

Flight Test Validation of a Vision-Guided Multirotor Terminal Interception System

Ye-Ran Kim[1], Jinheui Hong[1], YongJun Jang[1], Jung-Ho Moon[1]

[1]Cheongju University

Abstract

Image-based terminal guidance technology is one of the core technologies for performing precise approach and interception of targets. However, there is still limited research that has performed image-based terminal guidance by integrating object detection results with flight control and verified the system's operational feasibility through flight tests. In this study, we implemented an image-based terminal guidance system that performs terminal guidance using object detection results and target-centered errors relative to the image center. Additionally, we integrated object detection, guidance command generation, and flight control functions by applying a communication architecture based on ROS 2 and MAVLink. To train the object detection model, a dataset of approximately 18,000 images was constructed using UAV video data collected from real-world operational environments and UAV video data from open platforms such as YouTube. Flight tests were conducted using a fixed-wing target with a wingspan of 2.3 m, and the operational feasibility of target detection, tracking, and terminal guidance functions was verified at distances of over 100 m.

The test results confirmed the ability to detect and track targets during flight, as well as the capability to perform terminal approach and collision with a stationary aerial target. This confirmed the applicability of a closed-loop terminal guidance system—which integrates object detection, guidance, and flight control—in real-world flight environments.

Keywords: IBVS (Image-Based Visual Servoing), Terminal Interception, Vision-Based Terminal Guidance, Flight Test, Closed-Loop

1. Introduction

Recently, unmanned aerial vehicles (UAVs) have been used in a wide range of fields, including reconnaissance, surveillance, disaster response, and autonomous missions. In particular, multicopter-based UAVs are increasingly being deployed in complex operational environments due to their compact size, low cost, and simple control systems [1]. In addition, the importance of image-based terminal guidance technology for autonomously detecting and approaching targets in GNSS (Global Navigation Satellite System) limited environments or dynamic target environments continues to grow [2,3].

Recently, with advances in deep learning-based object detection technology, research on target recognition in UAV environments has been actively conducted [4]. Zheng et al. proposed a deep learning-based detection system for detecting small UAVs in an air-to-air environment and analyzed its detection capabilities in real-world flight conditions [5]. Furthermore, Duan et al. proposed the M-YOLOv8s small-target detection algorithm, based on YOLOv8s, to address the challenge of detecting small targets against complex backgrounds in UAV aerial imagery [6]. However, existing research on image-based autonomous guidance systems has primarily focused on simulation-based validation or algorithm-level performance evaluation; there are relatively few cases in which the real-time performance and operational feasibility of systems integrating object detection, guidance, and control have been empirically verified through actual flight tests [7,8].

In image-based terminal guidance systems, target recognition, calculation of relative position errors, generation of guidance commands, and application of vehicle control inputs are linked in a continuous closed-loop structure. However, since multicopters have nonlinear dynamic characteristics in which attitude control and position control are strongly coupled, image processing latency, data transmission delays, and control cycle mismatches can cause aircraft vibrations and overshoot, which may lead to degraded terminal guidance performance [9]. Furthermore, since factors such as changes in lighting,

variations in target size, and increased background complexity often occur simultaneously in real-world environments, the development of a stable image-based guidance system remains a critical research challenge [10].

In this paper, we propose a terminal guidance system capable of incorporating the results of image-based object detection into real-time guidance control within a multicopter platform environment. Furthermore, through flight tests, we aim to verify terminal approach and collision with static aerial targets, thereby confirming the operational feasibility of a closed-loop system that integrates object detectionn, guidance, and flight control in a real-world environment.

This paper is organized as follows. Section 2 describes the overall architecture of the proposed image-based terminal guidance system, as well as the object detection and guidance control techniques. Section 3 details the flight test platform, target configuration, and test scenarios. Section 4 analyzes the system's terminal guidance performance based on the flight test results. Finally, Section 5 presents the conclusions and future research directions.

2. System Architecture/Methodology

2.1 Overall System Configuration

In this study, we implemented an integrated system for performing vision-based terminal guidance on a multicopter platform. The proposed system consists of an image acquisition unit, an object detection unit, a guidance command generation unit, and a flight control unit; the overall structure is shown in Figure 1.

