Adaptive UAV Inspection of PV Panels Using Goal-Conditioned Reinforcement Learning and Zigzag Coverage Planning

Imen Habibi IMEN.HABIBI@ENSTAB.UCAR.TN
Ikbal Chammakhi Msadaa IKBAL.MSADAA@ENSTA.U-CARTHAGE.TN
Khaled Grayaa KHALED.GRAYAA@ENSTA.U-CARTHAGE.TN
LaRINa, ENSTAB, University of Carthage, Tunisia

Abstract

Accurate and efficient inspection of photovoltaic (PV) panels is critical for early anomaly detection and energy yield optimization. This study presents an autonomous Unmanned Aerial Vehicles UAV-based inspection framework that leverages Goal-Conditioned Reinforcement Learning (GCRL) for adaptive path tracking. The UAV follows a mathematically defined zigzag trajectory while dynamically responding to disturbances such as wind drift. Instead of rigid waypoint following, the agent is conditioned on successive inspection goals and learns optimal movement strategies using Proximal Policy Optimization (PPO). The environment incorporates realistic wind noise and UAV momentum, requiring the policy to learn corrective behaviors under uncertainty. Simulation results demonstrate the agent's ability to achieve robust full-surface coverage, minimize overlap, and maintain trajectory alignment, highlighting the effectiveness of this learning-based inspection strategy.

Keywords: PV inspection, UAV path tracking, Goal-Conditioned Reinforcement Learning, PPO, Zigzag trajectory, Wind robustness, Autonomous drone coverage.

1. Introduction

The global shift toward renewable energy has led to a significant expansion of photovoltaic (PV) solar installations worldwide, intensifying the need for efficient inspection and maintenance strategies. Timely identification of surface-level anomalies,such as cracks, soiling, or hotspots,is crucial to preserving system efficiency and maximizing energy yield (Habibi et al., 2024).

Traditional manual inspection techniques are labor-intensive, time-consuming, and often infeasible for large-scale deployments. In response, Unmanned Aerial Vehicles (UAVs) equipped with high-resolution RGB and thermal cameras have emerged as a scalable, cost-effective alternative (Habibi et al., 2024). However, the effectiveness of UAV-based inspection systems hinges on the quality of the underlying path planning algorithm. Inefficient trajectories can result in missed defects, excessive image overlap, prolonged inspection time, and increased energy consumption (Pérez-González et al., 2021).

Among various coverage strategies, the zigzag pattern remains widely used due to its simplicity and ability to uniformly cover planar surfaces such as solar panel arrays (Hammer Missions, n.d.). To systematize this approach, we formulate a mathematical model for generating discrete waypoints based on camera field of view (FOV), overlap constraints, and flight parameters. This enables predictive estimation of inspection completeness, time, and resource requirements.

Yet, rigid waypoint following is often inadequate in real-world deployments, where UAVs face disturbances such as wind gusts, inertia, and actuation delays. To enhance robustness and adaptability under such conditions, we propose a Goal-Conditioned Reinforcement Learning (GCRL) framework for adaptive trajectory tracking. Specifically, we condition the policy on successive goal waypoints derived from the zigzag path, enabling the UAV to learn optimal control strategies that minimize deviation while preserving coverage efficiency. The agent is trained using Proximal Policy Optimization (PPO) in a realistic simulation environment incorporating wind noise and momentum dynamics. This approach bridges the gap between structured path planning and reactive control, combining the advantages of geometry-based inspection with the flexibility of learning-based navigation.

Simulation results demonstrate the ability of the proposed GCRL-PPO framework to maintain robust trajectory alignment, ensure full PV surface coverage, and minimize redundant image overlap—even under dynamic conditions.

The remainder of this paper is organized as follows. Section 2 reviews related work on UAV-based inspection, path planning, and reinforcement learning in uncertain environments. Section 3 introduces the theoretical basis for modeling the inspection path. Section 4 details the implementation of the zigzag coverage strategy. Section 5 presents the proposed GCRL framework and PPO-based training methodology. Section 6 presents the experimental setup and simulation results, including environment design, agent architecture, and performance analysis. Finally, Section 7 concludes the paper and outlines directions for future work, including real-world deployment and anomaly-aware decision integration.

2. Related Work

Previous studies on UAV-based PV panel inspection emphasize the importance of Coverage Path Planning (CPP) to ensure efficient and complete surface analysis. Zigzag (boustrophedon) trajectories have proven effective for flat rectangular layouts, but conventional approaches such as the work by Pérez-González *et al.* (Pérez-González et al., 2021) rely on static offline planning and do not handle environmental disturbances like wind.

On the other hand, spiral and helical trajectories are better suited to curved structures. However, their application to flat PV arrays often results in overlap inefficiencies, as reported by Silberberg *et al.* Silberberg (2018), and assumes perfect tracking when extended to multi-UAV systems, as done by Luna *et al.* Luna et al. (2023).

To enhance robustness, reinforcement learning (RL) approaches have been proposed. Energy-Saving Path Planning-RL (ESPP-RL), introduced by Chen *et al.* (Chen et al., 2024), adapts drone navigation to 3D turbulent environments but lacks the capability for structured surface coverage. GCRL, explored by Lee *et al.* and Kim *et al.* (Lee et al., 2023; Kim et al., 2025), demonstrates improved generalization in navigation tasks but does not address full-coverage inspection under real-world disturbances.

