Article

Vision-Guided Tracking and Emergency Landing for UAVs on Moving Targets

Yisak Debele [1], Ha-Young Shi [1], Assefinew Wondosen [1], Henok Warku [2], Tae-Wan Ku [3,*] and Beom-Soo Kang [4,*]

1 Industrial Liaison Innovation Center, Pusan National University, Busan 46241, Republic of Korea; yisaktol@pusan.ac.kr (Y.D.); shy621@pusan.ac.kr (H.-Y.S.); wondebly@pusan.ac.kr (A.W.)
2 Department of Electronic Engineering, Interdisciplinary Program in IT-Bio Convergence Systems, Chosun University, Gwangju 61452, Republic of Korea; heniteg@chosun.kr
3 Department of Green Transportation System Design, Pusan National University, Busan 46241, Republic of Korea
4 Department of Aerospace Engineering, Pusan National University, Busan 46241, Republic of Korea
* Correspondence: longtw@pusan.ac.kr (T.-W.K.); bskang@pusan.ac.kr (B.-S.K.)

Abstract: This paper presents a vision-based adaptive tracking and landing method for multirotor Unmanned Aerial Vehicles (UAVs), designed for safe recovery amid propulsion system failures that reduce maneuverability and responsiveness. The method addresses challenges posed by external disturbances such as wind and agile target movements, specifically, by considering maneuverability and control limitations caused by propulsion system failures. Building on our previous research in actuator fault detection and tolerance, our approach employs a modified adaptive pure pursuit guidance technique with an extra adaptation parameter to account for reduced maneuverability, thus ensuring safe tracking of moving objects. Additionally, we present an adaptive landing strategy that adapts to tracking deviations and minimizes off-target landings caused by lateral tracking errors and delayed responses, using a lateral offset-dependent vertical velocity control. Our system employs vision-based tag detection to ascertain the position of the Unmanned Ground Vehicle (UGV) in relation to the UAV. We implemented this system in a mid-mission emergency landing scenario, which includes actuator health monitoring of emergency landings. Extensive testing and simulations demonstrate the effectiveness of our approach, significantly advancing the development of safe tracking and emergency landing methods for UAVs with compromised control authority due to actuator failures.

Keywords: multirotor UAVs; moving target; vision-based landing; pure pursuit; emergency landing

1. Introduction

Unmanned Aerial Vehicles (UAVs) are witnessing a rise in applications, with a notable rise in consumer use in industries like agriculture and construction leverage UAVs for tasks such as crop monitoring and infrastructure inspections. Delivery services are exploring UAVs for more efficient transportation, especially in remote areas. UAVs play a crucial role in emergency services, aiding in search and rescue missions, disaster response, and environmental monitoring. Military applications continue to advance, focusing on surveillance, reconnaissance, and defense [1].

The importance of this technological progress goes beyond just these applications. Mastering the skill of tracking and landing on moving targets is crucial for UAVs, particularly multirotor, as it enhances mission capabilities and improves versatility and efficiency, as shown in Figure 1 [2]. This proficiency proves essential in various scenarios. In search and rescue operations, envision a UAV autonomously landing on a moving ship in rough seas [3]. The ability to track and land on moving platforms enables UAVs to navigate and operate in dynamic environments that are otherwise inaccessible by conventional means. In military operations, UAVs can strategically land on moving bases or vehicles for tasks such

as refueling, rearming, or data exchange, thereby extending their operational range and enhancing flexibility in combat situations [4]. In logistics and delivery, precise landings on moving platforms, like trucks or trains, has the potential to revolutionize last-mile delivery, especially in remote or disaster-stricken areas [5]. Additionally, this capability is invaluable in dynamic surveillance scenarios, where UAVs can track and land on moving objects, such as wildlife or criminal suspects, providing real-time aerial footage from unique vantage points. Furthermore, the concept extends to autonomous refueling, where UAVs can land on refueling stations, effectively prolonging their flight time and enabling continuous operation for extended periods.
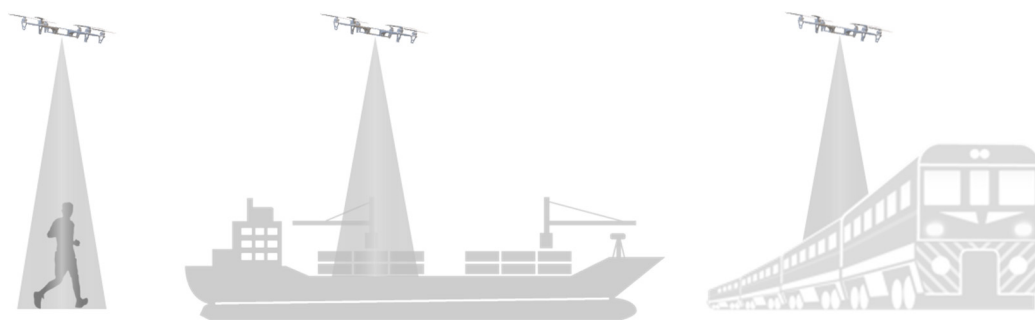
Figure 1. Application of UAV tracking moving object.

Various researchers propose methods for multirotor UAV tracking and landing on moving objects. A visual-aided landing approach utilizes designated markers for tracking and landing based on the vehicle's state. Detecting the moving object on the ground and estimating its pose are crucial initial steps in this process. Visual markers such as April tags and deep learning-based models like YOLO for detecting landing pads and safe landing zones have been implemented for this purpose [6,7]. Once the ground vehicle's pose is identified, to accurately track, different researchers suggest various methods. In [8], a vision-based guidance technique utilizing pure pursuit algorithms for tracking UGVs and a logarithmic polynomial closing velocity controller for landing on moving UGVs is proposed. However, this approach does not account for the varying movement conditions of UGVs, which demand different levels of maneuverability and responsiveness from the UAV. In [9], an algorithm for autonomous landings on moving platforms employs a single camera and unfolds in three phases—search, homing, and landing, coordinated via a 'safety sphere'. Utilizing backstepping control, it enhances landing safety. The study in [10] introduced a robust deep neural network for object detection and tracking. It further enhanced the original Kalman filter by developing an iterative multi-model-based filter to predict unknown motion dynamics of the target. The system's effectiveness was confirmed through tests in complex scenarios using ROS Gazebo. However, all the above-mentioned techniques are proposed by assuming the UAV system is in a nominal state, without considering midflight control degradations and maneuverability loss.

Multirotor UAVs are susceptible to propulsion system failures, including issues with motors, propellers, and electronic speed controllers (ESCs). To address these challenges that limit UAV integration into civilian airspace, various researchers have proposed methods to safely handle emergency situations. These methods are designed to enhance the resilience of UAVs, allowing them to maintain operation or perform controlled landings in the event of component failures, thereby ensuring safer integration into populated areas. A key focus is enabling UAVs to execute emergency landings, paralleling the emergency protocols of manned aviation. This capability is crucial for safe UAV operations in over-populated areas, mitigating risks during emergencies. Conventionally, UAVs are guided to predetermined safe zones, requiring up-to-date databases and constant communication with operators [11,12]. Alternatively, proposed solutions suggest equipping UAVs with technology to autonomously process information and select appropriate landing sites [x6].

Complementing this, our research is based on another emergency landing strategy: enabling UAVs to detect and land on moving platforms, such as Unmanned Ground Vehicles (UGVs), and we aim to propose safe tracking and landing algorithms.

Numerous studies, including our prior research, have demonstrated the ability to partially regain control authority in certain types of multirotor UAVs during mid-flight propulsion system failures [13–15]. Specifically, the hexarotor configuration noted in [16] exhibits enhanced resilience to actuator failures. Our assessments of multirotor UAV controllability during such events reveal a decrease in force and moment output, thereby restricting maneuverability [17]. This diminished controllability presents challenges for emergency response tactics that demand high maneuverability and the ability to cope with disturbances [18]. For example, as depicted in Figure 2, when following a ground vehicle (UGV), the UAV might need to perform more intense maneuvers and exert additional force in situations where the UGV executes sharp and quick movements. Meeting these demands becomes challenging due to reduced control authority. Furthermore, external factors like wind during landing can cause the UAV to deviate from its intended path, leading to an inaccurate landing. Consequently, in this paper, we introduce an adaptive tracking and landing algorithm that compensates for propulsion system failures and the consequent loss of control, ensuring safe tracking and landing on moving platforms. We specifically address the following identified challenges:
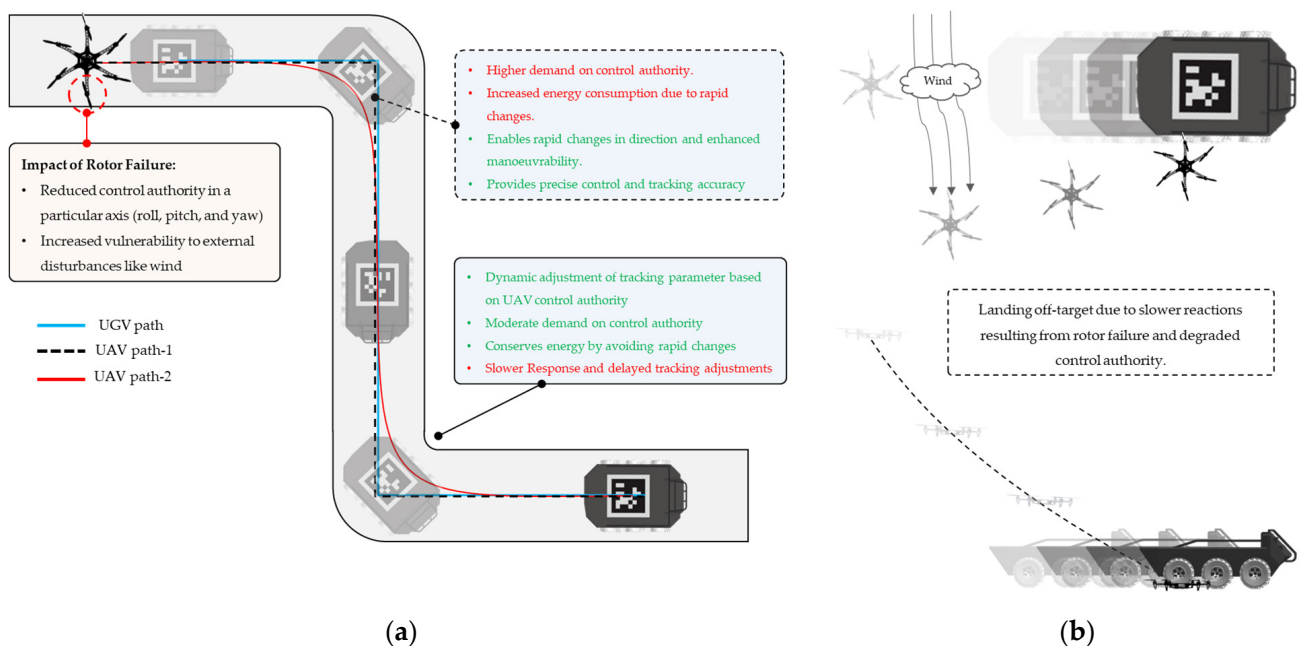

(**a**) (**b**)

Figure 2. The figure presents the problem definitions as follows: (**a**) The impact of rotor failure on UAV dynamics and tracking performance; (**b**) the influence of tracking perturbations caused by wind during landing on a moving UGV.

Problem 1: Ensuring safe UAV tracking and landing on a moving object despite limited maneuverability and an inability to counteract external disturbances like wind gusts (refer to Figure 2a).

Problem 2: Tackling the challenge of landing on a moving object amidst tracking disturbances caused by wind, and slower response times due to degraded control authority, resulting in off-target landings (refer to Figure 2b).

The primary objective of this research is to enhance the safety and effectiveness of UAVs in tracking and landing on moving targets such as UGVs during scenarios where the propulsion system fails, leading to reduced maneuverability and control. This malfunction significantly hampers safe UAV recovery. Previous studies have focused on developing methods to detect, isolate, and partially compensate for such control degradations. Build-

ing on this, we aim to develop adaptive tracking and landing algorithms that can handle reduced maneuverability and ensure safe engagements with moving objects. These algorithms will improve UAV emergency protocols by providing alternative recovery options, such as landing on moving targets, in addition to returning to a predefined location or searching for safe landing zones.

Specifically, our approach employs a modified adaptive pure pursuit guidance technique with an extra adaptation parameter to compensate for reduced maneuverability, thus ensuring safe tracking of moving objects. Additionally, in between landing processes, tracking perturbation is addressed by introducing an adaptive landing technique. This adaptive landing strategy adapts to tracking deviations and minimizes off-target landings caused by lateral tracking errors and delayed responses, using a lateral offset-dependent vertical velocity control. To obtain accurate ground vehicle pose estimation, we employ and test with both April tag detection and YOLOv5 deep learning model for landing tag detection, complemented by a Kalman filter to ensure smooth tracking, even during moments of occlusion. Extensive testing and simulations were conducted in a simulated environment with a UGV equipped with an integrated tag and a UAV with an integrated camera to validate the efficacy of our method. Additionally, our strategy was implemented on a hardware platform, yielding compelling results in detecting and estimating the ground vehicle's state during dynamic tracking scenarios. This approach reduces accident risks, adapts to control limitations, and advances tracking and emergency landing for fault-tolerant UAVs, especially in actuator failure scenarios, significantly enhancing mission success rates.