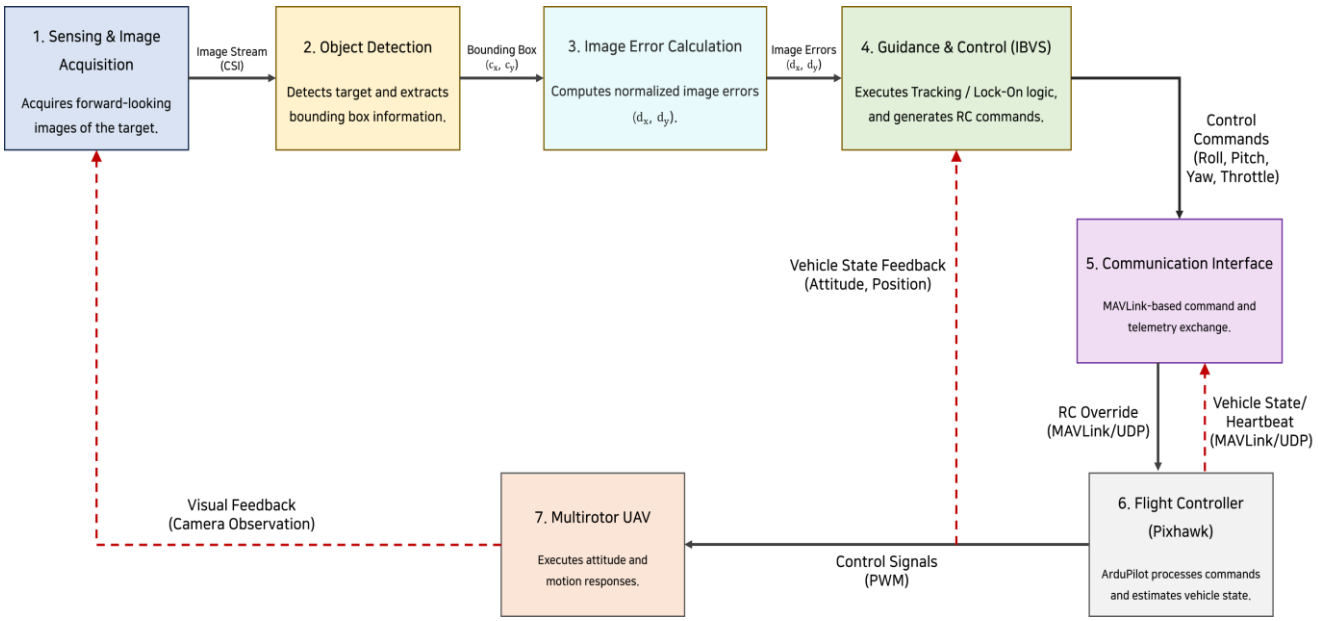


Figure 1 - Overall System Architecture.

Video footage captured by the onboard camera is transmitted to the onboard computing device, where object detection and navigation calculations are performed in real time. The software architecture is based on ROS 2, with image processing, object detection, navigation control, and flight state management implemented as separate nodes.

The image processing node receives camera images and forwards them to the object detection node. The object detection node performs YOLOv8-based object detection to generate bounding box information for targets, and the generated target information is transmitted to the guidance control node via ROS 2 topic communication. The center coordinates of the generated bounding box are then transmitted to the image error calculation module, where the horizontal and vertical errors between the image center and the target center are calculated. The calculated error information is transmitted to the guidance and control node. Based on this, Roll, Pitch, Yaw, and Throttle commands are generated. The generated control commands are transmitted to the flight controller via an MAVLink-based communication interface, and the flight controller uses the received commands to control the aircraft's attitude and motion. This process is repeated continuously during flight, forming a real-time

closed-loop structure that seamlessly connects image acquisition to flight control. The system is configured to enable real-time target tracking and terminal guidance using only image information.