Our contribution bridges the gap between traditional coverage path planning and adaptive UAV control by integrating geometric CPP with GCRL using PPO. Unlike prior approaches that follow fixed trajectories, we treat inspection waypoints as dynamic goals and train the UAV to autonomously transition between them. The agent learns to maintain full coverage while adapting to real-world disturbances such as wind and UAV momentum. This

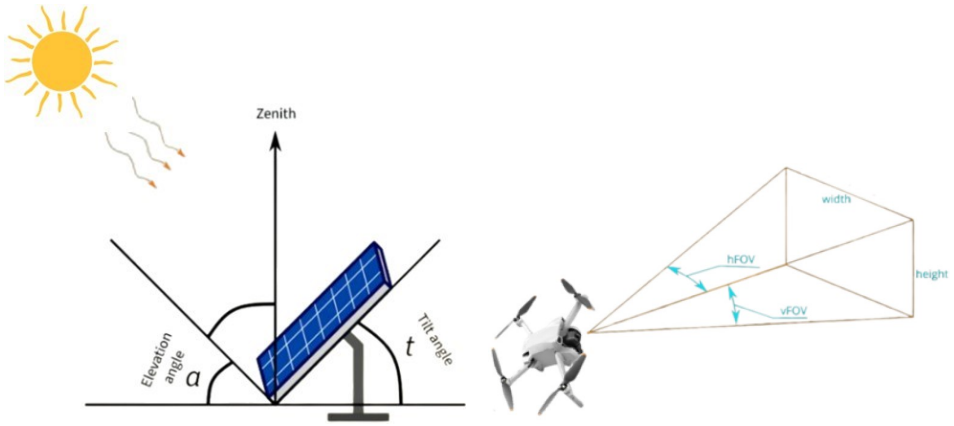
approach enables robust, scalable, and efficient inspection of large-scale PV arrays without relying on manually tuned control rules.

3. Theoretical Modeling of Inspection Paths

Efficient drone-based inspection of PV panels critically depends on the choice of the trajectory pattern, which directly influences coverage quality, inspection time, and energy consumption. This study centers on the mathematical formulation of a zigzag path, analyzing its effectiveness in terms of total coverage area, number of waypoints, inspection duration, and overall travel distance. The design and evaluation of the path planning strategy are informed by key constraints, including the drone's FOV, desired image overlap, and the geometric configuration of the PV panel layout.

Figure 1 provides visual references for two essential elements in inspection planning: the panel tilt angle (Figure 1(a)) and the drone's horizontal and vertical FOV (Figure 1(b)), both of which play a pivotal role in determining the spacing and orientation of the inspection trajectory.

This analysis is grounded in realistic operational assumptions regarding PV system dimensions, drone altitude, camera specifications, and desired overlap margins. These parameters are consolidated in Table 1, forming the basis for a robust and application-aware path planning framework.



(a) Panel tilt angle (b) Horizontal and vertical FOV

Figure 1: Illustration of (a) the tilt angle for solar panels; and (b) the horizontal and vertical FOV in drone imaging.

In Tunisia, for instance, the optimal tilt angle for PV panels is $\theta = 32°$ Nagar and Rai (2024). To maintain proper alignment with the panel's tilt, the vertical altitude offset Δaltitude of the UAV is computed using Equation (1) (Hammer Missions, n.d.).

$$\Delta\text{altitude} = \frac{W_{\text{panel}}}{2} \cdot \sin(\theta) \tag{1}$$

Table 1: Key Assumptions for PV Geometry and UAV Imaging

PV Panel Specifications	
Panel Height	163–194 cm (64”–76.5”)
Panel Width	99–131 cm (39”–51.5”)
Panel Depth	3–5 cm (1.2”–2”)
Cells per Panel	60, 72, or 96
Cell Dimensions	15 cm × 15 cm (6 × 6”)
Panel Efficiency	14%–17% (Polycrystalline)
Tilt Angle (θ)	32° (Tunisia)
Drone and Imaging Parameters	
FOV	Angular camera coverage (assumed square)
Overlap (o)	10% horizontal and vertical
Camera Focal Distance	Controls resolution and zoom
Drone Speed	$v = 500$ cm/s
Capture Time (t)	5 ms per image

where W_{panel} is the panel width and θ is the tilt angle of the PV surface.

For optimal imaging coverage and UAV positioning, the horizontal coverage width W_{h} at a distance d from the panel surface is given by Equation (2) (Carrot, 2024), where FOV denotes the field of view angle of the camera:

$$W_h = 2d \cdot \tan\left(\frac{\text{FOV}}{2}\right) \tag{2}$$

Assuming a square sensor, the vertical coverage is similarly $W_v = W_h$.

This study adopts a zigzag trajectory for inspection, a systematic coverage strategy commonly used in UAV-based surveys. It consists of sequential horizontal sweeps across the target area, followed by vertical shifts between rows, forming a back-and-forth scanning pattern. To ensure complete surface coverage with minimal overlap, the spacing between image capture points is determined by the effective field of view, desired overlap ratio, and the geometric layout of the panel array. This approach strikes a balance between flight efficiency and inspection quality, making it particularly well-suited for flat PV installations.

The geometric and temporal structure of the zigzag inspection path is governed by a set of core parameters, all of which are derived from the UAV's field of view, operating altitude, image overlap requirement, and the spatial layout of the inspection site (e.g., photovoltaic panel rows or vertical surfaces). Table 2 summarizes the main equations used to calculate the effective field of view, number of required waypoints, horizontal and vertical travel distances, and the total inspection time.

The effective width of coverage W_{eff} represents the portion of the scene that the UAV can inspect in a single image while maintaining a desired percentage of overlap. It is computed as a function of the camera's FOV, the UAV-to-surface distance d, and the overlap factor

o, using the formula $W_{\text{eff}} = 2d \cdot \tan\left(\frac{\text{FOV}}{2}\right) \cdot (1 - o)$. This equation ensures that adjacent inspection sweeps cover the surface without redundant imaging or coverage gaps.

Table 2: Compact Zigzag Path Planning Parameters and Equations

Symbol	**Description**	**Equation**
W_{eff}	Effective coverage width or height per image	$2d\tan\left(\frac{\text{FOV}}{2}\right)(1-o)$
N_h, N_v	Number of waypoints per row and column	$\left\lceil \frac{W_{\text{total}}}{W_{\text{eff}}} \right\rceil$, $\left\lceil \frac{H_{\text{total}}}{W_{\text{eff}}} \right\rceil$
$D_{\text{hor}}, D_{\text{ver}}$	Total horizontal and vertical travel distances	$N_v(N_h-1)W_{\text{eff}}$, $(N_v-1)\sqrt{W_{\text{effh}}^2 + W_{\text{effv}}^2}$
T_{total}	Total inspection duration (flight + capture time)	$\frac{D_{\text{hor}} + D_{\text{ver}}}{v} + N_h N_v t_{\text{capture}}$

The total number of waypoints along the horizontal and vertical axes of the inspection area are denoted by N_h and N_v, respectively. These values are computed by dividing the total width W_{total} and height H_{total} of the target area by the effective coverage dimension, then rounding up to ensure full coverage: $N_h = \left\lceil \frac{W_{\text{total}}}{W_{\text{eff}}} \right\rceil$, $N_v = \left\lceil \frac{H_{\text{total}}}{W_{\text{eff}}} \right\rceil$. This discretization directly determines the number of inspection images and traversal sweeps required.

