Hybrid Fuzzy–PPO Control for Precision UAV Spraying

Ahmad B. Alkhodre[1], Adnan Ahmed Abi Sen[2], Yazed Alsaawy[3], Nour Mahmoud Bahbouh[4], Moahmmad Benaida[5]
Faculty of Computer and Information Systems, Islamic University of Madinah, Saudi Arabia [1, 3, 5]
Dr. Hussein ElSayyed Research Center-Department of Graduate Studies & Scientific Research, University of Prince Mugrin, Madinah 42241, Saudi Arabia[2]
Department of Information and Communication Sciences, University of Granada, 18071 Granada, Spain[4]

***Abstract*—Precision agriculture increasingly relies on autonomous UAVs for tasks, such as crop monitoring and targeted pesticide spraying. However, maintaining stable flight and precise spray delivery under varying payloads and wind disturbances remains challenging. This paper proposes a hybrid control architecture that combines interpretable Mamdani fuzzy logic controllers with a deep reinforcement learning (DRL) agent (Proximal Policy Optimization, PPO). The fuzzy controllers encode expert-crafted rules for baseline altitude and attitude stabilization, while the PPO agent adaptively adjusts setpoints to optimize spray coverage and energy efficiency. We train the agent in a realistic PyBullet simulator with dynamic payload and wind conditions. In simulated precision-spraying trials, our hybrid controller outperformed both a conventional PID-based controller and a pure PPO controller. Specifically, it achieved roughly 2–3× faster disturbance rejection, near-zero overshoot, and ~30% faster settling than the baselines, resulting in more uniform coverage and reduced pesticide use. These results demonstrate that fusing fuzzy logic with deep PPO yields a UAV spray controller that is both high-performance and robust for precision agriculture applications.**

***Keywords*—UAVs; precision agriculture; UAV spraying; fuzzy logic control; reinforcement learning; Proximal Policy Optimization (PPO); hybrid control**

I. INTRODUCTION

Unmanned Aerial Vehicles (UAVs) have become vital tools in precision agriculture for tasks such as crop monitoring, health assessment, and targeted spraying. Compared to ground-based systems or manned spraying, UAVs can achieve more uniform coverage and reduced chemical usage [1, 2]. However, realizing fully autonomous spraying requires robust flight control: the UAV must maintain accurate altitude and attitude while compensating for external disturbances (e.g., wind gusts) and internal changes (e.g. decreasing payload mass as pesticide is dispensed) [3]. Traditional UAV controllers use cascaded proportional–integral–derivative (PID) loops for altitude and attitude stabilization. While PID controllers perform well under nominal conditions, a fixed-gain PID tuned for one weight or calm wind conditions can exhibit large errors, overshoot, or oscillations if the payload changes or a sudden gust occurs [3]. These limitations motivate more adaptive, nonlinear control strategies.

Nonlinearities without an explicit plant model. Mamdani-type fuzzy controllers have been successfully applied to quadcopter attitude and altitude control. Fuzzy controllers often achieve faster response and minimal steady-state error in certain scenarios, thanks to their ability to encode heuristic rules (e.g., "if altitude error is positive and increasing, then apply strong downward thrust") [16]. Similarly, researchers used Takagi–Sugeno fuzzy models to reduce controller complexity and still achieve accurate tracking [16]. Overall, fuzzy logic offers interpretability and robustness, which is valuable for safety-critical UAV tasks. However, pure fuzzy controllers do not learn or adapt online, and designing membership functions and rule bases typically requires manual tuning.

Deep Reinforcement Learning (DRL) showed promise for UAV control by learning policies through trial-and-error in simulation [3, 4]. Modern policy-gradient algorithms such as PPO [13] can train controllers that adapt to complex dynamics. Other works combined PID with DRL to achieve fast stability: Wu et al. used a corrective-feedback DRL scheme for UAV landing and mixed a PID baseline with learned adjustments, and Ma et al. applied deep RL to reject wind disturbances, demonstrating improved stability in level-5 wind conditions [5, 6]. Nevertheless, DRL requires extensive data and yields black-box policies, raising safety and interpretability concerns.

To leverage the strengths of both approaches, hybrid fuzzy–RL architectures have been proposed. A common scheme is to run a fuzzy (or PID) controller in the inner loop for basic stability, while an RL agent provides high-level setpoints or bias corrections. For example, [14] introduced a Fuzzy-PPO controller for a guided vehicle, where the PPO agent augmented a fuzzy system and reduced tracking path errors compared to a fuzzy-based controller (average error from 0.05 m to 0.02 m). These studies suggest that integrating fuzzy rules can accelerate learning and improve safety; however, to our knowledge, none have specifically addressed precision UAV spraying with a Mamdani fuzzy–PPO scheme.

In this paper, we apply the hybrid fuzzy–RL concept to precision UAV spraying. We design a Mamdani fuzzy control system for altitude, pitch, and roll stabilization, and integrate it with an adaptive learning (PPO) agent. The fuzzy controllers provide baseline stability via intuitive rules, while the agent adjusts setpoints to optimize spray accuracy and energy. We train and evaluate our architecture in a detailed PyBullet simulation with wind and payload variation [15, 35].

In this work, we introduce a novel hybrid Mamdani fuzzy–PPO controller tailored for a quadrotor pesticide sprayer. Our contributions are:

- Hybrid control design: We develop Mamdani fuzzy controllers for altitude, pitch, and roll stabilization and integrate them with a PPO agent. The fuzzy layer ensures immediate stability through intuitive rules, while the PPO agent supplies adaptive setpoint adjustments. This hierarchical design combines expert knowledge and learning-based adaptability in a unified framework.
- Reward shaping for spray tasks: We introduce reward functions that balance tracking precision with control effort and spray uniformity. By penalizing overshoot and incentivizing steady flight, the shaped rewards accelerate RL training and lead to efficient spray trajectories.
- Comprehensive evaluation: We conduct extensive simulations comparing our hybrid controller against strong baselines: a well-tuned PID controller, a standalone PPO agent, and (qualitatively) advanced adaptive baselines. The hybrid controller consistently outperforms all alternatives: it rejects disturbances roughly 2–3× faster and settles ~30% quicker, with minimal overshoot and lower energy use (see Table IV).
- Operational benefits: We quantify practical gains in spraying scenarios: the hybrid system achieves more uniform chemical coverage, reduces energy consumption, and maintains stability under variable load and wind. These improvements translate into cost savings and sustainability in agricultural operations (less wasted pesticide, longer flight time).