Key contributions of this paper include the following:

- Implementation of modified adaptive pure pursuit guidance technique with an extra adaptation parameter to compensate for reduced maneuverability, thus ensuring safe tracking of moving objects.
- Adaptive landing strategy that adapts to tracking deviations and minimizes off-target landings caused by lateral tracking errors and delayed responses, using a lateral offset-dependent vertical velocity control.
- Implementation of the proposed system in a mid-mission emergency landing scenario (*Bring Back Home mission*), which includes actuator health monitoring to trigger emergency landing and estimate resulting limitations in the system dynamics.

The paper is structured as follows. Section 2 provides a detailed description of the UAV and UGV system, encompassing their control architecture and mathematical model, which includes the assessment of control authority degradation in the UAV. In Section 3, the UAV's tracking and landing strategy on a moving UGV, along with vision-based UGV pose estimation, is presented. The integrated architecture's test results are subsequently discussed and summarized in Section 4. Finally, Section 5 contains the conclusions of this work.

2. UAV and UGV System

The proposed system comprises two agents: a UAV, a 6DOF aerial vehicle, and a UGV, a 3DOF ground vehicle. This section will first discuss the modeling and control system of each individual component before delving into the description and analysis of the integrated cooperative system.

2.1. Multirotor UAV System

Multirotor UAVs, characterized by their use of multiple fixed-pitch rotors, employ a propulsion system for controlled flight. This system, consisting of propellers, motors, Electronic Speed Controllers (ESCs), and batteries, generates thrust forces perpendicular to the rotor planes, determined by their spin direction and rate. The resulting reaction torques counter rotor rotations, requiring precise control for stabilization and maneuverability. This combination of force and torques grants multirotor UAVs exceptional agility and hovering capabilities.

2.1.1. Dynamics of UAV System

Developing a mathematical model for UAV flight involves assumptions like treating the centroid as the center, assuming rigidity without deformations, neglecting air resistance for low-speed scenarios, and maintaining constant mass and moment of inertia throughout the analysis [19].

Therefore, the thrust T_i and torque τ_{di} generated by the ith propeller spinning at a rotational speed ω_i, as shown in Figure 3, can be expressed as follows:

$$T_i = k_{t,i}\omega_i^2 \tag{1}$$

$$\tau_{di} = k_{d,i}\omega_i^2 \tag{2}$$

where $k_{t,i}$ and $k_{d,i}$ are generalized thrust and drag coefficients that depend on propeller geometry.
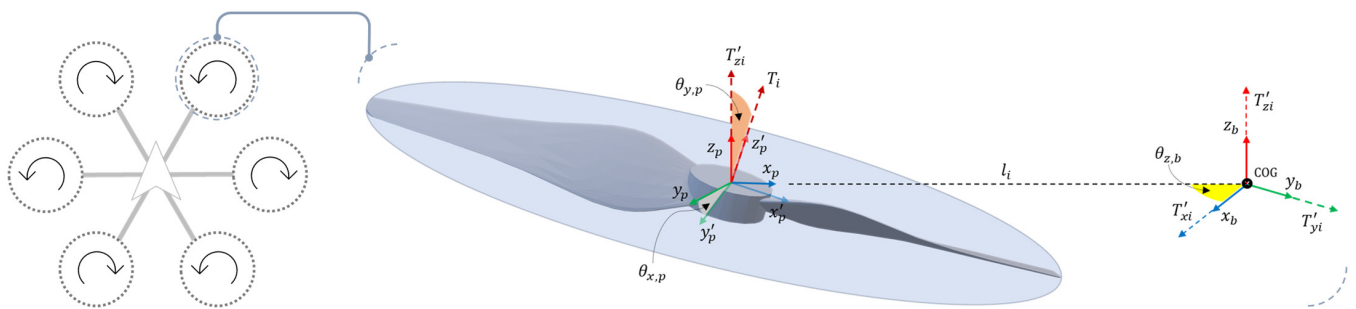

Figure 3. Thrust generation and vectorization.

For propellers arranged systematically about the z-axis of the body frame F_b in order to form a multirotor configuration system, the total thrust generated from n propellers situated with angle $\theta_{z,i}$ about body z-axis from positive x of the F_b and with tilting angles $\theta_{y,i}$ and $\theta_{x,i}$ about propeller coordinate y-axis and x-axis, respectively, that form the configuration can be given as

$$T = \sum_{i=1}^{n} T_i' \tag{3}$$

where $T_i' = \begin{bmatrix} T_{xi}' & T_{yi}' & T_{zi}' \end{bmatrix}$ is the transformed thrust due to propeller tilting by the rotation matrix, given as

$$T_i' = T_i\left(R_{Z_b}(\theta_{zb,i}) R_{Y_p}(\theta_{yp,i}) R_{x_p}(\theta_{xp,i}) e_3 \right) \tag{4}$$

Rotation moment is generated through the application of the differential thrust principle. This principle entails adjusting the speeds of individual rotors to control the rotation of the UAV. Independent control of each rotor's thrust is achieved by varying its angular velocity. The resulting differential thrust creates a torque imbalance, facilitating precise and responsive rotation due to its offset position from the center. The overarching principle of torque balance, achieved by adjusting rotor speeds to manage the total torque acting on the UAV, is fundamental for stability and controlled flight.

Let us denote the location of each rotor at distance l_i from the center rotor, which can be described as

$$\mathcal{L}_i = \begin{bmatrix} l_{ix} & l_{iy} & l_{iz} \end{bmatrix}^T = \begin{bmatrix} \cos\theta_{zb,i} \\ \sin\theta_{zb,i} \\ 0 \end{bmatrix} \times l_i \tag{5}$$

The moment force about the body frame is generated due to propeller placement, influencing thrust and drag dynamics.

$$\tau = \sum_{i}^{n} p_i \times T_i + \tau_{di} \tag{6}$$

where drag torque $\tau_{d,i}$

$$\tau_{d,i} = \begin{cases} -k_{d,i}\omega_i^2 & \text{counter clock wise, ccw} \\ k_{d,i}\omega_i^2 & \text{clock wise, cw} \end{cases} \tag{7}$$

Generally, the above formulation can be written compactly by using the effectiveness matrix, *B*, that maps actuator space to moment space $R^n \longrightarrow R^m$ as

$$\begin{bmatrix} T \\ \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = B\left(l, \theta_{zb}, \theta_{yp}, \theta_{xp}, k_d, k_t\right) \begin{bmatrix} \omega_i^2 \\ \vdots \\ \omega_n^2 \end{bmatrix} \tag{8}$$

The equation of the UAV motion can be given as

$$\dot{x} = V_u \cos\theta \cos\psi \tag{9}$$

$$\dot{y} = V_u \cos\theta \sin\psi \tag{10}$$

$$\dot{z} = -V_u \sin\theta \tag{11}$$

$$\ddot{x} = (cos\psi sin\theta cos\phi + sin\psi sin\phi)\frac{T}{m} \tag{12}$$

$$\ddot{y} = (cos\phi sin\theta sin\psi - cos\psi sin\phi)\frac{T}{m} \tag{13}$$

$$\ddot{z} = -g_o + (cos\theta cos\phi)\frac{T}{m} \tag{14}$$

$$\dot{\psi} = \frac{a_h}{V\cos\theta} \tag{15}$$

$$\dot{\theta} = \frac{a_v}{V} \tag{16}$$

$$\ddot{\phi} = \left(\frac{I_y - I_z}{I_x}\right)qr + \frac{\tau_\phi}{I_x} \tag{17}$$

$$\ddot{\theta} = \left(\frac{I_z - I_x}{I_y}\right)pr + \frac{\tau_\theta}{I_y} \tag{18}$$

$$\ddot{\psi} = \left(\frac{I_x - I_y}{I_z}\right)qr + \frac{\tau_\psi}{I_z} \tag{19}$$

where V_u is the velocity of the UAV, a_h is the horizontal velocity of the UAV, a_v is the vertical acceleration of the UAV, and $I = \begin{bmatrix} I_x, & I_y, & I_z \end{bmatrix}$, UAV rotational moment.

2.1.2. Actuator Failure and Control Degradation

An actuator failure in a multirotor UAV can lead to reduced control authority, asymmetric moments, unintended movements, and potential saturation of remaining operational actuators. This compromises stability, maneuverability, and overall control capabilities, while also increasing vulnerability to external disturbances and altering the Attainable Moment Envelope.

2.1.3. Control Degradation Assessment

In aviation, the attainable moment set represents the range of achievable moments or torques within a system, indicating the system's maximum capacity in generating moment force through permissible control inputs. This analysis is crucial for understanding the UAV's capabilities, limitations, and responsiveness to external forces or disturbances. The AMS significantly affects system performance, imposing limits on achievable time derivatives of states, constraining maneuverability, agility, and disturbance rejection capa-

bility. Limited AMS, particularly due to propulsion system failure, risks safe operation by degrading control authority [20].

From Equation (8), the effectiveness matrix that depends on the propulsion and configuration parameters can be rewritten as

$$B = f(k_t, k_t, \theta, \gamma, l) \tag{20}$$

Thus, the set of all attainable moments in its three axes (roll τ_ϕ, pitch τ_θ, yaw τ_ψ) is denoted by the Attainable Moment Set (AMS) $\Lambda \in \mathbb{R}^3$, generated within admissible control, which can be given as

$$\Lambda = \left\{ m \in \mathbb{R}^{m \times 1} \middle| m = Bu, u_{min} < u < u_{max} \right\} \tag{21}$$

where $B \in \mathbb{R}^{m \times n}$ is the effectiveness matrix determined by design parameters, mapping actuator control input to moment space, and u is admissible control within the operational range of actuators.

The effectiveness matrix B of the healthy configuration is adjusted by removing the column associated with the failed actuator, expressed as

$$B_f = B\chi \tag{22}$$

where B_f is the resulting matrix with the failed actuator contribution excluded, and χ is the actuator fault flag indicating the location of the failure. The achievable moment in all three axes, roll $\tau_{\phi f}$, pitch $\tau_{\theta f}$, yaw $\tau_{\psi f}$, after actuator failure can be given by modifying Equation (21). Therefore, the resulting operational envelope can be used to estimate and tune tracking parameters like that of the minimum lookahead distance to ensure that the UAV remains within the operational envelope, avoiding scenarios where the required moment exceeds its degraded capabilities.

2.2. UGV System

This study employs a six-wheel ground vehicle with a tag for detection to guide the UAV and serving as an emergency landing platform. In this section, the guidance, navigation, and control strategy is described to drive the UGV to the approximate location of the UAV emergency area and navigate to the destination.

2.2.1. UGV Kinematics

Each wheel is independent and independently driven so that each wheel can be controlled. The six driving wheels are not steering wheels; turns are made by changing the rotational speeds of the six wheels, and cornering radii vary depending on the rotational speed of the drive wheels and the indices of adhesion to the ground, the slip indices. The six-wheel UGV is shown in Figure 4.
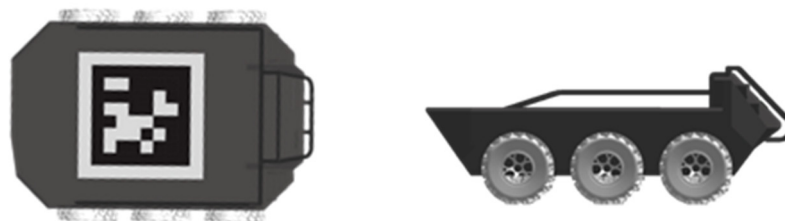

Figure 4. UGV model.

Differential drive robots are also called differential wheeled robots, and they involve calculating the appropriate speeds and directions for each wheel based on the desired

movement of the robot, such as going forward, turning at a specific angle, or following a predefined path [21].

$$
\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \cos\psi & 0 \\ \sin\psi & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V_g \\ \dot{\psi} \end{bmatrix}
\tag{23}
$$

where V_g and $\dot{\psi}$ are the tangential and the angular velocities of the vehicle:

$$
\begin{bmatrix} V_g \\ \dot{\psi} \end{bmatrix} = f \begin{bmatrix} \omega_r \\ \omega_l \end{bmatrix}
\tag{24}
$$

where ω_r is the right-side wheel's rotation rate, assuming $\omega_{r1} = \omega_{r2} = \omega_{r3}$, ω_l is the left-side wheel's rotation rate, assuming $\omega_{l1} = \omega_{l2} = \omega_{l3}$, and f is a geometrical information matrix depending on the distribution of the wheels in the system and wheel parameters.

2.2.2. UGV System Guidance Navigation and Control (GNC)

In robotics GNC, guidance plans optimal paths, and navigation monitors real-time positioning [22]. The Waypoint Navigation Method (WPN) sequentially guides the robot through waypoints. In our case, the waypoint list includes the UAV's approximate position for visibility to the UAV vision system and triggering tracking mode. Waypoints commence at the UGV home, pass through the UAV's emergency hovering location, and return home, as shown in Figure 5.
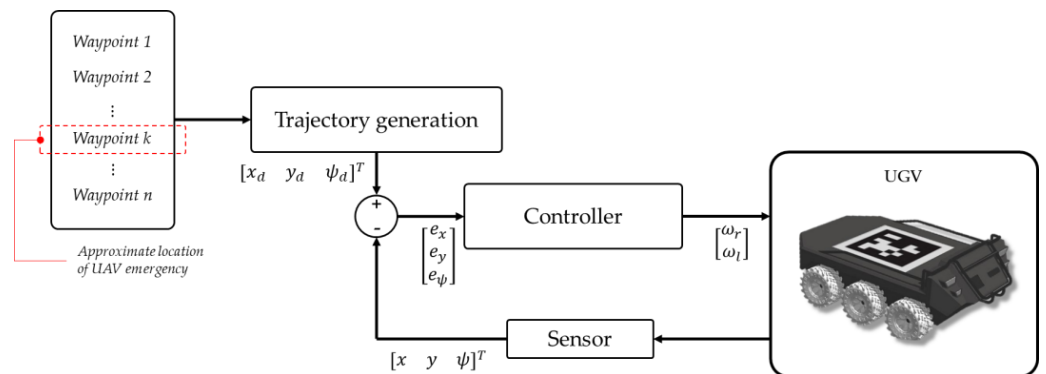

Figure 5. UGV control architecture.