2.2 Vision-Based Target Detection

In video-based guidance systems, since object detection results are directly incorporated into the guidance control loop in real time, not only high detection accuracy but also inference latency and computational load are critical design factors.

Among the YOLOv8 model family, YOLOv8n is a lightweight model with a relatively small number of parameters and low computational cost [11], and it possesses characteristics advantageous for real-time inference in UAV environments with limited onboard computing resources. Therefore, in this study, the YOLOv8n model was applied to ensure the real-time performance of the guidance control loop in the onboard environment. To train the object detection model, a training dataset of approximately 18,000 images was constructed using UAV video data collected from actual operational environments and UAV video data obtained from open platforms such as YouTube. The dataset was designed to include a variety of flight distances, attitude changes, and background conditions, with the goal of ensuring reliable object detection performance across diverse environments. Bounding box-based labeling was performed on the collected videos, and the dataset was structured so that the target UAV appeared in the images in various sizes and orientations. Model training was conducted at an input resolution of 640×640, and the trained model was deployed on an onboard computing device to perform real-time inference. Object detection was performed on input video acquired during flight, and bounding boxes were generated for the detected targets.

Figure 2 shows the object detection results based on YOLOv8.



Figure 2 - YOLOv8-based target detection results.

2.3 Guidance and Control Strategy

The proposed system is configured as an IBVS system that calculates the error relative to the image center using the center coordinates of the bounding box generated from the object detection results, and generates steering commands based on this to control the vehicle. In the guidance control module, horizontal and vertical errors were calculated using the difference between the transmitted bounding box center coordinates (C_x, C_y) and the image center $(W/2, H/2)$. The errors were normalized and calculated as follows.

$$d_x = \frac{c_x - \frac{W}{2}}{\frac{W}{2}}, \qquad d_y = \frac{c_y - \frac{H}{2}}{\frac{H}{2}} \tag{1}$$

Here, W and H represent the horizontal and vertical resolutions of the image, respectively.

The calculated values d_x and d_y denote the relative position of the target relative to the center of the image; when$(d_x = 0)$ and $(d_y = 0)$, it means the target is located at the center of the image.

The proposed guidance and control architecture consists of two states. The Tracking phase and the Lock-On phase. In the initial phase, the system uses object detection results to determine whether a target has been detected; if a target is detected, the system transitions to the Tracking phase.

In the Tracking phase, Roll and Yaw control commands were generated using the horizontal error d_x, while the Pitch command was set to perform a constant forward maneuver. Additionally, the Throttle command was set to maintain a hovering state. The control inputs for the tracking stage are defined as follows.

$$u_{roll} = 1500 + K_r d_x \tag{2}$$

$$u_{yaw} = 1500 + K_y d_x \tag{3}$$

$$u_{pitch} = u_{forward} \tag{4}$$

$$u_{throttle} = u_{hover} \tag{5}$$

Here, k_r and k_y represent the roll and yaw control gains, respectively, while u_{roll}, u_{yaw}, u_{pitch}, and $u_{throttle}$ represent the roll, yaw, pitch, and throttle RC commands, respectively.

If the target continues to be detected and tracked, the system transitions to the Lock-On phase, which is the stage for performing terminal guidance. In this implementation, the system is configured to transition to the next phase if target tracking is maintained for approximately 3 seconds or longer during the Tracking phase. Upon entering the Lock-On phase, the roll control gain is increased to enable more aggressive maneuvers for terminal guidance, and an altitude correction function is added using the vertical error d_y. The control inputs for the Lock-On phase are defined as follows.

$$u_{roll} = 1500 + 1.5K_r d_x \tag{6}$$

$$u_{yaw} = 1500 + 0.8K_y d_x \tag{7}$$

$$u_{pitch} = u_{forward} \tag{8}$$

$$u_{throttle} = u_{hover} - K_t d_y \tag{9}$$

Here k_t represents the altitude correction gain for vertical errors.

The proposed state-based guidance structure is designed to transition to the lock-on phase only when the target detection results meet the specified conditions, thereby reducing the likelihood of terminal guidance being initiated due to temporary false alarms. Figure 3 illustrates the guidance and control architecture for image-based terminal guidance.