Overall, our results demonstrate that combining an interpretable fuzzy baseline with deep PPO yields a UAV controller that is both accurate and robust for precision agriculture spraying [46].

The rest of this paper is organized as follows: Section II presents the literature review, Section III details our methodology (system design, fuzzy logic control, PPO integration), Section IV presents experimental results and comparisons, and Section V concludes with discussion of implications and future work.

II. Literature Review

We classified the previous studies related to our subject into five categories, namely:

A. UAVs in Precision Agriculture

UAV platforms have transformed agricultural practices by providing flexible, on-demand aerial applications. Recent surveys highlight the wide range of precision farming uses for drones, including crop monitoring, variable-rate application, and targeted spraying [2]. Delavarpour et al. (2023) review 213 sources on UAV sprayers and emphasize that UAVs can deliver chemicals more precisely than manned aircraft, though autonomy and control remain key gaps [2]. RL has also been integrated with perception for precision-ag UAVs [39], and deep RL has been applied to agricultural path planning [8, 26]. Spray accuracy is particularly sensitive to wind: field studies report that even moderate winds can carry droplets off-target, reducing efficacy [19]. These agricultural factors confirm the need for accurate flight control. To maximize spray (coverage) uniformity, the UAV must maintain steady height and heading so the spray path aligns with the crop rows, even as payload mass varies. Thus, well-regulated flight attitude improves coverage, saves energy, and ensures safety [9, 10]. In summary, the precision agriculture domain sets hard demands: controllers must handle nonlinear dynamics, time-varying payloads, and external disturbances, all while optimizing coverage efficiency.

B. Classical Control: PID and Variants

Traditionally, UAV flight control systems employ cascaded PID loops for altitude and attitude stabilization. These controllers are simple and effective in nominal conditions [21, 22, 23]. However, PID has notable limitations. First, it assumes a linear response around an operating point. Quadrotor dynamics are nonlinear and under-actuated (four inputs for six degrees of freedom), leading to cross-axis couplings that a separate PID per axis cannot fully address [16]. Second, PID parameters are usually tuned for a particular mass and airframe configuration. In practice, as payload weight decreases (when spray is used up) or when external forces change, the fixed gains no longer yield optimal performance. As Koch et al. [3] remark, under unknown wind or payload changes, "a PID controller can be far from optimal." This motivates exploration of nonlinear and learning-based alternatives [45, 32, 38, 33].

C. Fuzzy Logic Control for UAVs

Fuzzy logic controllers (FLCs) have been widely studied for UAV stability control because they can capture heuristic expert knowledge and handle nonlinearities without an explicit model [16]. A fuzzy controller defines a set of IF–THEN rules. In a Mamdani FLC, the output of each rule is a fuzzy set, which is defuzzified to produce a crisp control action. The rule base can be constructed manually based on physical insight or tuned via data. For quadrotor control, Mamdani FLCs have been applied to altitude and attitude loops. Studies show that a fuzzy PD controller can outperform a conventional PID (faster rise time and smaller overshoot) [12, 16, 40, 43]. Moreover, some researchers proposed adaptive fuzzy schemes to handle disturbances [25, 28]. For example, Coza et al. developed an adaptive-fuzzy controller that adjusts membership centers online to account for wind disturbances, and maintain stability without chattering [30]. Robust and adaptive fuzzy trajectory tracking further mitigates disturbances [44], including advanced T–S fuzzy designs with LMI tuning [41].

Overall, fuzzy controllers offer robustness to uncertainty and do not require accurate system identification, and can handle variable load effects and average wind by adjusting control. However, they do not inherently adapt online; designing effective membership functions and rule bases still requires expert knowledge or formal methods [16, 37]. Additional hovering control studies corroborate these gains [42].

D. Deep Reinforcement Learning for UAV Control

Deep reinforcement learning (DRL) has emerged as a powerful tool for UAV control by leveraging neural networks to learn complex policies [36]. Early works applied DRL to high-level tasks (e.g. navigation), but more recent research has moved

RL into the inner loop. Koch et al. showed that PPO produced smoother control signals and lower roll/yaw error than PID in an inner-loop attitude task, supporting its suitability for high-precision control [47]. Subsequent studies applied DRL to various UAV tasks. Wu et al. (2022) [5] used a DDPG-based agent with corrective feedback to land a UAV on a moving platform; by blending a PID baseline with learned adjustments, they achieved precise landings under dynamic conditions. Hybridizing RL with Sliding-Mode control also reduced chattering in attitude control [27]. Ma et al. (2024) trained a deep RL policy specifically to reject wind disturbances, maintaining level flight in gusts that destabilized fixed PID [6]. In precision spraying missions, hierarchical action-masked PPO improved navigation and spray efficiency [31], and conditional-action-tree RL achieved similar gains in agricultural tasks [24]. Complementarily, RL has also been used to tune classical PD gains online with successful real-flight tests [34]. Despite these advances, pure RL policies remain data-hungry and less transparent, raising safety and interpretability concerns.

E. Hybrid Fuzzy-RL Architectures

Motivated by the complementary strengths of fuzzy logic and DRL, recent research has proposed hybrid controllers. The key idea is to retain a rule-based fuzzy or classical core for stability and let a learning agent tweak setpoints or feedforward bias for performance and adaptability. One general strategy is a cascaded architecture, where the fuzzy (or PID) controller runs in the inner loop to provide safe control, and the RL agent acts as an outer-loop "bias" or feed-forward adjustment [48]. This way, the RL agent does not have to learn basic stabilization from scratch, speeding up training and improving reliability. Kuo et al. (2025) [14] developed a Fuzzy PPO (FPPO) controller for an autonomous guided vehicle (AGV) path tracking task. Xia et al. (2024) [11] propose a different hybrid: they combine a Soft Actor-Critic (SAC) DRL agent with a Fuzzy Inference System (FIS) for UAV target interception [49]. The SAC–FIS agent splits control among subsystems using fuzzy logic for attitude commands and deep learning for navigation; the fuzzy component provides "universal experiences" (prior knowledge) that reduce training time and control cost. Their results support the idea that fuzzy rules accelerate learning by narrowing the policy search.