To achieve smooth turning paths and generate trajectories between waypoints, we employed the Cubic Spline interpolation method. Cubic Spline interpolation constructs continuous curves that pass through or near given waypoints, creating a refined trajectory for the robot. It utilizes piecewise third-order polynomials, ensuring they pass through a set of n control points [23]. A function $f(x)$ on $[a, b]$ becomes an interpolated cubic spline function if the following two conditions are met.

$$
y_d = \begin{cases} f_1(x_d) & x_0 \le x_d \le x_1 \\ \cdots & \cdots \\ f_i(x_d) & x_{i-1} \le x_d \le x_i \\ \cdots & \cdots \\ f_n(x_d) & x_{n-1} \le x_d \le x_n \end{cases}
\tag{25}
$$

where each $f_i(x) = a_i + b_i x + c_i x^2 + d_i x^3$, $d_i \ne 0$, $i = 1, \cdots, n$.

Let us suppose that at some time instant t, the robot position is given by the coordinates x and y. Also, let us assume that at the time instant t, the orientation of the robot body is given by ψ. Our goal is to steer the robot to the point d with the coordinates (x_d, y_d) on the planned path. To arrive at the target, the velocity vector of the robot should rotate to ψ_d

such that it is in the direction of the line connecting the center of the robot and the target point d.

$$\psi_d = archta2\left(\frac{y_d - y}{x_d - x}\right) \tag{26}$$

Using the proportional controller, the direction of the UGV velocity vector can be controlled as follows.

$$\dot{\psi} = K_\psi(\psi_d - \psi(t)) \tag{27}$$

where $\psi(t)$ is the current value of the orientation of the robot and K_ψ is the proportional control gain for orientation control.

$$V_g = K_v\sqrt{(x_d - x(t))^2 + (y_d - y(t))^2} \tag{28}$$

where K_v is the proportional constant for controlling the robot's velocity.

3. Tracking and Emergency Landing Scenario on Moving Object

3.1. Emergency Landing Scenario

In this section, we develop the method for tracking using a 2D image obtained from the camera mounted on the UAV. Unfortunately, while the UAV is in action, if one of its actuators fails unexpectedly, it results in a reduction in control capability. This limits the UAV's ability to maneuver effectively, potentially causing the mission to come to a halt. As explained in Section 2, the hexarotor UAV considered in this study can maintain null controllability even with a single actuator failure. This is achieved by redistributing the control command to the remaining healthy actuators. Upon detection of a failure by the FDI system in the UAV health monitoring system, the typical immediate response is to enter the recovery phase, where the UAV hovers and attempts to regain stability. After achieving stability, if the UAV is unable to return home or continue its mission due to degraded control, the vision and landing tag detection system is activated to initiate the search for safe landing options.

The architecture of the integrated system of UAV-UGV for making emergency landing on a moving UGV is presented in Figure 6. In the very beginning, the UAV with all components working fine is given a mission to perform in scenarios like that of surveying or urban traffic monitoring. During this operation, unintended component failure, specifically actuator failure, will result in controllability degradation that limits the maneuverability, hence halting mission completion. Once the occurrence of the failure is detected by the FDI system of the UAV health monitoring system, the immediate action commonly taken is hovering and trying to regain stability, which is known as the recovery phase. The real-time fault detection and isolation system is adopted from our previous work, as described in [24]. As discussed in Section 2, the type of hexarotor UAV under consideration has the capability of null controllability for a single actuator failure in the system by redistributing the control command to the other healthy ones. In the meantime, the vision and landing tag detection system is triggered by this signal and starts searching for the existence of safe landing options.

Once the UAV system decides to search around for emergency landing, the approximate location of the current UAV location is sent to UGV. This request activates the UGV system and includes the sent location point from the UAV into the waypoint list that guides and returns the UAV home by generating a trajectory. The UGV navigates to the UAV location upon triggering the tracking mode when seen by the UAV camera. Despite degraded control authority, the tracking algorithm parameters are reconfigured to accommodate the loss of controllability and reduce the required turning rate.

Once the UAV accurately tracks the UGV, it enters landing mode, descending while maintaining precise tracking. The proposed strategy employs adaptive landing logic, adjusting descending speed based on tracking precision. If tracking precision decreases, the

logic slows the descent; conversely, if the UAV precisely tracks the UGV, the logic increases the descending speed.
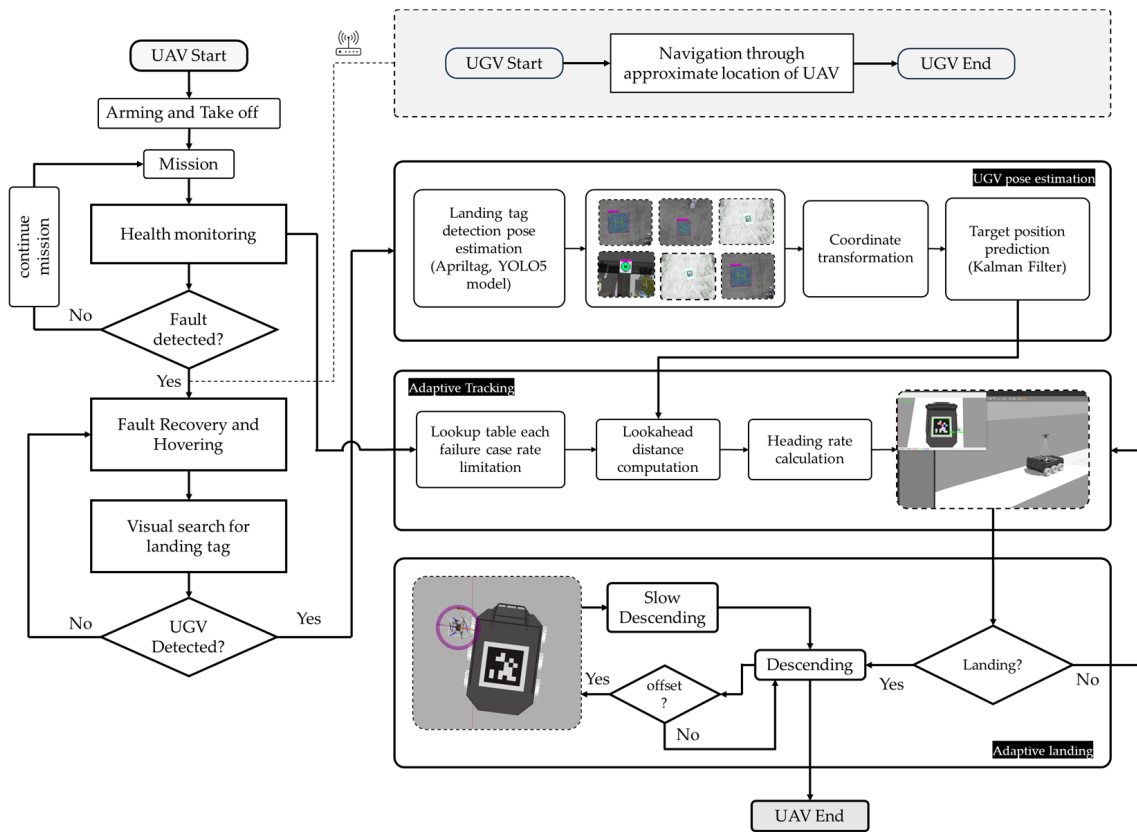

Figure 6. Proposed system architecture for emergency landing scenario "bring back home" mission.

3.2. UGV Pose Estimation

In the tracking process, the precise and swift detection of the landing pad is crucial. This study employs two types of landing tags, namely April tags (see Figure 7) and standard H-type landing pads (see Figure 8), for estimating the UGV pose. While the primary focus is not an exhaustive examination and comparison of the performance of these markers, the proposed framework was tested with both April tags and standard H-type landing pads to ensure the generality and applicability of the system.
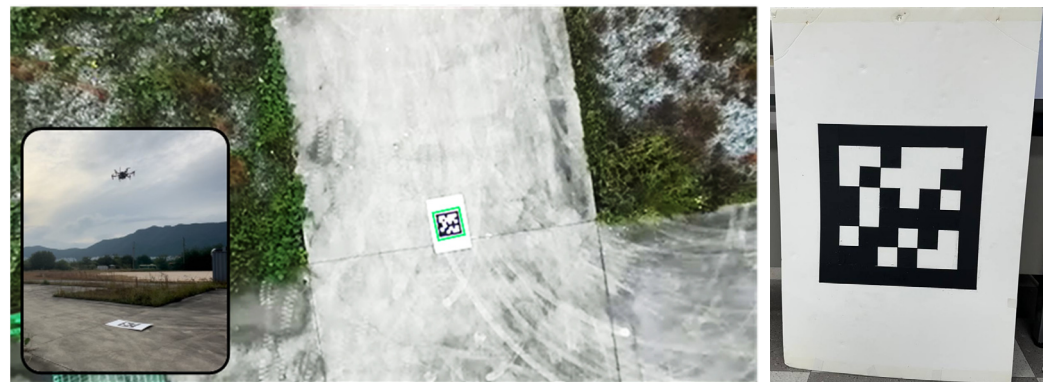
Figure 7. April tag-based UGV pose estimation.
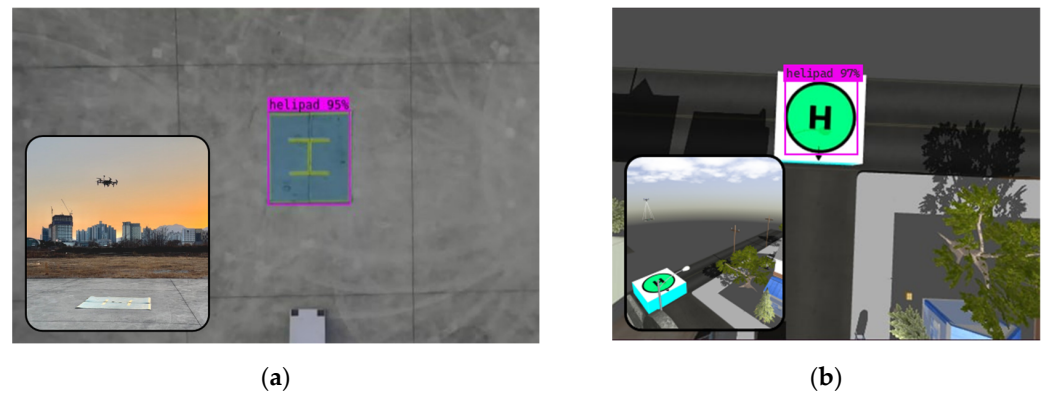
Figure 8. Standard landing pad detection using YOLO v5 machine learning model and pose estimation. (**a**) Real flight experiment. (**b**) Simulation environment.

April tag, a widely used fiducial, utilizes 2D coded information on a tag to provide the camera with the marker's position and orientation. It identifies four-sided regions with a darker interior, computes pixel gradients, groups them into components, and fits lines using a weighted least squares technique. These lines form quads with a valid code scheme, and the system extracts the 6-DOF pose of the tag in the camera frame using homography and intrinsic estimation. April tag is advantageous for its low cost and computational simplicity. However, its use as a localization system may result in erroneous localization due to factors such as viewing angle, distance, and camera rotation [25].

Furthermore, in this investigation, the UGV's state is ascertained utilizing standard landing helipads through the implementation of YOLOv5, a rapid and highly accurate object detection algorithm using a deep learning model developed in PyTorch. The selected deep learning model is trained with a diverse dataset that extends beyond helipads to encompass persons and cars. This comprehensive approach ensures versatile and precise state estimation, combining standard helipad tags, April tags, and a machine learning model. The dataset encompasses various helipad tags, illustrating their variations, along with drone images from different perspectives, as depicted in Figure 7. The dataset, comprising 901 images (including 223 person images, 433 helipad images, and 245 car images), undergoes labeling and augmentation using the Roboflow API and is partitioned into 70% training and 30% test images.

The relative location of the detected helipad relative to the UAV COG was computed and subsequently transformed from camera frame to UAV body frame. The estimation of 3D pose of the helipad on the moving UGV with respect to the camera is summarized in Figure 9. If (X, Y, Z) is a 3D point in known local coordinate space, we can calculate in the camera coordinate system (x_c, y_c, z_c) by the rotation matrix R and translation matrix t as follows:

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = [R|t] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{29}$$

In the absence of radial distortion, the coordinates (X_c, Y_c) in the image coordinates are given by [26–32]

$$\begin{bmatrix} X_c \\ Y_c \\ 1 \end{bmatrix} = s \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} \tag{30}$$

where f_x, and f_y are focal length, (c_x, c_y) is the optical center, and s is the scaling factor.