The total horizontal travel distance D_{hor} accounts for the movement along each row of the zigzag path, assuming the UAV passes through N_h points per row for each of the N_v layers. Since the UAV alternates directions between rows, the cumulative horizontal distance is given by $D_{\text{hor}} = N_v(N_h - 1)W_{\text{eff}}$. Similarly, the vertical transition distance D_{ver} represents the drone's movement between rows and is approximated by a summation of diagonal distances between waypoints: $D_{\text{ver}} = (N_v - 1) \cdot \sqrt{W_{\text{effh}}^2 + W_{\text{effv}}^2}$. This accounts for the slanted movement between rows when wind drift or imperfect alignment is present.

Finally, the total inspection time T_{total} includes both the UAV's flight duration and the cumulative time required for capturing images at each waypoint. The flight time component is given by $\frac{D_{\text{hor}}+D_{\text{ver}}}{v}$, which includes both horizontal and vertical traversal time based on UAV velocity v. The second component $N_h \cdot N_v \cdot t_{\text{capture}}$ accounts for image acquisition time at each of the inspection points, assuming a fixed per-capture delay t_{capture}. The total inspection time is given by Equation (3).

$$T_{\text{total}} = \underbrace{\frac{D_{\text{hor}} + D_{\text{ver}}}{v}}_{\text{Flight time}} + \underbrace{N_h \cdot N_v \cdot t_{\text{capture}}}_{\text{Image capture time}} \tag{3}$$

This decomposition provides a comprehensive estimate of the total inspection duration and supports path planning decisions that consider energy constraints, data storage limits, and mission completion time.

Before proceeding with the implementation of the zigzag path planning algorithm, we conducted in Habibi (2024) a comparative study of various inspection strategies, including spiral and other common patterns, applied to different structure shapes. The results demonstrated that the zigzag path consistently provided the most efficient coverage in terms of completeness, time, and overlap minimization. Therefore, we selected it as the preferred path planning approach for the inspection of PV solar panels.

4. Zigzag Path Planning and Implementation

The implementation assumes a structured layout of 12 solar panels per row, where each panel measures 115 cm in width and 178.5 cm in height. Panels are composed of uniformly spaced solar cells of 15 cm × 15 cm. The UAV is set to fly at a constant speed of 500 cm/s and captures an image at each waypoint with a delay of 5 ms. To avoid inspection gaps and ensure seamless stitching, a 10% overlap is maintained in both horizontal and vertical directions. These parameters define the coverage footprint per capture and allow for accurate calculation of the total number of waypoints required for full inspection.

To automate this process, the zigzag waypoint generation algorithm accounts for panel layout geometry, FOV constraints, and overlap settings. The UAV sweeps horizontally across each row and alternates direction as it moves vertically through the grid. The complete path generation is summarized in Algorithm 1.

Algorithm 1: Zigzag Path Planning for PV Inspection

Input: W_{panel}, H_{panel}, N_{panels}, C_w, C_h, o
Output: Waypoint list
Compute effective FOV area and total layout size;
Determine N_h and N_v from layout and overlap;
for *each row* i *in* N_v **do**
    Generate N_h waypoints across row;
    if i *is odd* **then**
        Reverse row waypoint order;
    end
    Append row waypoints to path;
end
return *waypoint list*;

The simulation results of the computed zigzag path are shown in Figure 2. The drone's trajectory fully spans the inspection area, following calculated waypoints with uniform spacing. The figure represents the PV panel grid and cell structure alongside the inspection path. Total travel distance and inspection time are derived from the modeling equations. The visualization was created using Matplotlib for clarity and accurate 2D representation.

The result confirms that the proposed path planning strategy achieves full surface coverage without gaps or excessive overlap. This validates the suitability of the zigzag pattern for UAV-based inspection of structured PV arrays.

Following this modeling, we integrate the zigzag trajectory into a GCRL framework. This enables the UAV to autonomously follow the planned path and adapt its behavior in

real time to dynamic environmental conditions, such as wind, ensuring efficient and resilient inspection performance.