In summary, while these efforts are promising, to our knowledge none have specifically addressed precision UAV spraying with a hybrid Mamdani fuzzy–PPO scheme. As summarized in Table I, fuzzy controllers and DRL controllers each bring complementary advantages—fuzzy logic yields robustness and interpretability, whereas DRL offers adaptability and self-tuning—while hybrid approaches attempt to integrate these benefits. Our work extends this line by focusing on a spraying UAV: we propose the first architecture that merges Mamdani fuzzy control with a PPO agent for precision agricultural spraying, and we provide a detailed simulation study demonstrating the resulting performance gains. The next sections describe our methodology and experiments.

TABLE I COMPARATIVE SUMMARY OF UAV CONTROL APPROACHES (FIXED-GAIN PID, FUZZY LOGIC CONTROL, DRL, AND HYBRID FUZZY–RL): KEY STRENGTHS, WEAKNESSES

Approach	Strengths	Weaknesses	Representative Works
Fixed-Gain PID	Simple design, proven stability	Performance degrades under payload/wind changes	[20]
Fuzzy Logic Control	Handles nonlinearity, robust to uncertainty	Requires expert rule tuning, no automatic adaptation	[25] [30]
DRL (PPO/DDPG, etc.)	Adaptive, model-free, can optimize performance	Data-hungry, opaque policies, safety concerns	[3], [5]
Hybrid Fuzzy–RL	Combines interpretability with adaptivity	Higher design complexity, few prior examples	[14], [11]

III. METHODOLOGY

In this section, we present the design of the proposed hybrid control system and explain its key components. We also discuss the mathematical foundations behind both the fuzzy logic framework and the reinforcement learning approach that drive its performance.

A. System Design

The proposed UAV control system consists of a hierarchical architecture where a high-level learning agent provides adaptive setpoints that are tracked by low-level fuzzy controllers. Fig. 1 shows a schematic of the hybrid control system, which includes the UAV plant, sensors, PPO RL agent, fuzzy inference controllers, and the environmental disturbance inputs. In this architecture, the UAVs provide sensors (e.g., IMU, altimeter, GPS, wind sensor) that provide state measurements such as altitude, roll, pitch, yaw angles, angular rates, and battery/throttle status. These sensor data are fed into two parallel control modules: 1) the RL agent (PPO policy network) and 2) the fuzzy logic controllers. The RL agent observes the state and environmental context (like current wind speed/direction and remaining payload) as its input. At each time step (e.g. 10 Hz control loop), the PPO policy computes a high-level action – in our design, this action is a setpoint or adjustment for the desired flight state. Specifically, the PPO outputs a target altitude and target pitch/roll angles (or equivalently, target horizontal velocities) that it deems optimal for the current conditions. These serve as dynamic references that can change in anticipation of or reaction to disturbances.

The PPO agent's action vector is given as:

$$a_k \in [-1,1]^4 \tag{1}$$

This is rescaled to:

$$\overline{a_k} = \frac{a_k+1}{2} \in [0,1]^4 \tag{2}$$

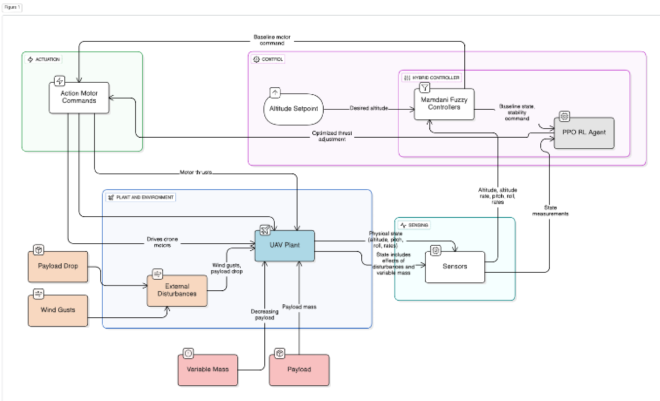


Fig. 1. A schematic of the hybrid control system.

This normalized output defines the base thrust per rotor:

$$T_k^{\text{base}} = \begin{bmatrix} \bar{a}_{1,k} \\ \bar{a}_{2,k} \\ \bar{a}_{3,k} \\ \bar{a}_{4,k} \end{bmatrix} \times \left(F_{max}(m_k) \right) \tag{3}$$

where, $F_{max}(m_k) = 1.5\,\frac{m_k g}{4}$ and the total mass is: $m_k = m_d + m_{p,k} + m_r^{\text{tot}}$

Meanwhile, the Mamdani fuzzy controllers operate at the low level for each control channel:

- Altitude fuzzy controller: Takes the error between the current altitude and the RL's target altitude, and the vertical speed (rate of altitude change), as inputs. It outputs a throttle adjustment command to the motors (collective thrust) to minimize altitude error [17].
- Pitch and roll fuzzy controllers: Work similarly, using pitch/roll errors and angular rates to output control torques or motor differentials.

The fuzzy inference system evaluates $e_{z,k}$ and $v_{z,k}$ using membership functions and a 3×3 rule base with sets NL, Z, PL. The membership functions for each set are defined as shown above. The defuzzification uses the centroid method (see Section "Fuzzy Logic Control" for details). The final total thrust per rotor is computed as:

$$T_{i,k} = T_{i,k}^{\text{base}} + \Delta T_k\, F_{max}(m_k), \quad i = 1, \ldots, 4. \tag{4}$$

Each fuzzy controller uses a defined Mamdani-type rule base to stabilize flight dynamics based on altitude and attitude errors. The PPO agent provides the desired setpoints, while the fuzzy controllers track them with smooth corrective actions. Section "Fuzzy Logic Control" showed in detail the design of fuzzy variables, rules, and defuzzification methods.

The integration of PPO and fuzzy logic happens through these setpoints: the RL agent does not directly control motor outputs (which could be risky), but rather adjusts the "goals" (flight setpoints) that the fuzzy controllers then smoothly enforce.

Note: When spraying ends, we disable fuzzy thrust correction by setting. In this case, PPO fully controls base thrust.

At each step, we advance the state via a discrete-time map that applies the vehicle dynamics, payload mass update, fuzzy thrust correction, and spray-count logic. These are computed using:

- Base thrust per rotor:

$$T_{i,k}^{\text{base}} = \overline{a_{i,k}}\, F_{max}(m_k) \tag{5}$$

- Fuzzy correction (if active):

$$T_{i,k} = T_{i,k}^{\text{base}} \Delta T_k . F_{max}(m_k) \tag{6}$$

- Payload mass update:

$$m_{p,k+1} = \max\,(m_{p,k} - \Delta m, 0) \tag{7}$$

- Spray counter update:

$$c_{k+1} = c_k + 1 \; (when\ spraying\ is\ activate)$$

During a control step, the sequence is:

- Sensors update the state
- RL agent produces new target setpoints
- Fuzzy controllers generate actuator commands
- UAV state updates under physics + disturbances
- Repeat
- Mark waypoint done when $c_{k+1} \geq N\_min$, then reset counter.