After obtaining the coordinate of the bounding box from the detection result, the center of the tag was calculated and transformed to the camera coordinates, followed by transformation to the UAV body coordinate frame. Subsequently, for tracking purposes, the relative position and orientation of the tag will be transformed to the local coordinate system.
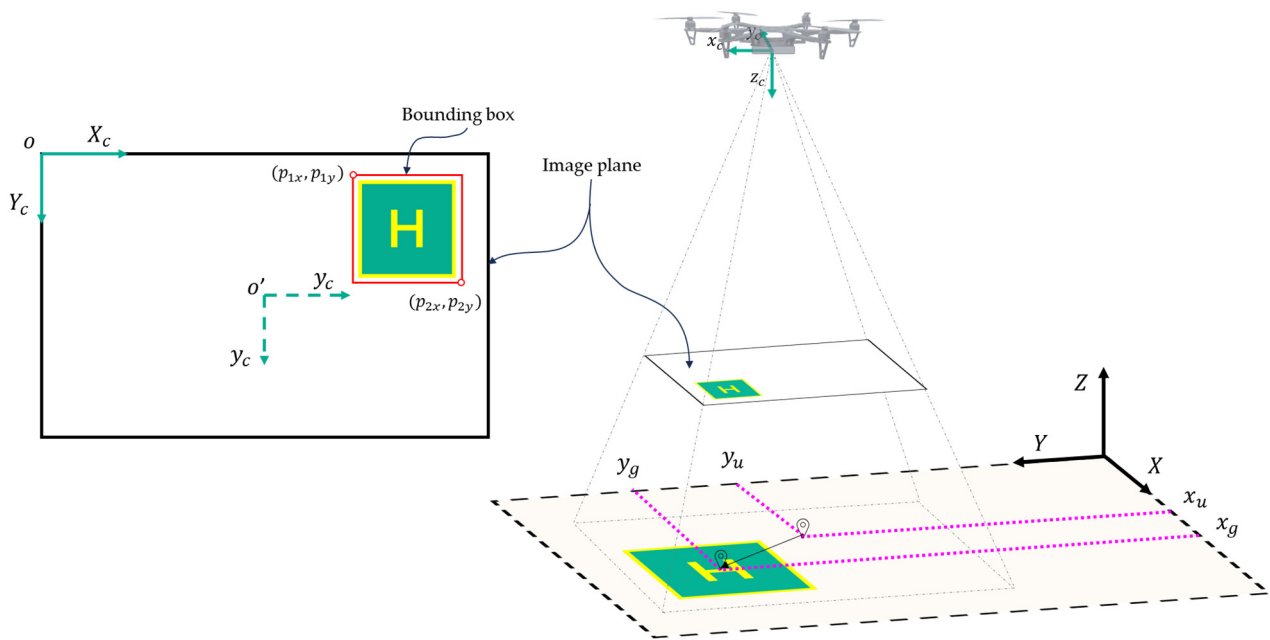
Figure 9. Camera frame and pose estimation.

However, there are scenarios that make the position and orientation noisy and intermittent in estimation, as a result of occlusion. During the landing process, the target may be partially or fully occluded, or may move out of the field of view. Hence, we use Kalman filter (KF) to estimate the target parameters. A Kalman filter is designed to predict the position of the UGV at the next time step. This helps in cases of temporary loss of image or target occlusion. The Kalman filter determines the position whether it is detected in the image or not. For such scenarios, the Kalman filter uses the previous estimate to give the best estimate of the position. So, when the position is estimated, the Kalman filter predicts its state and then uses fresh measurements to correct its state and produce a filtered position. The state of the filter is represented by $\hat{x}_{k|k}$, and the two steps for the filter, Predict and Update, are as follows.

Predict:

$$\begin{cases} \hat{x}_{k|k-1} = F_k \hat{x}_{k-1|k-1} + B_k u_k \\ \hat{P}_{k|k-1} = F_k \hat{P}_{k-1|k-1} F_k^T + Q_k \end{cases} \tag{31}$$

Update:

$$\begin{cases} y_k = G_k x_{k-1|k-1} + n_k \\ S_k = G_k P_{k|k-1} + R_k \\ K_k = P_{k|k-1} H_k^T S_k^{-1} \\ \hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k y_k \\ P_{k|k} = (1 - K_k H_k) P_{k|k-1} \end{cases} \tag{32}$$

where F_k, is the state estimation matrix, B_k is the control input applied to the control vector u_k, H_k is the observation matrix, n_k is the observation noise, S_k is the innovation covariance matrix, and K_k is the Kalman gain.

3.3. Target Tracking Using Adaptive Pure Pursuit Guidance

The Pure Pursuit approach establishes a continuous curved trajectory connecting the current UAV position to a point predetermined at a specified distance ahead of the UGV (see Figures 9 and 10) [33–37]. A reduced lookahead distance necessitates more precise path tracking, improving the system's ability to follow paths with higher curvature. A smaller lookahead distance leads the vehicle to return to the path more aggressively when separated. Conversely, a longer lookahead distance enables the vehicle to initiate turning

before reaching a curve, preventing overshooting upon return and resulting in smoother trajectories. In the event of actuator failure, reducing controllability and maneuverability, the lookahead distance Is dynamically adjusted to avoid sharp turns that require high turning moment force.
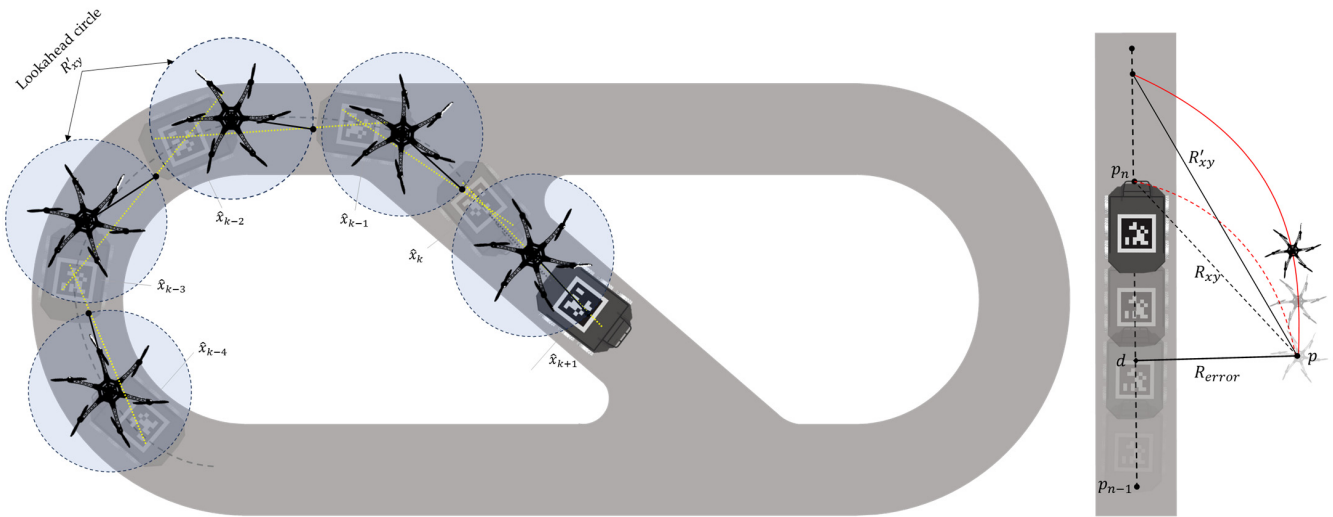

Figure 10. Visualization of UAV tracking UGV and adaptive pure pursuit tracking.

Let us take lookahead point p_n taken from in front of the estimated UGV location, given by (x_t, y_t, z_t), previous lookahead point p_{n-1}, given as $(x_{t-1}, y_{t-1}, z_{t-1})$, and current UAV position p, given as (x_u, y_u, z_u). From the current state of the UAV, the lookahead can be calculated as (see Figure 11)

$$R = \sqrt{(x_t - x_u)^2 + (y_t - y_u)^2 + (z_t - z_u)^2} \tag{33}$$

$$R_{xy} = \sqrt{(x_t - x_u)^2 + (y_t - y_u)^2} \tag{34}$$
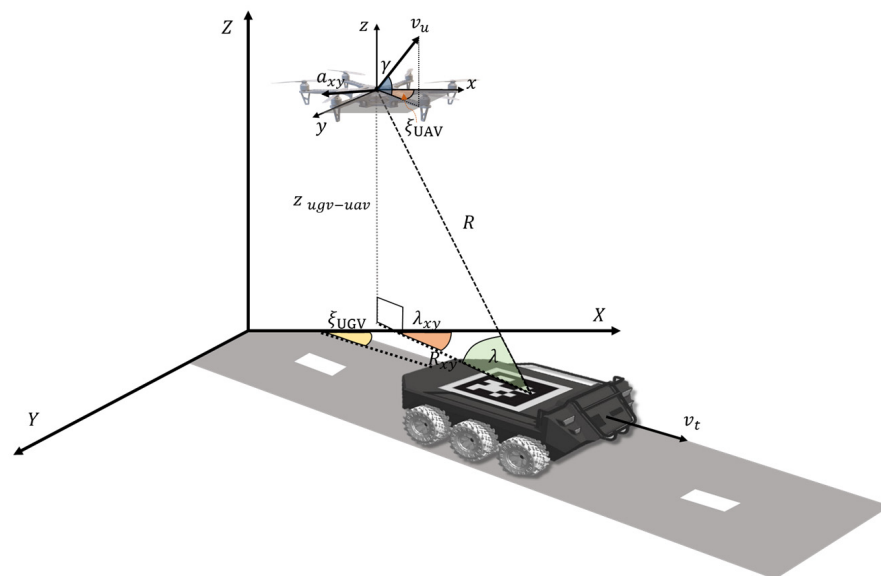
$$R_z = z_t - z_u \tag{35}$$

Figure 11. Pure pursuit tracking formulation.

As covered in Section 2, actuator failures in UAVs constrain the UAV's attitude control authority. Consequently, to mitigate substantial attitude moments during sharp UGV turns and address significant deviations between UAV and UGV positions, the lookahead distance must be fine-tuned. This adjustment facilitates smoother and slower turns, reducing the demand for yawing moments. In the adaptive algorithm, the lookahead distance is no longer a constant value; instead, it adapts based on the perpendicular distance between the current UAV position and the path linking p_n and p_{n-1}. From point d, the projection of p, and the line connecting p_{n-1} and p_n, the error can be obtained as

$$R_{error} = \sqrt{(x_u - x_p)^2 + (y_u - y_p)^2} \tag{36}$$

Therefore, R_{error} adjusts the lookahead distance as

$$R'_{xy} = R_{xy} + R_{error} + R_{AMS} \tag{37}$$

where R_{AMS} is a minimum lookahead distance that maintains the UAV within the range of achievable force, given in Equation (23). This parameter limits and protects the UAV from demanding more than the achievable force that the UAV can generate, as shown in Figure 12.
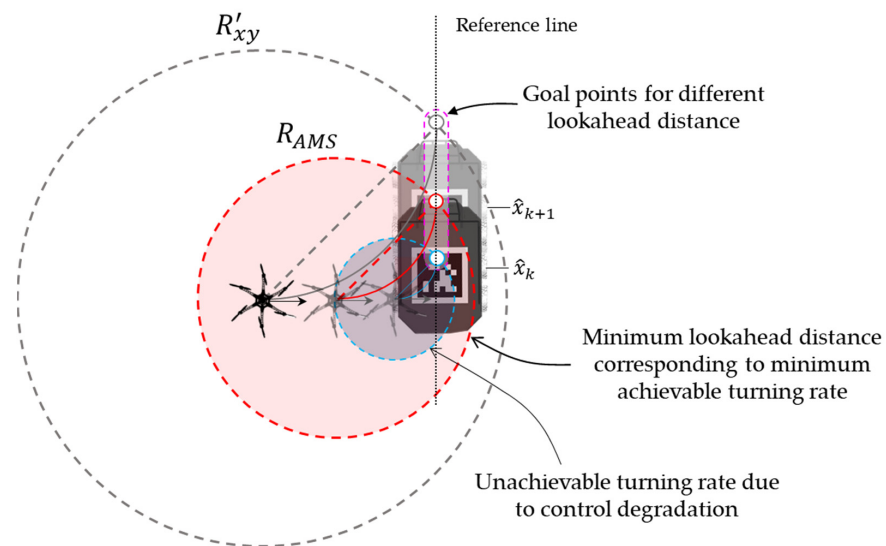

Figure 12. Adaptive tracking that considers rate limitations and achievable rate, resulting from failure in rotor.