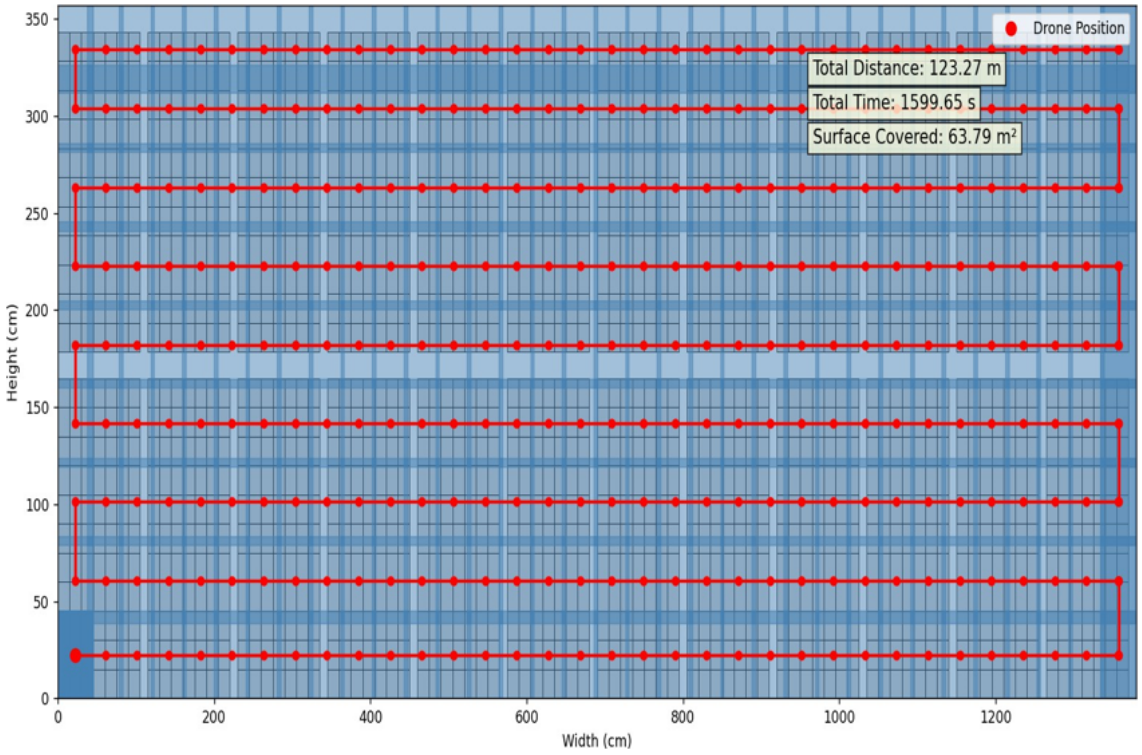


Figure 2: Visualization of Zigzag Path for UAV Inspection.

5. GCRL for UAV Zigzag Path Tracking

To achieve reliable and adaptive tracking of the predefined zigzag inspection trajectory, we design a GCRL framework. Traditional waypoint-following approaches often struggle in dynamic environments, particularly when external disturbances such as wind introduce deviations from the planned path. In contrast, GCRL allows the UAV to learn a flexible control policy that is explicitly conditioned on successive inspection goals, enabling it to continuously reorient its behavior toward the next target location while maintaining awareness of the overall path structure.

By representing waypoints along the zigzag pattern as dynamic goals, the UAV is not merely executing a rigid sequence of commands but is instead learning a goal-directed navigation strategy. This formulation enhances robustness by allowing the policy to generalize to off-nominal situations,such as drift, overshoot, or delayed actuation, and correct its trajectory on-the-fly without relying on hand-crafted control rules.

Moreover, the GCRL framework naturally supports continuous learning and adaptation, enabling the UAV to balance two competing objectives: adhering closely to the inspection

trajectory and responding effectively to real-time perturbations. This results in more reliable surface coverage, reduced overlap and redundancy, and improved inspection quality in uncertain outdoor conditions. In this context, GCRL proves particularly effective for inspection missions where environmental variability and precise maneuvering are critical.

5.1. Problem Formulation

The inspection task is formulated as a Goal-Conditioned Markov Decision Process (GC-MDP):

$$\mathcal{M}_g = (\mathcal{S}, \mathcal{A}, \mathcal{G}, \mathcal{R}) \tag{4}$$

where:

- $\mathcal{S}$: The state space, defined as the drone's normalized 2D position.
- $\mathcal{A}$: Continuous 2D action space $\in [-1, 1]^2$, representing motion vectors $[\Delta x, \Delta y]$.
- $\mathcal{G}$: The goal space, a dynamic sequence of waypoints derived from the zigzag path.
- $\mathcal{R}$: The reward function, designed to guide coverage and penalize revisits or drift.

At each time step t, the agent receives a state:

$$s_t = (\text{observation}_t, \text{achieved\_goal}_t, \text{desired\_goal}_t) \tag{5}$$

where:

- observation_t: Current UAV position, normalized to the grid.
- achieved_goal_t: Current position (used to track coverage).
- desired_goal_t: Next waypoint on the zigzag trajectory.

5.2. Modeling Wind Disturbance and UAV Momentum

To realistically simulate environmental uncertainty during UAV inspection, we introduce a wind disturbance model based on Gaussian noise, which perturbs the UAV's intended action output. This mechanism mimics atmospheric turbulence and wind gusts that affect aerial vehicles during flight. At each time step t, the policy network outputs a continuous action vector $a_t = (dx_t, dy_t)$ representing the UAV's intended direction of movement. Before this action is applied, it is disturbed by anisotropic Gaussian noise to reflect wind dynamics, as shown in Equation (6):

$$a_t^{\text{disturbed}} = \text{clip}(a_t + \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})) \tag{6}$$

where:

- a_t: original action vector from the policy network.
- $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$: zero-mean multivariate Gaussian noise with covariance matrix $\mathbf{\Sigma}$.
- $\mathbf{\Sigma} = \text{diag}(\sigma_x^2, \sigma_y^2)$: defines anisotropic variance, with $\sigma_x > \sigma_y$ to simulate stronger wind influence in the x-direction.

- clip$(\cdot)$: bounds the final disturbed action within $[-1, 1]$ per dimension, ensuring it lies within the action space.

This noise-driven disturbance introduces stochasticity into the agent's dynamics and challenges the policy to adapt its control strategy under external forces. Importantly, the anisotropic nature of $\mathbf{\Sigma}$ reflects realistic wind profiles where lateral gusts are more prominent than vertical turbulence at low altitudes.

In addition to the wind effect, UAV dynamics are influenced by inertia, which causes gradual transitions in direction due to the vehicle's physical momentum. This is modeled via an exponential smoothing mechanism where the UAV's current velocity v_t is updated based on the previous velocity v_{t-1} and the disturbed action $a_t^{\text{disturbed}}$, as given in Equation (7):

$$v_t = \gamma v_{t-1} + (1 - \gamma)\, a_t^{\text{disturbed}} \tag{7}$$

where:

- v_t: velocity vector at time t.
- v_{t-1}: previous velocity, capturing the inertia of the UAV.
- $\gamma \in [0, 1]$: momentum coefficient controlling the influence of past velocity (typically set close to 1).
- $a_t^{\text{disturbed}}$: wind-affected control action.