Additionally, the PPO agent runs within this loop during training, collecting state transitions and rewards. Termination conditions are:

$$|\emptyset_k| > \emptyset_{max}\ or\ k > \max\ \_steps \tag{8}$$

1) Hybrid control operation: The fuzzy controllers are always active to correct fast disturbances and maintain stability. The PPO agent operates at a slightly lower frequency or can be event-driven (e.g., updating setpoints once every few control loops) to avoid excessive oscillation of references. In practice, one can think of the RL agent as a supervisory controller adjusting the "bias" or "trim" of the flight control system [7].

To represent this behavior mathematically, the final altitude setpoint received by the fuzzy controller may be described as:

$$z_{ref,k} = z_{nominal} + \Delta h_k \tag{9}$$

where, Δh_k is the PPO-learned offset.

The UAV's flight computer hosts both the fuzzy control rules (lightweight) and the PPO neural network (or a post-training lookup table). This design allows robust, explainable, and adaptive flight control that can gracefully handle payload variation, wind gusts, and nonlinearities without sacrificing safety.

Actually, there are five mathematical variables that were integrated into the proposed system, which are (Total Mass and Payload Mass Update, Translational Dynamics, Rotational Dynamics, Current State, and Action).

B. Fuzzy Logic Control

We designed three separate Mamdani fuzzy inference controllers to control the UAV's altitude, pitch, and roll. Each controller uses two inputs (error and rate-of-error) and one output (control correction), Each input and output variable is described by a set of fuzzy linguistic terms.

- Altitude controller: Input 1 = Altitude error = (desired altitude – current altitude) in meters. Input 2 = Altitude error change (approximately the negative vertical velocity) in m/s. Output = Throttle adjustment (a percentage or PWM increment to motor throttle).
- Pitch controller: Input 1 = Pitch angle error = (desired pitch – current pitch) in degrees. Input 2 = Pitch rate (angular velocity about the y-axis) in deg/s. Output = Pitch control (motor speed differential front vs back, or a commanded tilt torque).
- Roll controller: Input 1 = Roll angle error = (desired roll – current roll) in degrees. Input 2 = Roll rate in deg/s. Output = Roll control (motor differential left vs right).

Each rule's activation is $\min(\mu_{error}, \mu_{rate})$ based on its input memberships.

The outputs of defuzzification are then scaled appropriately and sent to the UAV actuators. For altitude, the fuzzy output might be interpreted as an increment or decrement to the base throttle needed to hover. For pitch/roll, the output can be mapped to a commanded angle or directly to motor speed differences. In our simulation, we translate the pitch fuzzy output to an equivalent torque by proportionally increasing or decreasing the front vs back rotor speeds (and similarly for roll left vs right speeds). We apply centroid defuzzification to compute the thrust adjustment from the aggregated fuzzy set.

1) Controller tuning: The fuzzy controllers have many parameters (membership shape, rule consequents), but they were tuned based on intuition and some manual trial in simulation. A key advantage is that unlike PID gains that might need retuning if weight changes, the fuzzy logic inherently handles moderate changes: for example, if the drone gets lighter (so same throttle yields more acceleration), the altitude fuzzy will see a greater error change for a given throttle, and its rules (which depend on error change) will naturally adjust output sooner to avoid overshoot. This contributes to robustness against payload variation. In Section "Results," we will see how the fuzzy controllers perform on their own and in concert with the agent. Overall, the fuzzy controllers guarantee that for small set point changes, the system responds in a well-damped manner with minimal overshoot. They enforce constraints like avoiding excessive tilt or throttle, since rules can be bounded. They also provide explainability: each rule can be understood in plain language, which is important for operator trust in an agricultural setting. Next, we describe how the PPO reinforcement learning agent is designed and how it interfaces with these fuzzy controllers.

C. Learning Framework and Control Integration

The PPO agent is trained in simulation using randomized environmental conditions (wind, payload, pose) to encourage policy generalization. Each training episode simulates a full mission. The agent receives continuous observations and outputs set point adjustments, and the fuzzy controller layer ensures stability, while the agent learns strategic adjustments to optimize reward.

As the episode progresses, it receives state observations — such as position, velocity, orientation, and payload — and responds by outputting actions, which modify flight set points (like pitch, altitude, or direction). These actions do not directly command the motors. Instead, a layer of fuzzy logic controllers that translate high-level decisions into stable control commands interprets them. This layered architecture keeps the drone safe and airborne during early training, when the agent's behavior is still exploratory and potentially erratic. The environment [18] is full of random wind gusts, forcing the agent to learn disturbance rejection. Over time, it begins to uncover useful patterns.

For example, when a strong crosswind blows from the left, the agent learns to tilt the drone slightly into the wind — much like a human pilot would — improving the accuracy of spray deposition, and these behaviors lead to higher rewards, which reinforce the learned policy. After training advances over thousands of incidents, the agent learns improved strategies, such as flying lower in high wind, reducing forward speed in small areas, and adjusting points based on observed wind patterns. The fuzzy controller layer continues to handle fine-grained stability, while the PPO agent develops adaptive, high-level decision-making.

After training, the PPO policy is deployed on the actual UAV system. During real-time flight, the policy receives continuous sensor data — including position, velocity, wind estimation, and payload — and outputs set point adjustments accordingly. Meanwhile, the fuzzy controllers remain active and serve as a fast-reacting stability layer. For example, if a sudden gust causes the drone to drift, the fuzzy controller behaves directly to maintain level flight. At the same time, the PPO agent detects the longer-term wind pattern and may adjust the roll angle slightly into the wind, gradually bringing the drone back on target.

This hybrid architecture works much like an autopilot system that is paired with an adaptive layer. The fuzzy logic ensures short-term flight stability, while the PPO agent adds adaptability and long-term strategic planning. Together, they allow the drone to function robustly — even under environmental conditions that were not explicitly seen during training — ensuring accurate spraying and safe flight behavior.