Let us take the modified lookahead point $\widetilde{p}_n$, $\left(\widetilde{x}_t, \widetilde{y}_t, \widetilde{z}_t\right)$. The angle λ_{xy} formed by R'_{xy} and the x axis and the angle λ formed between lines R and R'_{xy} can be computed as

$$\lambda_{xy} = \tan^{-1}\left(\widetilde{y}_t - y_u / \widetilde{x}_t - x_u\right) \tag{38}$$

$$\lambda = \tan^{-1}\left(\widetilde{z}_t - z_u / R'_{xy}\right) \tag{39}$$

The require heading correction can be given as

$$\dot{R}_{xy} = V_t \cos(\xi_{ugv} - \lambda_{xy}) - V_u \cos\gamma \cos(\xi_{uav} - \lambda_{xy}) \tag{40}$$

$$R'_{xy}\dot{\lambda}_{xy} = V_t \sin(\xi_{ugv} - \lambda_{xy}) - V_u \cos\gamma \sin(\xi_{uav} - \lambda_{xy}) \tag{41}$$

Therefore, the guidance command will be

$$a_{xy} = V_u \cos \gamma \, \dot{\lambda}_{xy} - K_a \left(\xi_{uav} - \lambda_{xy} \right) \tag{42}$$

$$\dot{\xi}_{uav} = \frac{a_{xy}}{(V_u \cos \gamma)} \tag{43}$$

3.4. Adaptive Autonoumus Landing on a Moving Target

In this section, the objective is to land the vehicle smoothly on the target while persistently tracking it, as depicted in Figure 13. To meet these two requirements, we propose an adaptive descending velocity that depends on the tracking performance R_{error} and the smoothness of the landing process by introducing penalty factor α. In scenarios where R_{error} is lower ($\alpha \approx 1$), the UAV follows proportional tracking, whereas if the UAV deviates from tracking, the descending rate will be penalized, and for a large deviation from tracking, the landing will be halted and the UAV will maintain its altitude until the tracking performance of the UAV is regained. The presented pseudocode clearly shows the proposed adaptive system (see Algorithm 1).
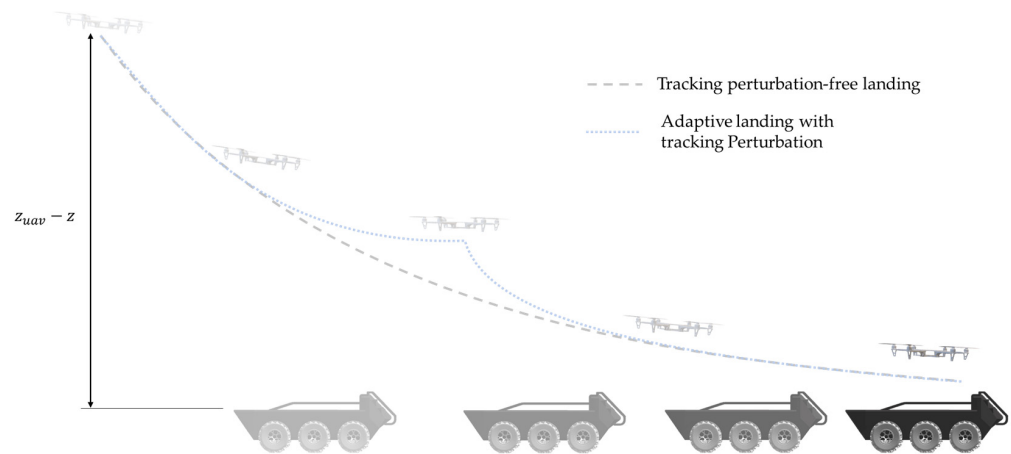

Figure 13. Adaptive landing on moving UGV.

For the current altitude of the UAV given by z_{uav} and the landing plane altitude given as z_l, the descending velocity can be given as

$$V_z = -k_z \alpha (z_{uav} - z) \tag{44}$$

Algorithm 1: Adaptive landing algorithm

    Input: UGV position offset relative to UAV position
    Output: Velocity and heading rate command
1   **Initialize**: *Landing Mode*
2   **While** *True* **do**
3   **If** UGV_detected **then**
4       $V_x, V_y, \dot{\psi} \longleftarrow$ controller_lateral ($error(x_{off}, y_{off}, \psi_{off})$)
5   **If** $z_{off} < 0.5$ m then
6          $V_z \longleftarrow 3$ m/s
7   **else**
8          $V_z \longleftarrow$ controller_vertical ($error(Z_{off}), \alpha$) $\longleftarrow \alpha \longleftarrow (R_{error}(error(x,y)))$
9   **end**
10   **end**
11   **else If** UGV_not_detected **then**
12   Increase altitude: $V_z \longleftarrow$ *trust command*
13   **end**
14 **end**

During the landing process, to slow the descent when the UAV is not precisely tracking the UGV (Equation (44)), we take into account the tracking penalty factor $\alpha = e^{-bR_{error}}$, where $0 < \alpha < 1$. As shown in Figure 14, when the UAV is precisely following the UGV, $R_{error} \rightarrow 0$, $\alpha \rightarrow 1$, and therefore, V_z will be proportional to control with gain k_z, whereas in the scenario where the tracking is not precise, $R_{error} \rightarrow \infty$, $\alpha \rightarrow 0$, this slows the descending velocity and gives time for the tracking algorithm to correct for deviation.
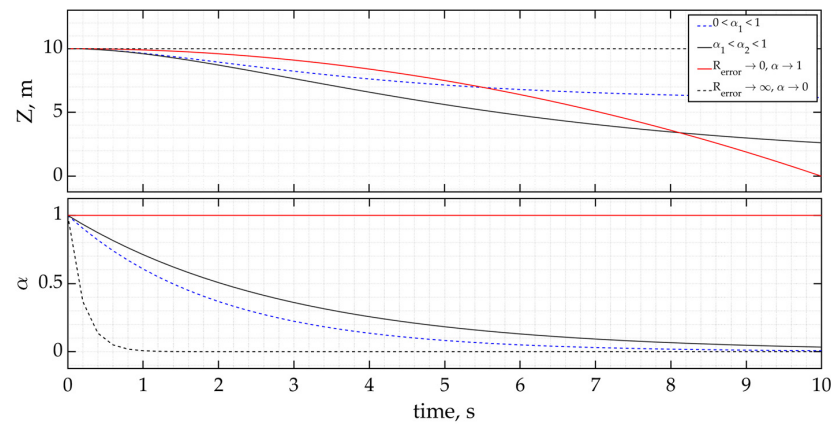
Figure 14. Smooth landing and effect of tracking performance on landing.

4. Results and Discussion

4.1. Test Environment Setup Preparation

In this research, we executed comprehensive simulation tests and flight experiments to assess a proposed architecture for tracking and emergency landing. The testing encompassed diverse scenarios mimicking real-life occurrences within a simulated environment created using GAZEBO, a dynamic 3D model simulation tool. The integration of PX4 flight controller software version stable 1.14 and ROS-based packages facilitated control over both the UAV model, equipped with six rotors and an integrated gimbal camera, and a six-wheeled ground robot. The ground robot featured a flat platform with a tag serving as a mobile landing target system.

During the experiments, the GNC system employed waypoint navigation, utilizing UGV position and speed data from the wheel speed odometer. For tracking and landing, an ROS-based package with Python code was created, integrating an April tag detection algorithm and a trained YOLOv5 model. This package subscribed to camera image data, performed inference to determine the detected object's position in image coordinates, and transformed it to camera and local coordinates. After estimating the UGV's pose, a tracking module used the pure pursuit algorithm for accurate UGV tracking. Flags were set to activate modes like "bring me home" for UGV, recovery mode, tracking mode, and landing mode for UAV touchdown on the moving UGV. Additionally, a PX4 firmware module was implemented to inject faults and reallocate control commands, with the source code compiled.

A hardware setup, illustrated in Figure 15, was prepared to test the proposed system. In addition to standard multirotor UAV components, such as motors, propellers, ESC, GPS, Pixhawk 4 autopilot, and LiPo Battery, the setup included a vertically mounted ZED stereo camera for visual perception. The NVIDIA JETSON NANO 4GB served as the companion computer for image processing. The camera connected to the companion computer via USB, and the ZED ROS wrapper package facilitated image capture and calibration. Communication between the Pixhawk 4 and NVIDIA JETSON NANO 4GB was established through MAVLINK-enabled serial ports, ensuring the exchange of telemetry data and setpoint information for the controller at a rate exceeding 2 Hz.

The integration of the vision-based target pose estimation package, tracking, adaptive landing package, and a custom fault detection and reallocation module in PX4 underwent testing across various tracking and landing scenarios. The evaluation encompassed both

healthy and degraded control conditions, with the summarized results presented in the following sections.
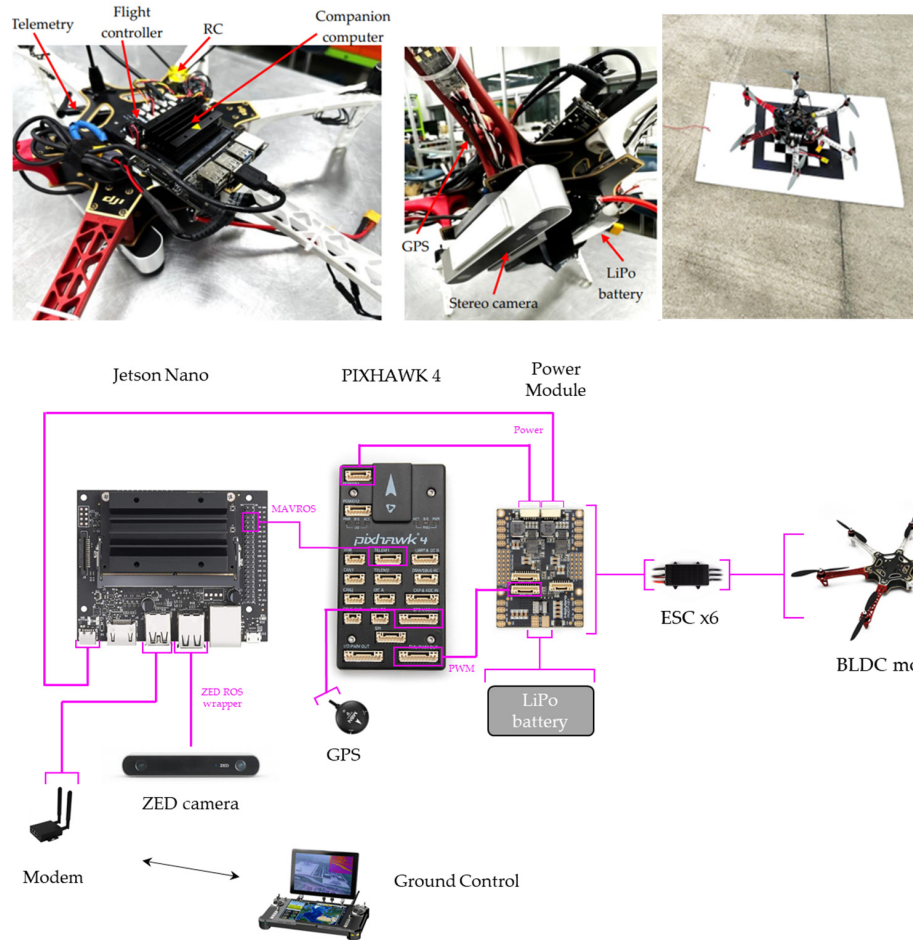

Figure 15. Hardware setup for experiment containing JETSON NANO companion computer and ZED camera integrated on Pixhawk 4 flight controller-based DJI F550 Hexacopter configuration.

4.2. Tracking and Landing Simulation Result

4.2.1. Straight Line Profile Landing

In this particular evaluation, we are testing the proposed adaptive tracking and landing strategy under a scenario where the target is moving in a straight line with a constant speed, denoted as V_t. The experiment involves varying the speed of the target, specifically at values of $V_t = 0.5 \frac{m}{s}$, $V_t = 1.5 \frac{m}{s}$ and $V_t = 3 \frac{m}{s}$, as illustrated in Figures 16 and 17.
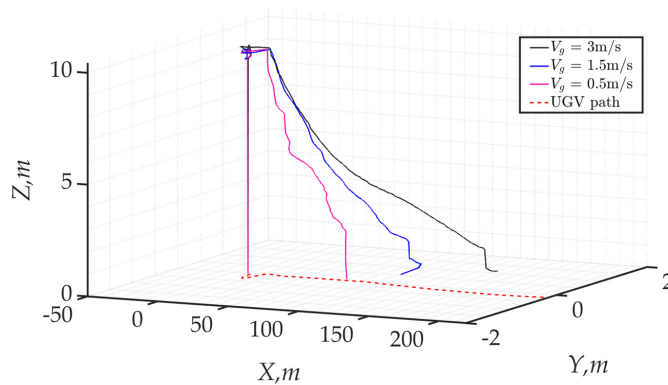
Figure 16. Straight line tracking and landing at different speeds of UGV $V_t = 0.5 \frac{m}{s}$, $V_t = 1.5 \frac{m}{s}$ and $V_t = 3 \frac{m}{s}$.
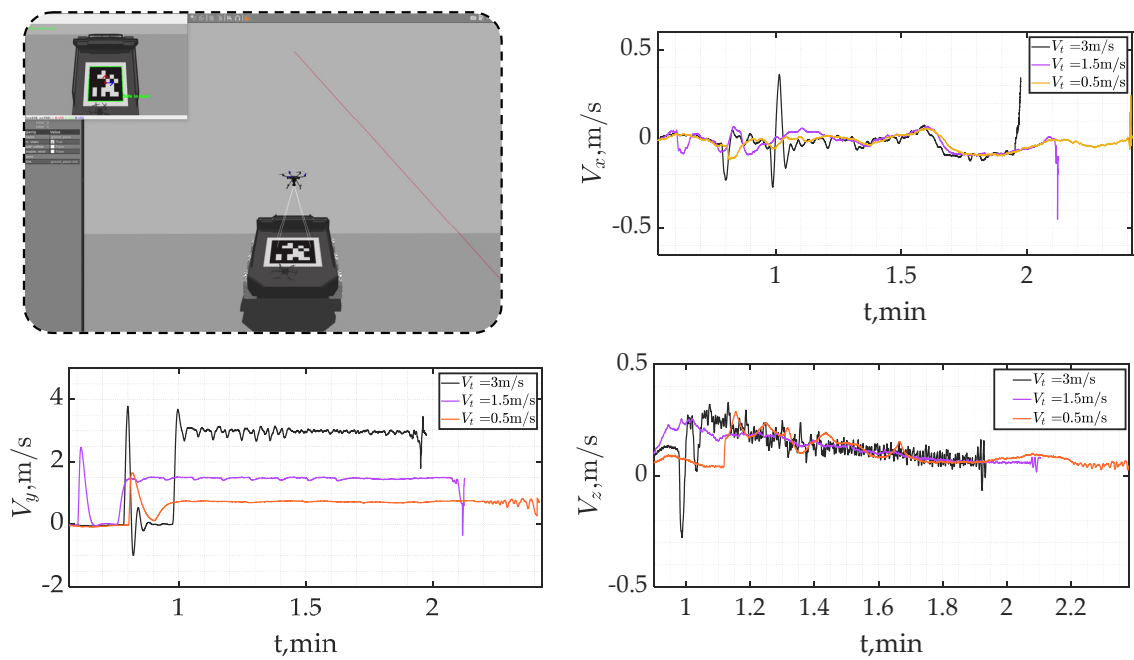
Figure 17. Velocity result for straight line tracking and landing at different speeds of UGV $V_t = 0.5 \frac{m}{s}$, $V_t = 1.5 \frac{m}{s}$ and $V_t = 3 \frac{m}{s}$.

