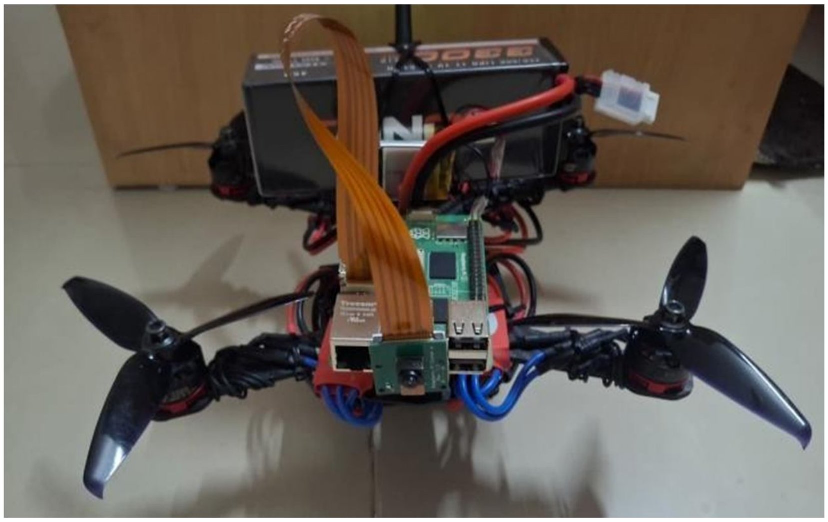

Fig. 6: Hardware integration of the quadcopter system.

Fig. 7: Assembled drone platform with onboard electronics.

Fig. 8: Real-time face detection with 89% recognition

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

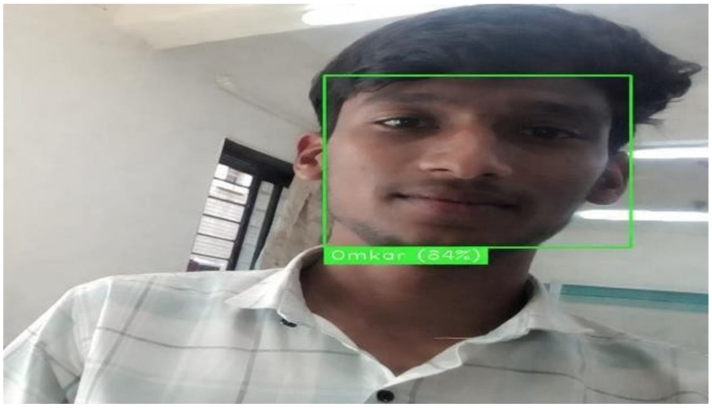

Fig. 9: Face recognition output with 84% during tracking.

The percentages of recognition in Fig. 8 and Fig. 9 are the confidence score produced by LBPH face recognition algorithm during the matching process. This score indicates the resemblance between the identified feature histogram of the face, and the trained facial sample recorded in the system. In the experiment, the recognition threshold was adjusted to 80% and therefore, the faces that give a similarity score higher than this threshold will be accepted as legitimate matches. The values 89% and 84% at the time of testing were gained in the conditions of indoor light that at the distance of 3–5 meters, so the recognition performance is reliable under the conditions of practical working.
There is also experimental evidence of stability in flight when under continuous track operations. The angular stabilization provided by the PID-based controller on the ESP32 provides angular stability within +3 in the unstable state during hovering and target-following operation, whereas the SPI communication between the Raspberry Pi vision module and the ESP32 flight controller provides the ability to manage visual feedback control with low latency. The system also manages to hold the identified face in the middle of the camera, which proves successful closed-loop control of the vision processing and the flight control. The low weight embedded architecture can minimize computational load and power usage than the use of GPUs on the UAV systems, which prove the possibility to provide intelligent tracking functions on small flying vessels.

VII. CONCLUSION

The proposed vision-directed quadcopter incorporates the computer vision system into its embedded flight controller so that it can track and detect faces autonomously. The system consists of a camera module that is linked to a Raspberry Pi that performs real-time face detection and recognition via LBPH algorithm and the ESP32 microcontroller ensures flight stability via sensor feedback of the MPU6050 and GPS module. A control mechanism which is based on a PID control allows the motors to be controlled by Electronic Speed Controllers (ESCs), so that the motors will provide a stable hover and responsive navigation. It has been experimentally demonstrated that the system can achieve a present frame rate of, on average, 25 FPS and a face recognition accuracy of, on average, 80 90 percent with a short range of detection of 3 to 5 meters. The proposed architecture offers an efficient and cheap solution to intelligent UAV surveillance and autonomous tracking applications.

VIII. FUTURE SCOPE

It is also possible to improve the suggested UAV system with the help of sophisticated deep learning-based recognition models that will raise the detection precision in diverse lighting and dynamic settings, and hardware acceleration to help to perform quick real-time calculations and track multiple targets. LiDAR and ultrasonic obstacle sensing or stereo vision can be added to allow full autonomous operation in complex environments and SLAM techniques can be used to allow fully autonomous operation in GPS-denied regions. Endurance of flights can be increased by enhancement of propulsion optimization and smart management of power, and adaptive or self-tuning PID controllers can increase stability in different payloads. Besides, remote monitoring, data analytics, and secure encrypted communication using cloud integration can increase its usage in surveillance, disaster management, and smart security systems.

REFERENCES

[1] S. Bouabdallah, P. Murrieri, and R. Siegwart, “Design and control of an indoor micro quadrotor,” Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2004, pp. 4393–4398.
[2] Y. Chen, Z. Wang, and Y. Qiao, “Real-time object tracking for UAV using deep learning,” IEEE Access, vol. 8,
[3] pp. 124321–124332, 2020.
[4] I. Sa, Z. Ge, F. Dayoub, B. Upcroft, T. Perez, and C. McCool, “DeepFruits: A fruit detection system using deep neural networks,” Sensors, vol. 16, no. 8, pp. 1222, 2016.
[5] T. Ali and R. Hussain, “Integration of ESP32 with sensors for IoT-based real-time control applications,” IEEE Access, vol. 9, pp. 103421–103430, 2021.
[6] G. Bradski, “The OpenCV library,” Dr. Dobb’s Journal of Software Tools, 2000.
[7] T. Ahonen, A. Hadid, and M. Pietikäinen, “Face description with local binary patterns: Application to face recognition,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 28, no. 12, pp. 2037–2041, 2006.
[8] C. Lugaresi et al., “MediaPipe: A framework for building perception pipelines,” arXiv preprint arXiv:1906.08172, 2019.
[9] E. Kaplan and C. Hegarty, Understanding GPS: Principles and Applications, 2nd ed. Norwood, MA, USA: Artech House, 2005.
[10] R. Patel and D. Singh, “GPS-based localization and tracking in unmanned aerial vehicles,” International Journal of Advanced Robotic Systems, vol. 17, no. 4, 2020.
[11] M. R. Kamel, T. Stastny, K. Alexis, and R. Siegwart, “Vision-based navigation and control of unmanned aerial vehicles,” IEEE Robotics and Automation Letters, vol. 2, no. 3, pp. 1213–1220, 2017.
[12] R. Mahony, V. Kumar, and P. Corke, “Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor,” IEEE Robotics & Automation Magazine, vol. 19, no. 3, pp. 20–32, Sept. 2012.
[13] T. Madani and A. Benallegue, “Backstepping control for a quadrotor helicopter,” in Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS), 2006, pp. 3255–32