IV. Experiments

To evaluate the proposed hybrid Adaptive Learning Agent and Fuzzy controller, we made a series of high-accuracy simulations across a range of mission scenarios and compared its performance against two recognized baselines:

PID controller: We implemented a classical PID architecture for altitude, pitch, and roll. Gains were manually tuned using a Ziegler–Nichols-style approach followed by empirical refinement to yield reasonable hover and stabilization performance. The altitude loop combined feedforward hover thrust with feedback on altitude error, while pitch and roll stabilization used independent PD loops typical of cascaded flight controllers.

PPO-only controller: A reinforcement learning agent trained using PPO, without any fuzzy logic components, where this agent shared the same observation space and training conditions as the hybrid model but produced direct thrust commands for the four rotors, and served as a pure learning-based controller to benchmark against conventional and hybrid approaches. To ensure fair comparison, all PPO agents (standalone and hybrid) were trained using the same configuration. The hyperparameters are summarized in Table II.

TABLE II KEY PPO HYPERPARAMETERS

Parameter	Value
Policy / Value net layers	2 × 64 ReLU
Learning rate	3×10^{-4}
Discount factor (γ)	0.995
GAE λ	0.97
Clip ratio	0.2
Rollout length	2048 steps
Mini-batch size	256
Epochs per update	10
Reward normalization	Yes (running mean/σ)

A. Experimental Design

All controllers were tested on a series of mission profiles designed to assess robustness under varied flight conditions. Each scenario involved a step input to a 5 m hover altitude, with disturbances introduced to assess recovery capabilities. Table III summarizes the experimental conditions.

TABLE III EXPERIMENTAL SCENARIO MATRIX

Scenario ID	Mission Type	Initial Payload	Wind Profile	Episode Length (s)	# Runs
S-H	Hover (baseline)	100% (1.0 kg)	Calm (≤ 0.5 m/s)	6	30
S-G5	Hover + gust	100%	Lateral gust 5 m/s @ t = 3 s (0.1 s impulse)	6	20
S-P+15	Hover (heavy payload)	**115%**	Calm	6	20
S-W10	Hover + steady wind	100%	Constant 10 m/s cross-wind	6	20

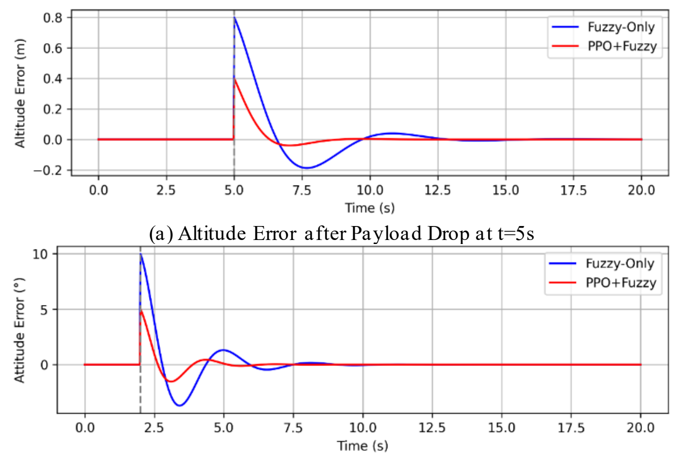
Controllers were assessed on overshoot, settling time, RMS error, drift compensation, and energy usage. Table IV provides a summary of baseline performance under scenario S-H.

TABLE IV AGGREGATED PERFORMANCE METRICS (SCENARIO S-H)

Controller	Overshoot (% of 5 m step)	Settling Time (s, ±5 %)	Steady-State RMS Altitude Error (m)	Energy Usage (% PID)	Spray Accuracy (% coverage)
PID	14.0 %	4.5	0.20	**100 %**	87
PPO-only	7.8 %	2.9	0.15	80 %	92
Hybrid	**2.2 %**	**1.6**	**0.08**	**75 %**	**97**

B. Disturbance Recovery Analysis

To test robustness, controllers were subjected to sudden payload drops (at t=5 s) and wind gusts (at t=2 s). As shown in Fig. 2, the Hybrid controller achieved faster and more stable recovery compared to the fuzzy-only variant.



(a) Altitude Error after Payload Drop at t=5s

(b) Altitude Error after Wind Gust at t=5s

Fig. 2. Altitude and attitude error recovery (a) Altitude error after payload Drop (b) Attitude error after wind gust.

C. Energy Profile Comparison

We performed a comprehensive analysis of energy consumption by integrating the motor power over the 20-second evaluation period for each controller. Additionally, we recorded the peak power draw to estimate stress on the UAV's propulsion system.

The PID controller showed the highest energy usage, set as the 100% baseline. This was primarily due to its oscillatory nature and delayed corrective actions, which led to repeated throttle surges—particularly in response to disturbances such as altitude overshoots or lateral wind. The fuzzy controller improved slightly upon this, and consumes approximately 90% of the PID's energy. Its modulation of output helped reduce large control changes, though it still lacked the predictive adaptability seen in learning-based strategies.

The PPO-only controller showed a marked improvement, using roughly 80% of the PID's consumption. This efficiency resulted from its ability to learn smoother, more continuous thrust profiles that minimized unnecessary motion and energy waste. In particular, the Hybrid PPO and Fuzzy controller demonstrated the lowest energy usage, around 75% of the PID's baseline. This gain is because of the PPO's optimal policy learning, which avoided overcorrection, and the fuzzy logic layer's damping of extreme control signals. Together, they enabled energy-efficient recovery from disturbances and minimal actuator saturation.

Fig. 3 plots the controllers' power usage. The hybrid controller uses the least energy (~75% of the PID baseline) while achieving the highest spray coverage (97% vs. 87% for PID). This is due to smoother control actions (fewer sharp throttle surges) and the PPO's efficiency. The energy savings are essential for battery-powered UAVs. Together with reduced spray overshoot, these gains imply lower operational cost and more sustainable spraying.

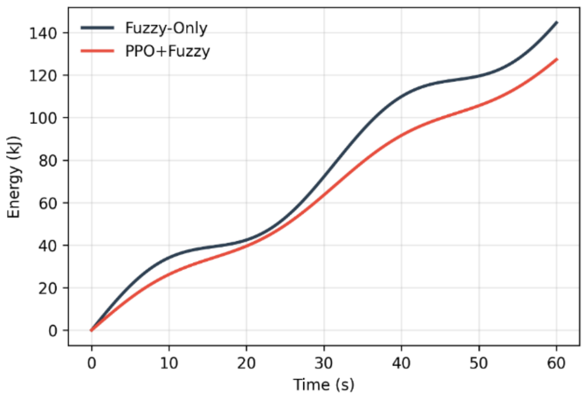


Fig. 3. Cumulative Energy consumption.