The key observation from the results is that the UAV adeptly follows the moving target, ensuring precise tracking. This, in turn, leads to a seamless landing process where the tracking error ($R_{error} \cong 0$) is approximately zero, indicating a highly accurate tracking performance. Importantly, this shows that the proposed adaptive landing strategy imposes no penalties on the descent speed.

4.2.2. Addressing Tracking Errors, R_{error}, with Adaptive Landing

In this scenario, the assessment focused on evaluating the effectiveness of the proposed adaptive landing, which introduces penalties on the descent speed when a tracking error R_{error} occurs during landing. As illustrated in Figure 18a, an external force was applied in the middle of the landing, causing disturbance and displacement of the UAV from precise tracking. The resulting path of the UAV, depicted in Figure 18b, reveals the perturbation induced in the tracking.
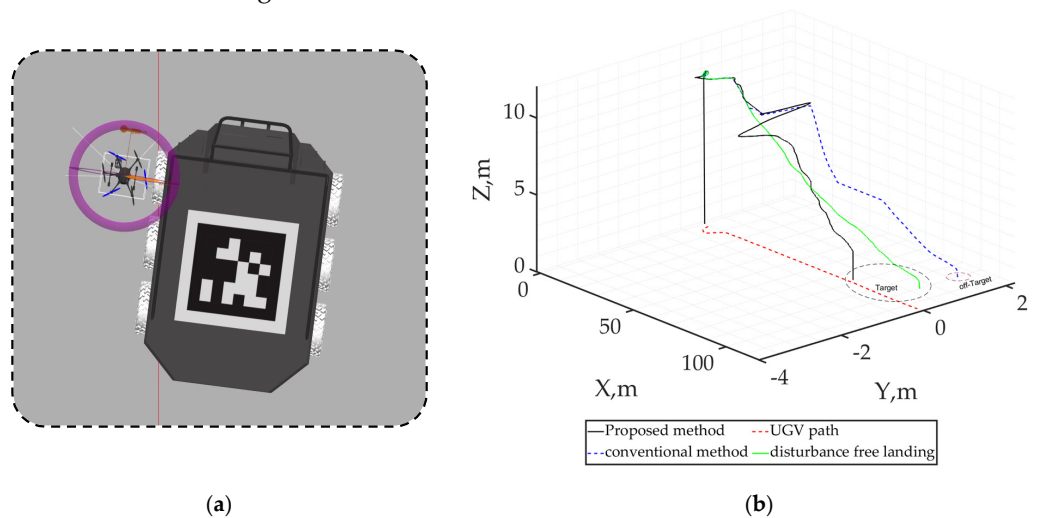

(**a**)　　　　　　　　　　　　　　　　　　　　　　(**b**)

Figure 18. Effect of perturbation tracking on landing process. (**a**) Disturbance injection in simulation. (**b**) Resulting position estimate.

Upon detecting a deviation from precise tracking, the proposed adaptive landing method responds by penalizing the descending speed. This deliberate slowdown in the landing progress allows time for the tracking precision to be restored, whereas the convectional method results in off-target landing. As illustrated in Figure 19, the highlighted area indicates that the altitude remains approximately at 8 m until the disturbance is rejected, and precise tracking is re-established. Consequently, this enhancement contributes to the precision of landing on moving objects.
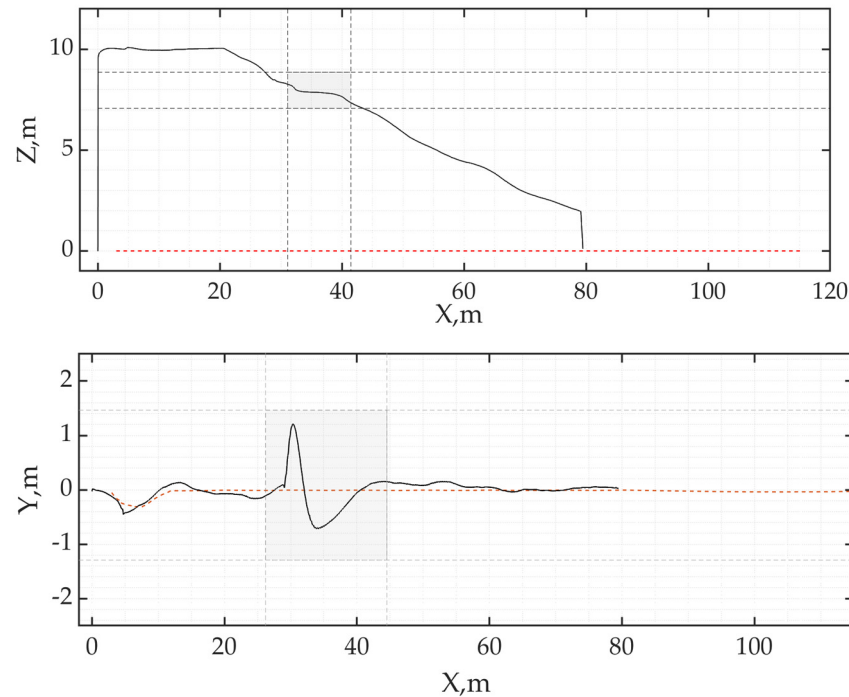
Figure 19. Effect of offsetting from tracking on landing process. The proposed adaptive landing strategy slows down the descending speed when the UAV is offset from tracking UGV precisely.

4.2.3. Adaptive Tracking in Circular and Rectangular Profile

In this section, we assessed the tracking and landing capabilities of the proposed method when the UAV faces a complete rotor failure, leading to a 30% loss in achievable torque, as per Equation (22). The first test examined how varying lookahead distances help smoothen the UAV's tracking response during sharp turns by a UGV, compensating for the UAV's reduced control authority. The second test evaluated the system's performance in tracking and landing on a moving UGV traveling at a constant speed of 1.5 m/s along a circular path.

As shown in Figure 20, the UGV followed a rectangular course with four sharp turns. Initially, the UAV was instructed to take off, identify the tag, and estimate the UGV's position. A reference line was then drawn connecting the previous and current estimated positions of the UGV. The tests analyzed the UAV's tracking effectiveness and the effects of various lookahead distances on its turning smoothness and agility when navigating sharp turns of the UGV. Three different lookahead distances were evaluated, repeating the simulation for each setting. The results indicated that a longer lookahead distance ($R_{AMS3} = 6$ m) resulted in smoother but less accurate turns, which was deemed acceptable. Conversely, a shorter lookahead distance ($R_{AMS2} = 2$ m) led to more precise but aggressive turns, and distances below 1.78 m caused the UAV to fail in tracking the UGV during sharp turns. These findings highlight that adjusting the minimum lookahead distance is crucial for managing the UAV's limited control capabilities, ensuring that operations stay within feasible limits. The ideal lookahead distance should be determined based on the UAV's dynamics and the severity of control degradation due to the rotor failure.
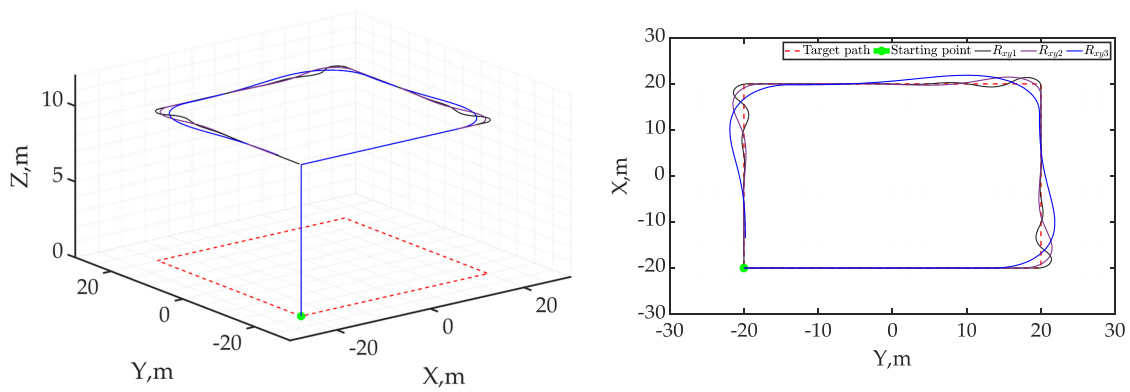
Figure 20. Tracking performance of proposed algorithm for different values of R_{AMS}.

In the second test, the UAV was initially commanded to take off and search for the tag, while the UGV followed a circular profile. Consequently, the UAV tracked the UGV along the circular path, as illustrated in Figure 21. After a period of tracking, the UAV transitioned to landing mode, descending while maintaining tracking within the circular profile. Ultimately, the UGV transported the UAV back to its initial position. The results affirm the robustness of the tracking and landing strategy across various profiles.
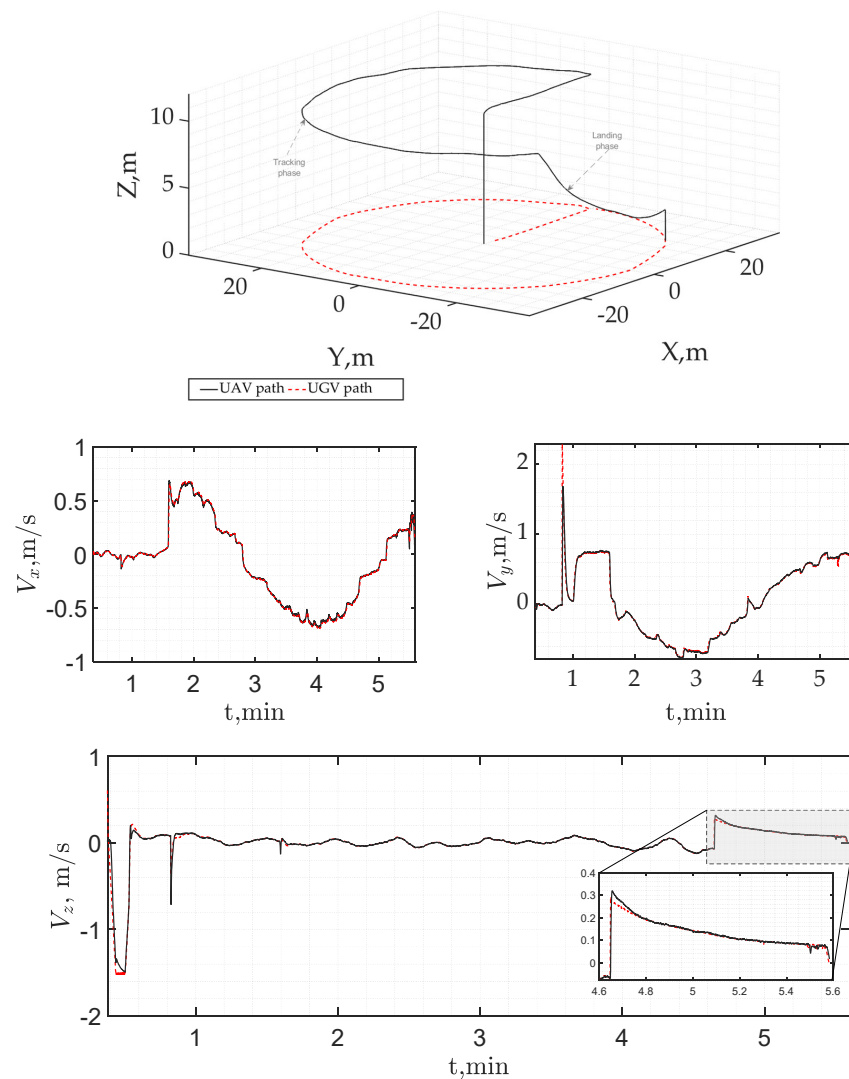

Figure 21. Tracking and landing in circular profile trajectory of UGV.