This momentum model provides a low-pass filter effect on the UAV's motion, preventing sudden jerks or unrealistic movement transitions. When combined with wind disturbance, the UAV exhibits smoother and more lifelike motion trajectories.

By integrating both wind disturbance and momentum, the simulation environment presents the agent with the dual challenge of (1) resisting stochastic perturbations and (2) planning corrective motions to reach goals under uncertainty. This formulation encourages the policy to learn robust control strategies that dynamically compensate for drift, thereby enhancing the UAV's ability to maintain precise coverage of the inspection area. Moreover, this setup enables studying the impact of environmental noise on inspection efficiency, energy consumption, and trajectory stability in safety-critical scenarios.

5.3. Reward Design

In real-world autonomous inspection tasks, the success of a UAV does not lie solely in reaching predefined goals,it lies in how it reaches them. A drone navigating turbulent airflows, subject to inertia, limited energy, and imaging constraints, must be guided not just by destination, but by discipline. To instill this behavior, our reward function is not designed as a checklist of incentives, but as a continuous signal that teaches the agent how to move with purpose, how to recover from drift, and how to maximize inspection integrity.

The reward formulation is therefore crafted to induce smooth, corrective motion while enforcing full, non-redundant coverage. A sparse bonus is granted when the UAV reaches a goal waypoint, but this is only part of the signal. To avoid the brittleness of sparse-only

rewards, we embed shaping terms that reinforce coverage efficiency and trajectory discipline. The final reward function is defined as:

$$r_t = +1.0 \cdot \mathbb{1}_{\text{goal}} - 0.1 \cdot \mathbb{1}_{\text{overlap}} - 0.01 \cdot \|s_t - g_t\|_2 - 0.001 \tag{8}$$

Here, $\mathbb{1}_{\text{goal}}$ is an indicator function that returns 1 if the UAV has reached its current goal g_t, and 0 otherwise. The overlap penalty term $\mathbb{1}_{\text{overlap}}$ penalizes the UAV for revisiting previously covered grid cells, encouraging exploration and enforcing full inspection coverage. The term $\|s_t - g_t\|_2$ represents the Euclidean distance between the UAV's current position s_t and the active goal, providing a continuous signal that promotes convergence toward the target even when stochastic wind introduces deviation. Lastly, the constant term -0.001 models energy consumption and introduces a cost for each step, pushing the agent to complete its inspection using the most efficient trajectory possible.

This reward design is not intended to reward mere success, but to shape behavior, discouraging hesitation, overlap, and wandering, while nurturing robustness and adaptability. In environments where wind and momentum distort ideal paths, the reward acts as a stabilizing guide. By rewarding not only the arrival at goals, but the manner in which they are reached, we enable the policy to learn control strategies that are smooth, efficient, and recoverable. Such behavioral shaping is essential in inspection settings where safety, coverage completeness, and trajectory robustness are interdependent.

5.4. PPO-Based Learning Framework

To optimize the inspection policy, we adopt the PPO algorithm, which is well-suited for continuous control tasks under uncertainty. PPO is chosen for its balance between stability and sample efficiency, making it ideal for training policies in environments with high variance due to wind disturbances and momentum dynamics.

The policy is modeled using a goal-conditioned actor-critic architecture, where both the actor and critic are conditioned on the current observation and the desired goal. At each timestep t, the actor network $\pi(a_t \mid s_t, g_t)$ outputs a distribution over continuous 2D actions $a_t \in [-1, 1]^2$, representing movement directions. The critic network $V(s_t, g_t)$ estimates the expected cumulative reward (value) from the current state-goal pair.

PPO optimizes the policy using a clipped surrogate objective that constrains how much the new policy is allowed to deviate from the previous one, reducing the risk of policy collapse. The clipped PPO objective is given by:

$$L_{\text{PPO}} = \mathbb{E}_t \left[\min \left(r_t(\theta)\hat{A}_t,\ \text{clip}\left(r_t(\theta),\ 1-\epsilon,\ 1+\epsilon\right)\hat{A}_t \right) \right] \tag{9}$$

Here, $r_t(\theta) = \frac{\pi_\theta(a_t|s_t,g_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t,g_t)}$ denotes the probability ratio between the new and old policy, measuring the shift in action likelihood. The term $\hat{A}_t$ represents the estimated advantage function, which quantifies how much better or worse the chosen action a_t was compared to the expected value baseline. This advantage is computed using Generalized Advantage Estimation (GAE), which blends temporal-difference (TD) errors across multiple time horizons for smoother and lower-variance updates.

The `clip` operation restricts the policy update to stay within a conservative range around the old policy by clipping $r_t(\theta)$ to the interval $[1-\epsilon, 1+\epsilon]$, where ϵ is a small hyperparameter (commonly $\epsilon = 0.2$). This constraint prevents excessively large policy updates that might degrade performance, especially in unstable environments.

By optimizing this clipped surrogate loss, the agent incrementally improves its policy while maintaining stability and preventing destructive updates. This makes PPO particularly effective for training in the UAV inspection scenario, where precise action control, recovery from drift, and generalization under wind noise are critical for success.

6. Experiments and results

This section presents the detailed implementation setup, training configuration, and empirical results of our proposed UAV inspection framework. We evaluate the learning dynamics, policy robustness, and coverage performance under realistic disturbances to validate the effectiveness of the PPO-based controller in achieving full inspection with minimal redundancy.

6.1. Simulation Setup

To simulate autonomous UAV inspection over a structured environment, we developed a custom control environment using the Gymnasium framework, tailored to reflect both realistic inspection constraints and aerodynamic dynamics. The inspection zone is modeled as a discrete 10×10 two-dimensional grid, with each grid cell representing a potential inspection point. The drone's objective is to follow a complete coverage trajectory through this grid, which is generated as a zigzag pattern,sweeping row-by-row while alternating directions. This sweeping pattern not only ensures total area coverage but also minimizes transition time and sharp turning, which are physically costly for UAVs.