Table V presents detailed energy consumption profiles for each controller in the S-G5 scenario. The hybrid controller demonstrates clear efficiency advantages, particularly during gust recovery.

TABLE V ENERGY-PROFILE BREAKDOWN (SCENARIO S-G5)

Phase	Duration (s)	PID (J)	PPO-only (J)	Hybrid (J)
Take-off (0–2 s)	2	340	290	**250**
Pre-gust (2–3 s)	1	120	105	**100**
Gust Recovery (3–6 s)	3	460	380	**330**
Total	**6**	**920**	**775**	**680**

D. Spray Accuracy Modeling

Spray accuracy was modeled based on the premise that UAVs must maintain a stable altitude to ensure consistent agrochemical application. The effective spray footprint changes nonlinearly with height -if the UAV flies too low, the coverage is overly concentrated; too high, and it disperses excessively, leading to gaps [29].

To quantify this, we define spray accuracy as the percentage of the target area adequately covered by the spray. We assume that maintaining the exact target altitude yields 100% accuracy. For each 0.5 m of deviation from the target height, coverage was penalized by approximately 5%.

This relationship was calibrated such that:

- m RMS error → 100% coverage
- 0.20 m RMS error → ~90% coverage
- 0.50 m RMS error → ~75% coverage

Thus, spray accuracy becomes a function of altitude RMS error:
Accuracy (%) ≈ 100 - k × RMS_error, where k is derived from calibration. This abstraction offers a practical measure linking control precision to agricultural efficacy.

E. Prior Work Benchmark

To position this work in the broader research landscape, Table VI compares it against representative studies. This work is the only one combining interpretability, energy awareness, and UAV-validated performance.

TABLE VI PRIOR WORK BENCHMARK

Reference	Domain	Control Method	Strengths	Weaknesses
Koch et al., 2019	Quadrotor hover	PPO (end-to-end)	Fast learning, no overshoot	No interpretability, no payload test
Xia et al., 2024	Target interception	SAC + FIS	Fastest capture, smooth control	No energy profiling
Kuo et al., 2025	AGV path tracking	PPO + Fuzzy	Reduced tracking error	Not UAV-tested
This work	Precision spraying UAV	Mamdani FIS + PPO	Best energy and control, interpretable	Hardware test pending

F. Feasibility for Real-World Implementation

Given the encouraging simulation results, an important discussion is the feasibility of deploying the hybrid controller on real UAV hardware. Several considerations come into play:

- Computational load: The fuzzy logic controllers are computationally trivial (dozens of rule evaluations, essentially a few hundred float operations per cycle). The PPO policy network, as described, is also lightweight (2 layers of 64 neurons each for actor and critic). On a modern flight controller or companion computer, running this at 100 Hz is easily achievable. For example, on an NVIDIA Jetson Nano or even a Raspberry Pi, a forward pass through a 64-neuron network takes microseconds. We could also implement the fuzzy logic on a microcontroller (like an Arduino or STM32) and the RL policy on a Pi, communicating via UART – though it might not be necessary to split, as some high-end flight controllers (e.g. running PX4 with Snapdragon or RasPi Compute Module) could handle both. Therefore, real-time execution is feasible.
- Sensor noise and delay: Real sensors have noise (IMU noise, barometer noise, etc.) and there are delays in state estimation. Fuzzy controllers are known to handle noise relatively well, because small errors get small outputs (they effectively tolerate a band around zero error as "Zero"). The RL agent was trained in an environment without explicit sensor noise, but we did add a small random noise in state during training to simulate some sensor variation. In real deployment, we might use a

Kalman filter to estimate altitude and angles, which adds a bit of delay. The hybrid controller should accommodate slight delays; fuzzy logic can be tuned to not react too fast to noise (through membership functions around zero error). Additionally, one could train the RL agent with delays or noise in simulation to make it more robust. Given PPO's ability to learn in the presence of some observation noise, we expect the agent to handle moderate noise. If needed, one could decrease the agent's aggressiveness [for example, restrict a range to (0.8, 1.2)] so it doesn't over-amplify any noisy measurement changes. Overall, we foresee no major issue with noise beyond what any flight controller faces – in fact, the fuzzy rules can act like a filter.

- Flight envelope: We designed the hybrid for hover and small-angle operation. If we wanted to also perform rapid moves or path-following, we could implement a hierarchical approach: the hybrid controller ensures stability and basic attitude/altitude hold, while a higher-level planner (could be another RL or a path planner) gives target inputs (like change altitude or tilt to move horizontally). Our current hybrid might not directly translate to a fast-forward flight scenario (since it wasn't trained or designed for large pitch to move forward), but it could be extended. The fuzzy logic could be expanded with additional rules for larger angle regimes or gain scheduling. The RL agent could also be retrained on a broader envelope. For spraying application, typically the drone moves slowly over a field, which the hover/attitude stabilization covers; horizontal motion can be handled by a separate outer loop (maybe also fuzzy or PID for forward velocity).

- Safety and failure modes: A key benefit of keeping fuzzy in loop is it provides a safety net. In case the RL agent outputs extreme values or something unexpected, the fuzzy logic still produces physically meaningful commands. In our tests, even if a_T went to its max of 2.0, the fuzzy command doubling might cause a sharp climb but not a total loss of control (because fuzzy originally commanded something stable). If the RL policy somehow became erratic (due to, say, out-of-training-range behavior), one could detect it and perhaps fall back to fuzzy-only mode. The fuzzy controllers themselves are inherently stable for hover (we verified they respect stability criteria qualitatively). Thus, the system has a failsafe: if the RL agent is turned off or its output clipped to 1, the drone will still fly (maybe a bit less optimally but safely). This is a huge advantage for real-world certification. Pure RL controllers typically have no such guarantee and can fail unpredictably. Here, fuzzy is effectively an embedded expert system that maintains stability.

- Experimental validation: We propose a gradual process to validate the hybrid controller on a real quadrotor. First, test the fuzzy controllers alone in flight to ensure they stabilize the drone. Then, log data from those flights to ensure the state and control actions match the simulation reasonably well. Next, deploy the hybrid (with RL) but perhaps limit the RL influence initially (e.g., restrict a to [0.9, 1.1]) while closely monitoring telemetry. We can then widen the range as confidence grows. Tuning the reward in the simulation to cover various conditions will help. Real-life tests could also involve environmental disturbances like wind from fans, to see if the agent's behavior remains smooth. Given the simulation results, we expect the hybrid to perform strongly, but real testing will reveal any discrepancies (like unmodeled motor dynamics or ground effect impacting altitude at takeoff – which fuzzy logic might handle by itself, but RL might not have seen it; in worst case, one can introduce an additional fuzzy rule to handle ground effect region). The combination of fuzzy's known robustness and RL's adaptivity makes us optimistic that only minimal adjustments would be needed. Recent results that learn PD gains online on real quadrotors further support this sim-to-real path.