4.2.4. Emergency Landing Scenario in the Event of Actuator Failure: "Bring Back Home" Mission

This section illustrates a "bring back home" mission scenario involving a UAV using a UGV as a landing platform to guide it back home. This scenario replicates real-life situations where a UAV encounters actuator failure during a mission, lacks sufficient control authority to return home, and cannot land at the event location due to the absence of a designated landing site. Additionally, for the purpose of testing the proposed methodology while maintaining consistency and minimizing complexity, we assume that the battery levels of both vehicles are fully charged and sufficient to carry out the operation.

As depicted in Figure 22, a test plan with multiple phases—mission flight (A), midflight fault injection (B), recovery (C), tracking (D), and emergency landing with degraded control (E)—was devised and implemented using our proposed framework to address such challenges. Initially, the UAV was assigned a survey mission at 10 m and during its mission flight, a fault was injected to disable one of its actuators. This triggered the recovery phase, allowing the UAV to regain control. Once control was reestablished, the UAV initiated a search for a landing option by activating the camera and detection module. The event location was communicated to the UGV, which then planned a path passing through the UAV's current approximate location.
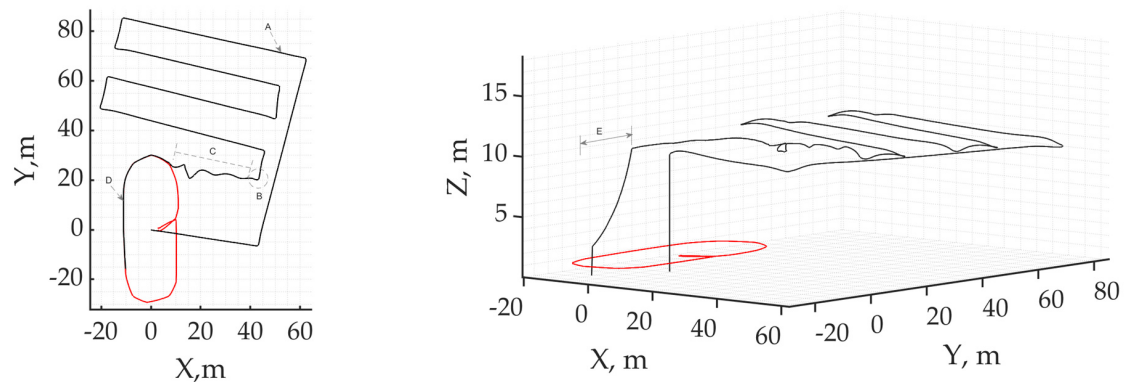

Figure 22. Result of emergency landing scenario due to actuator failure in the middle of mission flight.

As the UGV approached the event location, it became visible to the UAV's vision system, prompting the initiation of the UAV tracking module to receive the UGV's estimated pose. The tracking algorithm guided the UAV to follow the UGV, and upon precise tracking, the UAV landed on the UGV, as illustrated in Figure 23, showcasing the sequential landing process with precise tracking. Subsequently, the UGV carried the UAV back home. A simulation video of the entire scenario is available in Supplementary Material S1.

Figure 23. Emergency landing sequence from simulation.

The fault injection and recovery process are illustrated in Figure 24, depicting the normalized control commands (u) for each rotor. During the mission phase, all actuators operated normally, and control signals were allocated accordingly. However, at the third minute, a fault was injected into actuator -2, triggering the UAV's entry into the recovery

phase. In this phase, the contribution of the failed actuator was mitigated by reallocating commands to other healthy actuators, stabilizing the Hexacopter configuration to hover and regaining control shortly, as demonstrated in the recovery phase section.
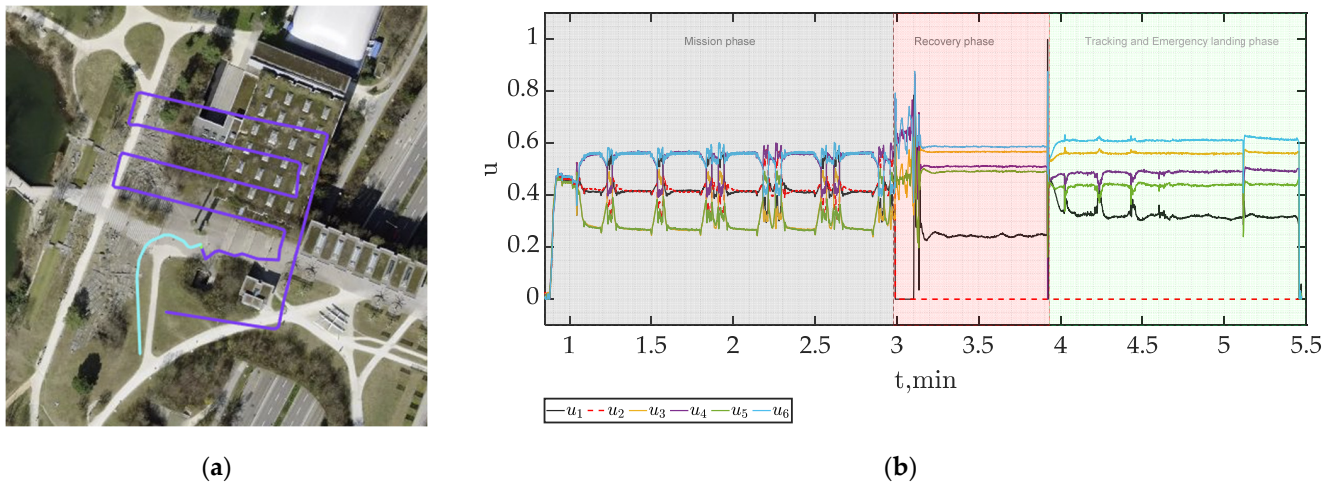

Figure 24. Emergency landing scenario result. (**a**) Trajectory formed by the UAV flight. (**b**) Actuators control command in each segment flight scenario. In the recovery phase, the controller reallocates the control command to other healthy ones and stabilizes the system.

Simultaneously, the adaptive pure pursuit algorithm adjusted its lookahead distance to prevent overshooting during tracking. Consequently, the results indicate the effectiveness of implementing our framework for tracking and executing emergency landings in situations where controllability is compromised due to actuator failure.

4.2.5. Experimental Results

This section outlines the efforts undertaken to validate the presented simulation results. While the experimental work is still in progress, initial tests have exhibited promising outcomes regarding the practical application of the proposed strategy. Due to constraints in resources and the multidisciplinary nature of the proposed system, the scope of the experimental tests was confined to real-time vision-based ground moving object pose estimation. This involved both April tags and standard H-type helipads, utilizing the YOLOv5 model for detection and tracking. Specifically, the focus was on tracking a prepared tag intended to simulate a UGV.

A custom April tag, measuring 0.4×0.4 m and equipped with a rope at one edge, was crafted to mimic UGV behavior. In the initial real flight experiment, an integrated UAV hardware setup was deployed to fly at an altitude of 10m and hover. The objective of this experiment was to estimate the pose of the tag relative to the UAV coordinate system. While the UAV hovered, the tag was manipulated using the prepared rope, as illustrated in Figure 25. The UGV module accurately reported the estimated pose.

Following successful verification, the UAV transitioned to offboard mode, initiating the tracking module, which began publishing the position set point at a rate of 2 Hz. Throughout this test, the UAV endeavored to track the moving tag on the ground, yielding affirmative results. However, challenges arose due to the simulated tag's inconsistent and intermittent movement, induced by human manipulation on uneven surfaces. Consequently, the testing process encountered difficulties.

Figure 25. Experimental flight test.

As of the composition of this report, ongoing efforts persist in refining the testing process. Furthermore, a real UGV is under development, capable of controlled movement at desired speeds and directions, to facilitate more rigorous validation in subsequent experiments.

The proposed system underwent testing using a crafted board as a moving target for tracking and landing purposes. As illustrated in Figure 26, the tag moved at a roughly constant speed of 1.5 m/s. The position estimation, carried out through an installed camera and an April tag detection model, yielded unsmooth and intermittent results. However, the integration of a Kalman filter algorithm, which treated the tag's movement as a linear system model, enabled the prediction of the target's path. This predicted path was then used to create a reference trajectory for the proposed adaptive pure pursuit algorithm. Consequently, the Unmanned Aerial Vehicle (UAV) was able to anticipate the target's movement, allowing for smoother tracking and landing, as depicted in the figure.
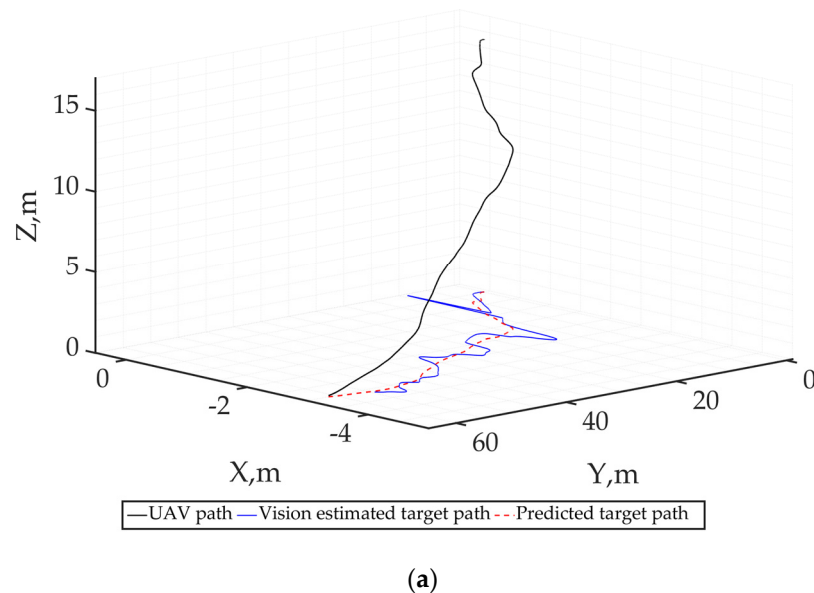

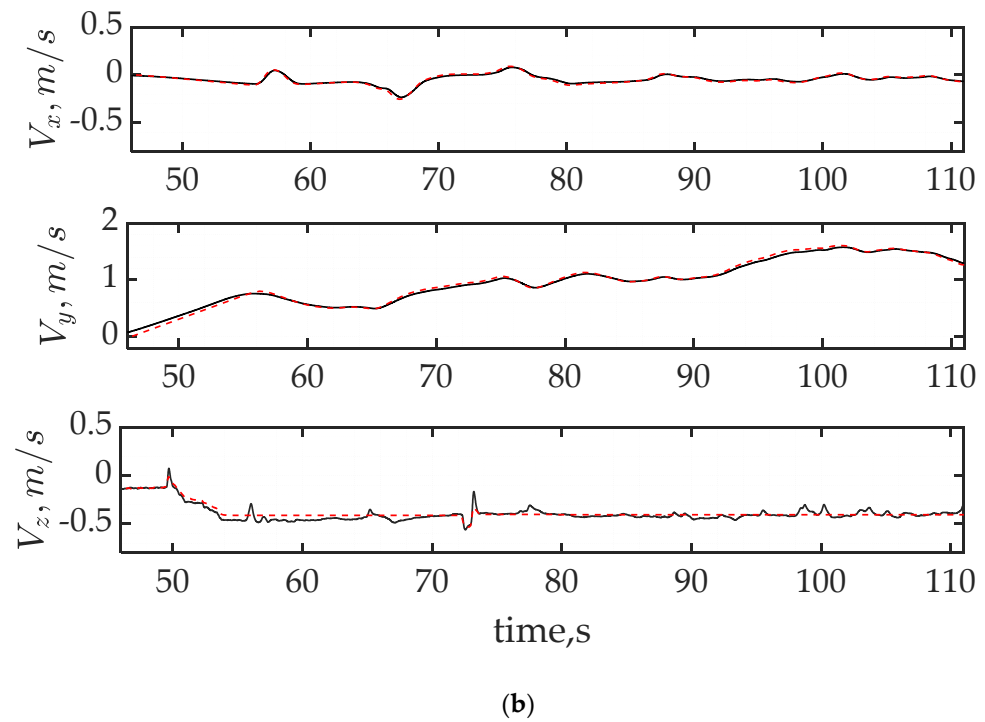(**a**)

Figure 26. *Cont*.

(**b**)

Figure 26. Experimental test result summary for UAV tracking and landing on moving object with degraded control.

Likewise, another landing pad under consideration for utilization and evaluation is the H-type helipad. As expounded upon in Section 3, the deep learning model that was trained has been seamlessly integrated into our detection package. It has undergone real-time testing to identify the helipads and estimate their poses. As illustrated in Figure 27, the model exhibits a robust ability to detect the helipad with a high class probability, underscoring its viability for effective pose estimation.
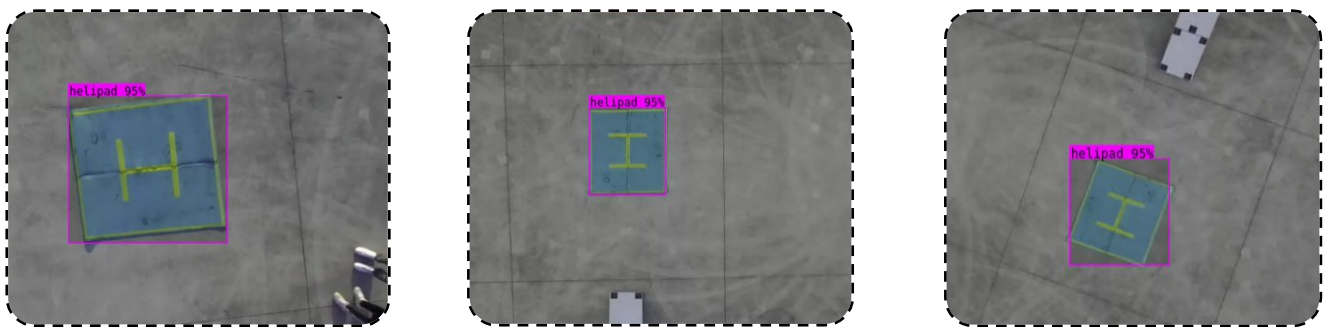

Figure 27. Standard H-type helipad real-time detection and pose estimation using YOLOv5 trained model.