The drone's control input at each timestep consists of a two-dimensional continuous action vector (dx, dy), bounded within $[-1, 1]$, representing directional movement. Before this action is applied to the system, it is perturbed to account for environmental noise. This perturbation is modeled as additive Gaussian wind disturbance, drawn from a zero-mean anisotropic distribution with stronger variance along the x-axis to simulate lateral gusts. The disturbed action is then passed through a momentum-based update to simulate UAV inertia. Specifically, the drone's velocity is updated using an exponential smoothing function with a momentum factor $\gamma = 0.8$, which reflects the inertial dampening present in real quadrotor motion. The resulting velocity is applied to the UAV's position, which is clipped to remain within the grid boundaries. The state observed by the policy at each timestep includes the current normalized UAV position, the achieved goal (i.e., current position in absolute coordinates), and the next desired waypoint along the predefined zigzag path. This triplet of information forms a structured observation dictionary, enabling the policy to operate in a goal-conditioned learning setup. Each episode begins with the drone at the origin, and a binary matrix tracks previously inspected grid cells, allowing the environment to detect redundant coverage.

The reward signal at each timestep is composed of multiple terms that serve both task-level and behavioral objectives. A positive reward is granted upon reaching a goal waypoint. Penalties are assigned when revisiting previously inspected cells, deviating from the current

Algorithm 2: Goal-Conditioned PPO for UAV Zigzag Path Tracking

Input: Grid size G, zigzag path $\mathcal{G}$, FOV, overlap, wind noise σ, momentum γ; PPO hyperparameters: learning rate α, clip ϵ, batch size, γ, λ, timesteps T

Output: Waypoint-tracking policy and coverage results

Initialize UAV position $\mathbf{pos} \leftarrow [0, 0]$, velocity $\mathbf{v} \leftarrow [0, 0]$, coverage map C;
Generate zigzag goal list $\mathcal{G} = \{g_1, g_2, \ldots, g_n\}$ and set $goal\_idx \leftarrow 0$;
Initialize actor $\pi_\theta(a|s, g)$ and critic $V_\phi(s, g)$;
for *each timestep* t *in* T **do**
  Construct $s_t = [\text{pos}, g_t^{achieved}, g_t^{desired}]$;
  Sample action $a_t \sim \pi_\theta(a|s_t, g_t)$;
  Apply wind: $w \sim \mathcal{N}(0, \sigma^2)$; set $a_t^{dist} = \text{clip}(a_t + w)$;
  Update velocity: $\mathbf{v}_t = \gamma\, \mathbf{v}_{t-1} + (1 - \gamma)\, a_t^{dist}$;
  Update position: $\mathbf{pos}_t = \text{clip}(\mathbf{pos}_{t-1} + \mathbf{v}_t)$;
  Compute reward r_t: $+1$ if goal reached; -0.1 revisit; $-0.01 \cdot \|s_t - g_t\|_2$; -0.001 step;
  Store $(s_t, a_t, r_t, s_{t+1}, \pi_\theta(a_t|s_t), V_\phi(s_t))$;
  if *buffer full* **then**
    Estimate advantages $\hat{A}_t$ with GAE;
    Update policy with clipped PPO loss
    $\mathcal{L} = \min\big(r_t(\theta)\hat{A}_t,\ \text{clip}(r_t(\theta), 1 - \epsilon, 1 + \epsilon)\hat{A}_t\big)$;
    Update value with MSE loss $\mathcal{L}_{val} = (V_\phi(s_t) - \hat{V}_t)^2$ and apply entropy bonus;
  end
end
Output coverage results;

goal, or simply continuing to fly without meaningful progress. This reward formulation, detailed in Equation (8), encourages the UAV to learn robust, efficient, and coverage-focused behavior under uncertainty. The UAV agent is trained using PPO, as implemented in the Stable-Baselines3 library. A Multi-Input Multilayer Perceptron (MLP) policy architecture is employed to jointly process the raw observation and the embedded representation of the current goal. The training pipeline includes a logging wrapper (`VecMonitor`) to track episode statistics, as well as an evaluation callback to perform periodic validation and apply early stopping when improvement stagnates. The full training loop,including state construction, action sampling, wind perturbation, velocity updates, and PPO optimization—is formally presented in Algorithm 2.

The PPO hyperparameters used in training are presented in Table 3. These values were selected based on empirical tuning and standard PPO guidelines for stability and convergence. The agent is trained for a total of 100,000 timesteps, with evaluations conducted every 5,000 steps to assess performance. Training is terminated early if no improvement is detected across three consecutive evaluation windows, ensuring efficient resource usage and convergence toward optimal inspection behavior.

This section presents the performance analysis of the proposed Goal-Conditioned PPO framework for UAV-based inspection under dynamic disturbances such as wind. The evalu-

Table 3: PPO Training Hyperparameters

Parameter	**Value**
Grid Size	10×10
Wind Strength	$\sigma = 0.3$
Momentum Factor	$\gamma = 0.8$
PPO Algorithm	SB3 PPO
Policy Architecture	MultiInput MLP
Learning Rate	3×10^{-4}
Discount Factor (γ)	0.99
GAE Lambda	0.95
Clipping ϵ	0.2
Entropy Coefficient	0.01
Batch Size	64
Steps per Rollout	1024
Max Timesteps	100,000
Evaluation Frequency	Every 5000 steps
Early Stopping Strategy	Stop after 3 no-improvement evals

ation highlights cumulative reward trends, inspection path visualizations, and UAV behavioral progression throughout the training process.

6.2. Learning Performance: Reward Convergence

Figure 3 presents the evolution of cumulative episode rewards for both the training and testing phases across 900 episodes. The blue curve reflects the cumulative reward received during training, while the red curve corresponds to evaluation performance under deterministic policy execution without exploration noise.

In the initial episodes (0–150), the training agent undergoes extensive exploration, as indicated by high reward volatility and the presence of many negative returns. This is expected due to the stochasticity of early PPO policy updates and the challenge of operating in a disturbed environment with wind noise and momentum. During this stage, the agent has not yet developed strategies for stabilizing its motion or aligning its trajectory with the inspection goals. The reward signal, shaped to penalize deviation, redundancy, and wandering, pushes the agent toward more disciplined behavior over time.