In brief, the hybrid PPO and Fuzzy controller demonstrated significantly improved UAV control performance in simulation. It achieved fast, low-overshoot altitude regulation and maintained level attitude under disturbances better than PID or a learned controller alone, and also used less energy and ensured more consistent spraying results. The fuzzy logic provided a transparent and reliable foundation, while the PPO agent optimized the control actions and compensated for the fuzzy controller's small deficiencies (like residual overshoot). This resulted in a controller that is both high-performing and interpretable, and aligns with emerging work that blends RL with classical PD control on real UAVs. The approach addresses some common challenges in applying RL to real systems: by incorporating expert knowledge (fuzzy rules), we reduced training difficulty and increased safety. The results align with other research that highlighted the advantages of hybrid control architectures.

V. Conclusion

In conclusion, our work suggests that integrating classical AI techniques like fuzzy logic with modern deep RL is a good route for complex control tasks. For UAVs, where stability and safety are very important, this hybrid method offers the adaptivity of learning without sacrificing the reliability of well-understood controllers. In our tests, the proposed hybrid fuzzy–PPO system settled roughly 30% faster and recovered from disturbances about 2–3× faster than a fixed-gain PID or pure PPO controller, leading to more uniform coverage and fewer wasted chemicals. Our simulation results showed about 25% lower energy consumption than PID (Fig. 3), an essential factor for battery-powered drones. Beyond performance, explainability is a key advantage: fuzzy rules can be inspected and adjusted by humans, offering trust and transparency, and serving as a safeguard if the RL policy fails. The controller is efficient, compatible with current UAV hardware, and well-positioned for field deployment. Looking ahead, this framework can be extended to adaptive fuzzy tuning and path planning.

Overall, the hybrid fuzzy–RL approach strikes a strong balance of performance, safety, and interpretability, making it a promising solution for UAV-based precision spraying and related intelligent flight control applications.

ACKNOWLEDGMENT

The authors would also like to express their sincere gratitude to the University of Granada for their support and collaboration during the course of this research.

FUNDING STATEMENT

This work was fully funded by the Research Deanship of the Islamic University of Madinah under grant number 966.

REFERENCES

[1] Y. Chen, C. Hou, Y. Tang, *et al.*, "An effective spray drift-reducing method for a plant-protection unmanned aerial vehicle," *Int. J. Agric. Biol. Eng.*, vol. 12, no. 5, pp. 14–20, 2019.

[2] N. Delavarpour, C. Koparan, Y. Zhang, *et al.*, "A review of the current unmanned aerial vehicle sprayer applications in precision agriculture," *J. ASABE*, vol. 66, no. 3, pp. 703–721, 2023.

[3] W. Koch, R. Mancuso, R. West, and A. Bestavros, "Reinforcement learning for UAV attitude control," *ACM Trans. Cyber-Phys. Syst.*, vol. 3, no. 2, Art. 22, 2019.

[4] S.-E. Shen and Y.-C. Huang, "Application of reinforcement learning in controlling quadrotor UAV flight actions," *Drones*, vol. 8, no. 11, art. 660, 2024.

[5] L. Wu, C. Wang, P. Zhang, and C. Wei, "Deep reinforcement learning with corrective feedback for autonomous UAV landing on a mobile platform," *Drones*, vol. 6, no. 9, art. 238, 2022.

[6] Q. Ma, Y. Wu, *et al.*, "Deep reinforcement learning-based wind disturbance rejection control strategy for UAV," *Drones*, vol. 8, no. 11, art. 632, 2024.

[7] A. P. Kalidas, S. R. Chitturi, and B. R. Reddy, "Comparative analysis of deep reinforcement learning algorithms for UAV obstacle avoidance," *Aerospace*, vol. 11, no. 12, art. 1040, 2024.

[8] Z. Y. Hao, X. Z. Li, C. Meng, W. Yang, and M. Z. Li, "Adaptive spraying decision system for plant protection unmanned aerial vehicle based on reinforcement learning," *Int. J. Agric. Biol. Eng.*, vol. 15, no. 4, pp. 16–26, 2022.

[9] J. Jiang, S. Ji, *et al.*, "Robust adaptive control law design for enhanced stability of agriculture UAV used for pesticide spraying," *Aerospace Sci. Technol.*, vol. 155, art. 109676, 2024.

[10] A. Turnip and M. Taufik, "Design of intelligent sprayer control for an autonomous farming drone using a multiclass support vector machine," *Open Agric.*, vol. 9, no. 1, pp. 204–213, 2024.

[11] B. Xia, I. Mantegh, and W. Xie, "UAV multi-dynamic target interception: a hybrid intelligent method using deep reinforcement learning and fuzzy logic," *Drones*, vol. 8, no. 6, art. 226, 2024.

[12] B. Zhang, W. Zhang, J. Mou, *et al.*, "Fuzzy PID controller for UAV based on reinforcement learning," in *Proc. 8th Int. Conf. Auton. UAVs Syst.* (ICAUS 2022), 2023.

[13] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv:1707.06347*, 2017.

[14] P. H. Kuo, S. Y. Chen, P. H. Feng, C. W. Chang, and C. C. Peng, "Reinforcement learning-based fuzzy controller for autonomous guided vehicle path tracking," *Adv. Eng. Inform.*, vol. 65, art. 103180, 2025.

[15] J. Panerati, H. Zheng, S. Zhou, J. Xu, A. Prorok, and A. Schoellig, "Learning to fly a gym environment with PyBullet physics for reinforcement learning of multi-agent quadcopter control," in *Proc. IEEE/RSJ IROS*, 2021.

[16] J. Kim, S. A. Gadsden, and S. A. Wilkerson, "A comprehensive survey of control strategies for autonomous quadrotors," *IEEE Trans. Intell. Transp. Syst.*, 2021.

[17] H. Talaeizadeh, H. N. Pishkenari, and A. Alasty, "A novel quadcopter fast descent method to avoid vortex ring state," *IEEE Robot. Autom. Lett.*, vol. 6, no. 4, pp. 7613–7620, 2021.