Furthermore, a controllability test was conducted on the Hexacopter UAV to assess its response to control degradation caused by actuator failure. The experimental videos capturing these tests are provided in Supplementary Material S1.

Despite the ongoing nature of the experiment and the ongoing resolution of described limitations, this real flight experiment demonstrates the feasibility and adaptability of the proposed system.

5. Conclusions

In conclusion, this study advances the development of adaptive tracking and landing algorithms for UAVs targeting moving objects, specifically focusing on emergency scenarios involving propulsion system failures that impair maneuverability and control. The research begins with an in-depth analysis of the dynamics and control strategies of UAVs and UGVs. We critically evaluated the UAV's performance during failures, selecting a fault-tolerant Hexacopter-type Multirotor UAV as our experimental platform. This work incorporates advanced pose estimation using UAV-mounted cameras and tag detection algorithms, evaluating two state estimation methods for the UGV, using April tags and the YOLOv5 model for H-type landing pads.

A modified adaptive pure pursuit algorithm was developed to compensate for diminished control authority, allowing accurate tracking of the moving target's position and heading, with integration of a Kalman filter to maintain smooth tracking, even during occlusions. Additionally, the study introduces an adaptive landing strategy that adjusts dynamically to perturbations during landing by implementing vertical speed control dependent on lateral offset. The proposed algorithms are integrated into an emergency landing scenario named "Bring Back Home", which includes necessary modules like actuator fault detection, isolation, and tolerance. These modules detect and assess control authority degradation to trigger emergency landing protocols and adjust tracking parameters accordingly. The effectiveness of these techniques was tested both individually and within an integrated simulation environment. The first simulation tested the UAV's ability to track and land on a moving target at constant speeds of 0.5 m/s, 1.5 m/s, and 3 m/s with 30% reduced control authority due to a rotor loss. The second test evaluated the proposed technique's performance in landing on a moving target amid external perturbations. Further tests assessed the tracking capability against a target executing sharp turns, comparing the adaptive landing algorithm to conventional methods.

Overall, these tests validated the proposed methods' effectiveness in maintaining safe and controlled tracking and landing on moving targets under degraded control conditions. Additionally, hardware tests confirmed the system's performance in real-world tracking and landing scenarios on moving targets. This comprehensive validation under-scores the robustness of the proposed solutions in enhancing UAV operational reliability in dynamic and challenging environments. Simulation videos illustrating the conducted tests and the ongoing experimental flight test process are provided in Supplementary Material S1. As future work, efforts are underway to establish a complete experimental setup for testing UAV–UGV collaboration in emergency landing scenarios, including the implementation of a communication system between the UAV and UGV and other complex scenarios that include the battery level.

Author Contributions: Conceptualization, Y.D.; investigation, Y.D. and H.-Y.S.; methodology, Y.D.; software, Y.D. and H.-Y.S.; supervision, T.-W.K. and B.-S.K.; validation, Y.D., H.W and A.W.; visualization, Y.D. and H.-Y.S.; writing—original draft, Y.D.; writing—review and editing, Y.D., H.-Y.S. and H.W. All authors have read and agreed to the published version of the manuscript.

Data Availability Statement: The data presented in this study are available on request from the authors.

for high reliability stampede accidents using CCTV and Drone imaging, project number: 20024403, contribution rate: 50%).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Mohsan, S.A.H.; Othman, N.Q.H.; Li, Y.; Alsharif, M.H.; Khan, M.A. Unmanned Aerial Vehicles (UAVs): Practical Aspects, Applications, Open Challenges, Security Issues, and Future Trends. *Intell. Serv. Robot.* **2023**, *16*, 109–137. [CrossRef]
2. Ding, Y.; Xin, B.; Chen, J. A Review of Recent Advances in Coordination Between Unmanned Aerial and Ground Vehicles. *Unmanned Syst.* **2021**, *9*, 97–117. [CrossRef]
3. Sanchez-lopez, J.; Jesus, P.; Srikanth, S.; Pascual, C. An approach toward visual autonomous shipboard landing of a VTOL UAV. *J. Intell. Robot. Syst.* **2014**, *74*, 113–127. [CrossRef]
4. Palafox, P.R.; Garzón, M.; Valente, J.; Roldán, J.J.; Barrientos, A. Robust Visual-Aided Autonomous Takeoff, Tracking, and Landing of a Small UAV on a Moving Landing Platform for Life-Long Operation. *Appl. Sci.* **2019**, *9*, 2661. [CrossRef]
5. He, Z.; Xu, J.-X. Moving Target Tracking by Uavs in an Urban Area. In Proceedings of the 2013 10th IEEE International Conference on Control and Automation (ICCA), Hangzhou, China, 28 April 2013. [CrossRef]
6. Bie, T.; Fan, K.; Tang, Y. UAV Recognition and Tracking Method Based on Yolov5. In Proceedings of the 2022 IEEE 17th Conference on Industrial Electronics and Applications (ICIEA), Chengdu, China, 28 April 2022. [CrossRef]
7. Zhigui, Y.; ChuanJun, L. Review on Vision-based Pose Estimation of UAV Based on Landmark. In Proceedings of the 2017 2nd International Conference on Frontiers of Sensors Technologies (ICFST), Shenzhen, China, 28 April 2017. [CrossRef]
8. Gautam, A.; Singh, M.; Sujit, P.B.; Saripalli, S. Autonomous Quadcopter Landing on a Moving Target. *Sensors* **2022**, *22*, 1116. [CrossRef]
9. Ghommam, J.; Saad, M. Autonomous Landing of a Quadrotor on a Moving Platform. *IEEE Trans. Aerosp. Electron. Syst.* **2017**, *53*, 1504–1519. [CrossRef]
10. Morales, J.; Castelo, I.; Serra, R.; Lima, P.U.; Basiri, M. Vision-Based Autonomous Following of a Moving Platform and Landing for an Unmanned Aerial Vehicle. *Sensors* **2023**, *23*, 829. [CrossRef] [PubMed]
11. Fang, X.; Wan, N.; Jafarnejadsani, H.; Sun, D.; Holzapfel, F.; Hovakimyan, N. Emergency Landing Trajectory Optimization for Fixed-wing UAV under Engine Failure. In Proceedings of the AIAA Scitech 2019 Forum, San Diego, CA, USA, 7–11 January 2019. [CrossRef]
12. Lippiello, V.; Ruggiero, F.; Serra, D. Emergency Landing for a Quadrotor in Case of a Propeller Failure: A PID Based Approach. In Proceedings of the 2014 IEEE International Symposium on Safety, Security, and Rescue Robotics, Toyako, Japan, 28 April 2014.
13. Debele, Y.; Shi, H.-Y.; Wondosen, A.; Kim, J.-H.; Kang, B.-S. Multirotor Unmanned Aerial Vehicle Configuration Optimization Approach for Development of Actuator Fault-tolerant Structure. *Appl. Sci.* **2022**, *12*, 6781. [CrossRef]
14. Saied, M.; Shraim, H.; Lussier, B.; Fantoni, I.; Francis, C. Local Controllability and Attitude Stabilization of Multirotor Uavs: Validation on a Coaxial Octorotor. *Robot. Auton. Syst.* **2017**, *91*, 128–138. [CrossRef]
15. Du, G.; Quan, Q.; Yang, B.; Cai, K.-Y. Controllability Analysis for a Class of Multirotors Subject to Rotor Failure/wear. *arXiv* **2014**, arXiv:1403.5986. [CrossRef]
16. Du, G.-X.; Quan, Q.; Yang, B.; Cai, K.-Y. Controllability Analysis for Multirotor Helicopter Rotor Degradation and Failure. *J. Guid. Control Dyn.* **2015**, *38*, 978–985. [CrossRef]
17. Vey, D.; Lunze, J. Experimental Evaluation of an Active Fault-tolerant Control Scheme for Multirotor Uavs. In Proceedings of the 2016 3rd Conference on Control and Fault-Tolerant Systems (SysTol), Barcelona, Spain, 28 April 2016. [CrossRef]
18. Xia, K.; Shin, M.; Chung, W.; Kim, M.; Lee, S.; Son, H. Landing a Quadrotor UAV on a Moving Platform with Sway Motion Using Robust Control. *Control Eng. Pract.* **2022**, *128*, 105288. [CrossRef]
19. Bogdan, S.; Orsag, M.; Oh, P. *Multi-Rotor Systems, Kinematics, Dynamics, and Control of*; Springer: Berlin/Heidelberg, Germany, 2020. [CrossRef]
20. Zhang, J.; Söpper, M.; Holzapfel, F. Attainable Moment Set Optimization to Support Configuration Design: A Required Moment Set Based Approach. *Appl. Sci.* **2021**, *11*, 3685. [CrossRef]
21. Zhao, Y.; BeMent, S.L. Kinematics, Dynamics and Control of Wheeled Mobile Robots. In Proceedings of the 1992 IEEE International Conference on Robotics and Automation, Nice, France, 28 April 1992. [CrossRef]
22. Suzuki, S. Autonomous Navigation, Guidance and Control of Small 4-wheel Electric Vehicle. *J. Asian Electr. Veh.* **2012**, *10*, 1575–1582. [CrossRef]
23. Lian, J.; Yu, W.; Xiao, K.; Liu, W. Cubic Spline Interpolation-based Robot Path Planning Using a Chaotic Adaptive Particle Swarm Optimization Algorithm. *Math. Probl. Eng.* **2020**, *2020*, 20. [CrossRef]
24. Park, J.; Jung, Y.; Kim, J. Multiclass Classification Fault Diagnosis of Multirotor Uavs Utilizing a Deep Neural Net-work. *Int. J. Control Autom. Syst.* **2022**, *20*, 1316–1326. [CrossRef]
25. Abbas, S.M.; Aslam, S.; Berns, K.; Muhammad, A. Analysis and Improvements in AprilTag Based State Estimation. *Sensors* **2019**, *19*, 5480. [CrossRef] [PubMed]

26. Zhu, J.; Jia, Y.; Shen, W.; Qian, X. A Pose Estimation Method in Dynamic Scene with Yolov5, Mask R-CNN and ORB-SLAM2. In Proceedings of the 2022 7th International Conference on Signal and Image Processing (ICSIP), Suzhou, China, 28 April 2022. [CrossRef]

27. Subramanian, J.; Asirvadam, V.; Zulkifli, S.; Singh, N.; Shanthi, N.; Lagisetty, R.; Kadir, K. Integrating Computer Vision and Photogrammetry for Autonomous Aerial Vehicle Landing in Static Environment. *IEEE Access* **2024**, *12*, 4532–4543. [CrossRef]

28. Goshtasby, A.; Gruver, W.A. Design of a Single-Lens Stereo Camera System. *Pattern Recognit.* **1993**, *26*, 923–937. [CrossRef]

29. Ma, M.; Shen, S.; Huang, Y. Enhancing UAV Visual Landing Recognition with Yolo's Object Detection by Onboard Edge Computing. *Sensors* **2023**, *23*, 8999. [CrossRef]

30. Liu, Z.; Gao, X.; Wan, Y.; Wang, J.; Lyu, H. An Improved Yolov5 Method for Small Object Detection in UAV Capture Scenes. *IEEE Access* **2023**, *11*, 14365–14374. [CrossRef]

31. Nepal, U.; Eslamiat, H. Comparing YOLOv3, YOLOv4 and YOLOv5 for Autonomous Landing Spot Detection in Faulty UAVs. *Sensors* **2022**, *22*, 464. [CrossRef]

32. Jung, J.; Yoon, I.; Lee, S.; Paik, J. Object Detection and Tracking-based Camera Calibration for Normalized Human Height Estimation. *J. Sens.* **2016**, *2016*, 1–9. [CrossRef]

33. Ahn, J.; Shin, S.; Kim, M.; Park, J. Accurate Path Tracking by Adjusting Look-Ahead Point in Pure Pursuit Method. *Int. J. Automot. Technol.* **2021**, *22*, 119–129. [CrossRef]

34. Zhang, M.; Tian, F.; He, Y.; Li, D. Adaptive Path Tracking for Unmanned Ground Vehicle. In Proceedings of the 2017 IEEE International Conference on Unmanned Systems (ICUS), Beijing, China, 28 April 2017. [CrossRef]

35. Giesbrecht, J.; Mackay, D.; Collier, J.; Verret, S. Path Tracking for Unmanned Ground Vehicle Navigation: Implementation and Adaptation of the Pure Pursuit Algorithm. *DRDC Suffield TM* **2005**, *224*, 2005.

36. Chuang, H.-M.; He, D.; Namiki, A. Autonomous Target Tracking of UAV Using High-Speed Visual Feedback. *Appl. Sci.* **2019**, *9*, 4552. [CrossRef]

37. Teuliere, C.; Eck, L.; Marchand, E. Chasing a Moving Target from a Flying UAV. In Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, San Francisco, CA, USA, 25 September 2011. [CrossRef]