From episode 150 onward, the training reward exhibits a clear upward trend. The learning agent begins to internalize key behavioral corrections such as reducing overlap, minimizing distance to goal, and converging more quickly to the next waypoint. This progression highlights the effectiveness of the dense reward shaping terms in driving structured behavior even under continuous control and dynamic disturbances. By episode 400, the reward curve stabilizes with significantly reduced variance, indicating that the policy has transitioned from exploration to exploitation. From episode 450 onward, the agent maintains consistently high training performance, reflecting policy convergence to a locally optimal inspection strategy.

The test curve remains relatively high and stable across all episodes, showing that the trained policy generalizes well to unseen conditions. Unlike the training trajectory, it lacks the early noise from policy updates and exploration sampling, which is why the red curve appears flatter and more consistent. Notably, the testing agent demonstrates early-stage robustness, likely due to the reward formulation's ability to encode corrective behavior patterns, particularly under wind-induced disturbances and momentum effects. This indicates that the policy is not simply overfitting to training rollouts, but has learned transferable strategies that are invariant to trajectory-level noise.

Overall, the alignment between the converged training and stable testing performance confirms that the Goal-Conditioned PPO framework effectively enables the UAV to learn smooth, efficient, and disturbance-resilient inspection behavior. The reward structure, wind modeling, and policy architecture together contribute to a learning process that is both sample-efficient and generalizable in complex, partially stochastic environments.

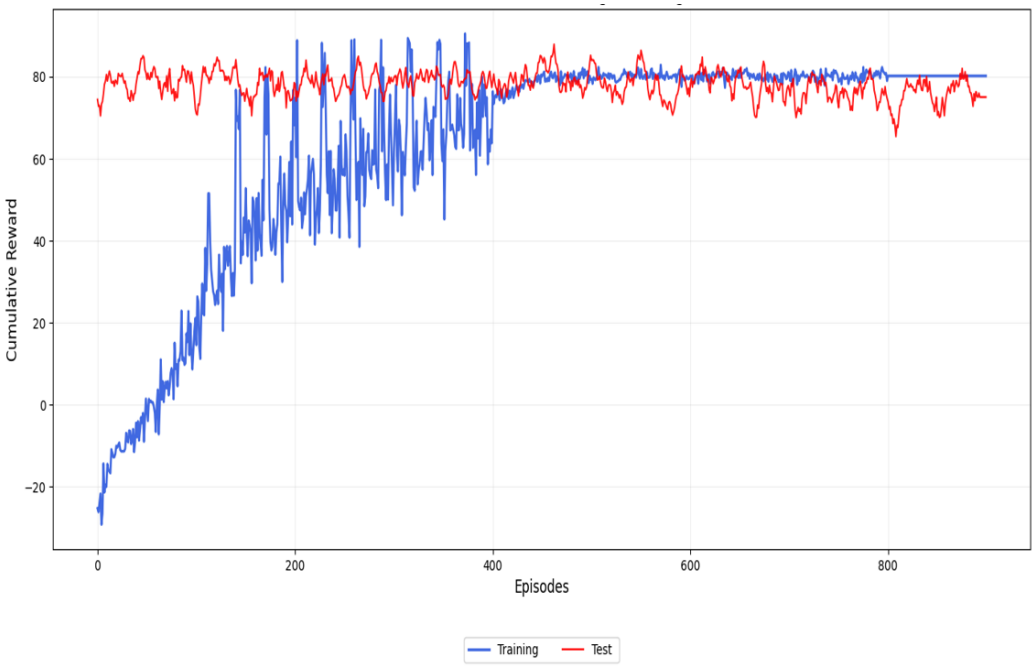
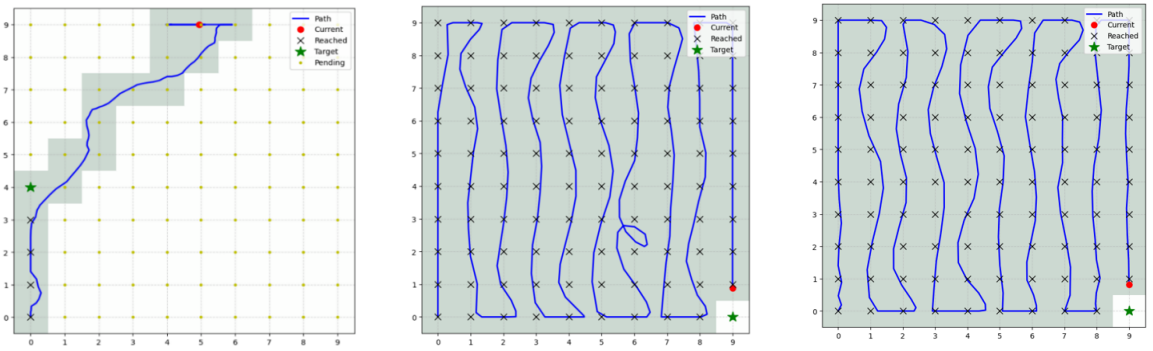


Figure 3: Cumulative reward during training and testing over 900 episodes.

6.3. Behavioral Progression: From Random to Structured Coverage

To qualitatively assess the UAV's learning progression and validate its ability to correct motion under disturbances, we visualize the trajectory followed by the agent at different stages of training. The three subfigures in Figure 4 illustrate the inspection path and waypoint coverage during (a) an early training episode, and (b–c) two evaluation scenarios after convergence.

Figure 4(*a*) represents a single episode from the early training phase. At this point, the UAV has not yet learned to follow the desired zigzag inspection path. Its trajectory appears



(*a*) UAV behavior during early training episode (*b*) UAV behavior during test scenario 1 (*c*) UAV behavior during test scenario 2

Figure 4: UAV behavior during (a) early training episodes and (b) test scenario N°1 and (c) test scenario N°2.

disorganized and erratic, with evident gaps between the planned goal waypoints and the actual reached positions. The drone often deviates from the intended coverage zone and fails to adapt to the effect of wind disturbances. For example, it misses several intermediate inspection targets and exhibits hesitation or looping behavior due to the momentum effect and lack of learned correction strategies. The inspection remains incomplete, and the drone's inability to align its control policy with goal-directed motion is clearly evident.