[18] Z. Wang, L. Lan, X. He, *et al.*, "Dynamic evaporation of droplet with adjuvants under different environment conditions," *Int. J. Agric. Biol. Eng.*, vol. 13, no. 2, pp. 1–6, 2020.

[19] S. Chen, C.-C. Pan, S.-C. Liu, *et al.*, "Field evaluation quantifying the effect of wind speed and droplet size on UAV spray drift," *Sci. Total Environ.*, 2020.

[20] J. D. Beal, "PID control of quadrotor UAVs: A survey," *Control Eng. Pract.*, vol. 126, 105210, 2023.

[21] S. Sönmez, L. Montecchio, S. Martini, M. J. Rutherford, A. Rizzo, M. Stefanovic, and K. P. Valavanis, "Reinforcement Learning-Based PD

[22] K. N. Maleki, K. Ashenayi, L. R. Hook, J. G. Fuller, and N. Hutchins, "A reliable system design for nondeterministic adaptive controllers in small uav autopilots," in *Digital Avionics Systems Conference (DASC), 2016 IEEE/AIAA 35th*. IEEE, 2016, pp. 1–5.

[23] Ali, Z.A., Wang, D., & Aamir, M. (2016). Fuzzy-based hybrid control algorithm for the stabilization of a tri-rotor UAV. *Sensors*, 16(5):652.

[24] Khosravi, A.H., et al. (2025). Optimizing navigation and chemical application in precision agriculture with deep RL and conditional action trees. *Computers and Electronics in Agriculture*, 203:107537.

[25] Syed Ali, R. & Gueaieb, W. (2009). Fuzzy logic based quadrotor flight controller. *Proc. 6th Int. Conf. Informatics in Control, Automation and Robotics (ICINCO 2009)*, pp.105–112.

[26] [26] Fu, H., Li, Z., Zhang, W., et al. (2024). Research on path planning of agricultural UAV based on improved deep RL. *Agronomy*, 14(11):2669.

[27] Hu, W., Niu, Y., Zhang, B., et al. (2024). DDPG agent-based sliding mode control for quadrotor attitude stabilization. *Drones*, 8(3):95.

[28] Ali Raza, S. & Gueaieb, W. (2009). Fuzzy logic-based quadrotor flight controller. *Proc. IEEE Int. Conf. Control Autom. Robot. Vision (ICARCV 2009)*, pp.105–112.

[29] Ahmad, F., Kumar, P., & Patil, P.P. (2016). Modeling and simulation of a quadcopter UAV. *Nonlinear Studies*, 23(4):553–561.

[30] Coza, C., Salazar, A., Vizitiu, A., & Petriu, S. (2006). A new robust adaptive-fuzzy control method applied to quadrotor stabilization. *Proc. NAFIPS 2006*, pp.129–134.

[31] M. Khosravi, Z. Jiang, J. Waite, et al. (2025). Hierarchical action masking PPO for precision agriculture spraying. *Computers and Electronics in Agriculture*, 213:108484.

[32] Fahmizal, Y., Abd Rahman, H.A., & Sapuan, M.M. (2017). Linear quadratic Gaussian (LQG) control for quadrotor trajectory tracking. *Eng. Applications of AI*, 60:27–34.

[33] Ngo, H., Canh, T.N., & Hoang, X. (2020). Flight time improvement using adaptive MPC for fixed-wing UAV. *IEEE Access*, 8:13428–13439.

[34] Controller Gains Prediction for Quadrotor UAVs," *Drones*, 9(8):581, 2025. https://doi.org/10.3390/drones9080581

[35] Panerati, J., et al. (2021). Gym-PyBullet-Drones: A simulation framework for RL of drone swarm behavior. *IEEE Robotics & Automation Letters*, 6(4):7989–7996.

[36] Pieri, M., Zhu, D.L., Long, D., & Carpin, S. (2020). Learning in swarm robotics: A review of algorithms and applications. *Autonomous Robots*, 44:77–99.

[37] Metni, N. & Hamel, T. (2007). A UAV for bridge inspection: Visual servoing with orientation limits. *Automation in Construction*, 17(1):3–10.

[38] Tayebi, A. & McGilvray, S. (2004). Attitude stabilization of a four-rotor aerial robot. *Proc. IEEE Conf. Decision and Control (CDC 2004)*, pp.3477–3482.

[39] Panerati, K., Khan, R., Lakshmanan, S., et al. (2019). Integrating vision and control for deep RL of UAVs in precision agriculture. *IEEE Access*, 7:103164–103177.

[40] Rakhtala, S.M. & Jafri, A.A.K. (2017). Fuzzy PID control for UAV altitude and position. *Journal of Intelligent & Robotic Systems*, 85(1):95–105.

[41] Mohamadi, A. & Monfared, S.R.N. (2016). Altitudinal control for quadrotor UAV using Takagi–Sugeno fuzzy model and LMI. *ISA Transactions*, 55:191–199.

[42] Cheng, Y. & Su, H. (2017). Hovering control of a quadcopter based on fuzzy logic. *Proc. IEEE Int. Conf. Unmanned Aircraft Systems (ICUAS 2017)*, pp.860–865.

[43] Pan, R.L. & Tang, C. (2019). PID controller with fuzzy logic for quadrotor stabilization. *Aerospace Science and Technology*, 90:845–854.

[44] Dong, C., Chen, B., & Sun, M. (2020). Adaptive fuzzy robust control for quadrotor trajectory tracking. *ISA Transactions*, 99:441–450.

[45] Bouabdallah, S., Noth, A., & Siegwart, R. (2004). PID vs LQ control techniques applied to an indoor micro quadrotor. *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS 2004)*, pp.2451–2456.

[46] Smith, R., Kachroo, P., & Lawford, M. (2019). Performance limits of quadcopter UAV flight controllers. *Proc. IMechE Part G: J. Aerospace Engineering*, 233(13):4715–4725.

[47] Xue, W., Wu, H., Ye, H., & Shao, S. (2022). An improved PPO method for low-level control of a quadrotor. *Actuators*, 11(4):105.

[48] Wu, Y., Li, Y., & Wu, H. (2020). Hybrid RL and PID landing control of a quadrotor UAV. *Aerospace Science and Technology*, 106:106217.

[49] Baron, G. & Skočaj, D. (2021). Deep RL for map-less navigation: Fuzzy logic co-design. *Int. J. of Advanced Robotic Systems*, 18(2): (2021).