After convergence, we test the trained policy under two different wind disturbance scenarios. Figures 4(*b*) and 4(*c*) capture complete inspection episodes using the final policy in unseen evaluation runs. In both cases, the UAV exhibits highly structured motion, closely following the predefined zigzag trajectory and successfully visiting all inspection points. Despite external wind disturbances, the drone learns to adjust its heading and velocity vector to maintain alignment with the target path.

In test scenario 1 (Figure 4(*b*)), a significant disturbance affects the drone's trajectory near waypoint $(6, 5)$. This causes a brief deviation from the path. However, the agent successfully recognizes the drift and corrects its motion to rejoin the inspection path shortly after. While the trajectory becomes slightly warped in that region, the agent eventually completes the loop and continues inspection, reaching all remaining waypoints and achieving full area coverage.

In test scenario 2 (Figure 4(*c*)), we observe multiple points of disturbance and correction. At position $(1, 5)$, the drone is affected by wind and briefly veers off-course, but it manages to return to the trajectory around $(1, 2)$. A similar event occurs near $(3, 4)$, where the UAV compensates for lateral deviation and reorients its path around $(3, 2)$, ultimately regaining full alignment with the zigzag sweep. These cases demonstrate the UAV's learned robustness and its ability to self-correct while preserving inspection completeness.

Overall, these visualizations confirm that the trained policy enables the UAV to perform structured, reliable, and disturbance-resilient inspection. The agent is no longer reactive

or goal-agnostic but demonstrates anticipatory motion planning, minimal redundancy, and full coverage performance even under aerodynamic uncertainty.

7. Conclusion

In this work, we proposed a GCRL framework powered by PPO to enable autonomous UAV-based inspection of PV panel arrays. Unlike traditional waypoint-following methods, the agent dynamically adapts its path in response to real-time goal updates and environmental disturbances such as wind.

Through extensive training and evaluation, the learned policy demonstrated the ability to ensure full panel coverage with minimal image overlap, while maintaining inspection accuracy and robustness under uncertainty. Visual and quantitative results confirmed that the UAV evolves from random exploration to highly structured and goal-driven behavior, capable of correcting deviations and tracking optimal zigzag patterns across the entire grid. This adaptability reflects the strength of GCRL in real-world inspection scenarios where disturbances are inevitable.

While this study uses a controlled simulation environment, future work will aim to bridge the gap between simulation and deployment by transitioning to more realistic platforms such as AirSim, which offer high-fidelity physics and sensor modeling. This will allow for more accurate validation of the learned policies under near-real-world conditions. Eventually, the framework will be extended toward real UAV deployments, incorporating adaptive goal scheduling and live feedback from onboard anomaly detection models. This enhancement would allow the UAV not only to follow precomputed optimal inspection paths, but also to dynamically prioritize regions of interest, thereby improving both inspection efficiency and diagnostic accuracy in operational PV systems.

References

Carrot. Drone gsd and overlap calculations, 2024. URL https://carrot.com/blog/drone-gsd-and-overlap-calculations/. Accessed: 2024-01-10.

Shaonan Chen, Yuhong Mo, Xiaorui Wu, Jing Xiao, and Quan Liu. Reinforcement learning-based energy-saving path planning for uavs in turbulent wind, 2024. URL https://www.mdpi.com/2079-9292/13/16/3190.

Imen Habibi. Design of drone-based inspection system for industrial applications. Master's thesis, University of Waterloo and LaRINa (University of Carthage), Waterloo,Tunisia, September 2024. URL https://drive.google.com/drive/folders/1ceZNy3joqVAz8KVBHR7J9TVTMX67fxX0?usp=sharing.

Imen Habibi, Ikbal Chammakhi Msadaa, and Khaled Grayaa. Exploring drone-based inspection for detecting anomalies in pv using darknet and image colorization. In *2024 IEEE/ACS 21st International Conference on Computer Systems and Applications (AICCSA)*, pages 1–8. IEEE, 2024.

Hammer Missions. How to decide flight altitude for drone mapping and inspection, n.d. URL https://www.hammermissions.com/post/

how-to-decide-flight-altitude-for-drone-mapping-and-inspection. Accessed: 2024-03-05.

Hyeonmin Kim, Jongkwan Choi, Hyungrok Do, and Gyeong Taek Lee. A fully controllable uav using curriculum learning and goal-conditioned reinforcement learning: From straight forward to round trip missions. *Drones*, 9(1), 2025.

GyeongTaek Lee, KangJin Kim, and Jaeyeon Jang. Real-time path planning of controllable uav by subgoals using goal-conditioned reinforcement learning. *Applied Soft Computing*, 146:110660, 2023. doi: 10.1016/j.asoc.2023.110660.

Marco Andrés Luna, Mohammad Sadeq Ale Isaac, Miguel Fernandez-Cortizas, Carlos Santos, Ahmed Refaat Ragab, Martin Molina, and Pascual Campoy. Spiral coverage path planning for multi-uav photovoltaic panel inspection applications. In *2023 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 679–686. IEEE, 2023.

Sweety Nagar and Praveen Kumar Rai. Anomaly detection and classification in solar panels using infrared image and deep learning. In *2024 2nd International Conference on Disruptive Technologies (ICDT)*, pages 1554–1557. IEEE, 2024.

Andrés Pérez-González, Nelson Benítez-Montoya, Álvaro Jaramillo-Duque, and Juan Bernardo Cano-Quintero. Coverage path planning with semantic segmentation for uav in pv plants. *Applied Sciences*, 11(24):12093, 2021. URL https://www.mdpi.com/2076-3417/11/24/12093.

Patrick H. Silberberg. Aircraft inspection by multirotor uav using coverage path planning. Master's thesis, Air Force Institute of Technology, Wright-Patterson Air Force Base, Ohio, 2018. URL https://apps.dtic.mil/sti/pdfs/AD1065375.pdf.