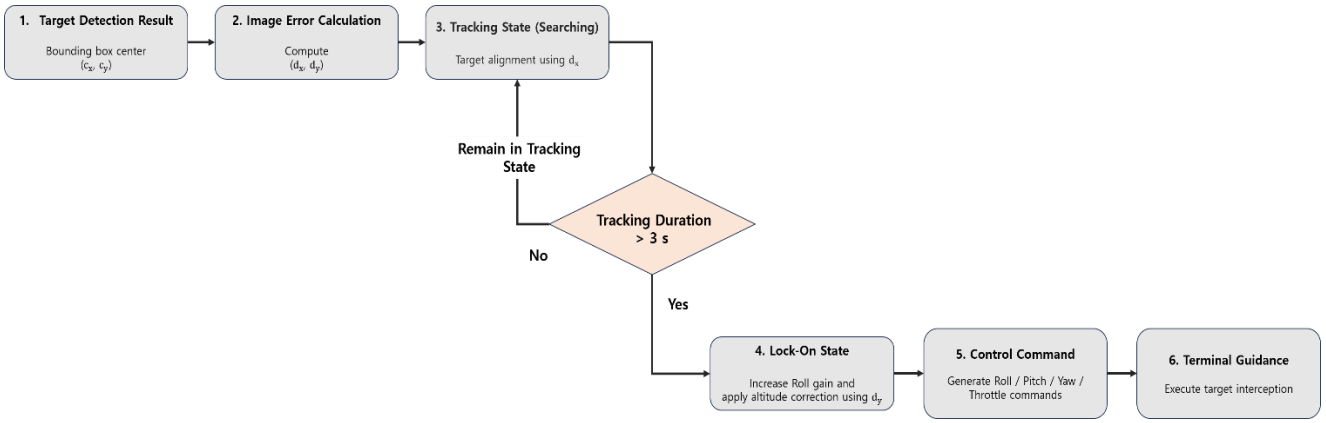


Figure 3 - Guidance and control structure for vision-based terminal interception.

3. Flight Test Setup

3.1 System Configuration

A multicopter-based flight test platform was developed to verify the operational feasibility of a video-based terminal guidance system in a real-world flight environment. The platform consists of a video sensor, an onboard computing device, and a flight control system. An IMX219 camera was used as the video sensor and mounted on the front of the aircraft to enable real-time observation of forward

aerial targets. Additionally, the onboard computing device utilized a combination of the NVIDIA Jetson Orin NX and the A603 Carrier Board. This onboard computing device offers sufficient computational performance to perform real-time object detection and guidance calculations while maintaining a size and weight suitable for deployment on small strike-class UAVs. Furthermore, the A603 Carrier Board was directly connected to the CSI (Camera Serial Interface)-based IMX219 camera, enabling real-time image data processing without the need for a separate video transmission device. This configuration ensured system lightweight design and integration, while also guaranteeing stable image processing performance even in low-power environments.

Figure 4 shows the hardware configuration of the multicopter platform used in the flight tests and the target UAV.



(a) Multirotor Platform and onboard hardware

(b) Target UAV used in Flight tests

Figure 4 - Hardware configuration of the flight test platform and target UAV.

A fixed-wing UAV with a wingspan of approximately 2.3 meters was used as a target to verify terminal guidance performance. To simulate an aerial target environment, the target was suspended in the air to create a static target environment, allowing for the verification of object detection, tracking, and terminal guidance capabilities while eliminating the influence of target maneuvering.

The software environment was built on the Ubuntu 22.04 operating system and JetPack 6.0, and the object detection and guidance control functions were implemented in the ROS 2 Humble environment. Additionally, a MAVLink-based communication interface was implemented to exchange command and status information between the onboard computing device and the flight controller. The flight control system utilized an ArduPilot-based flight controller. The flight controller performs attitude and motion control of the aircraft and is configured to execute terminal guidance using guidance commands generated by the onboard computing device. Table 1 shows the main hardware configuration used in the flight tests.

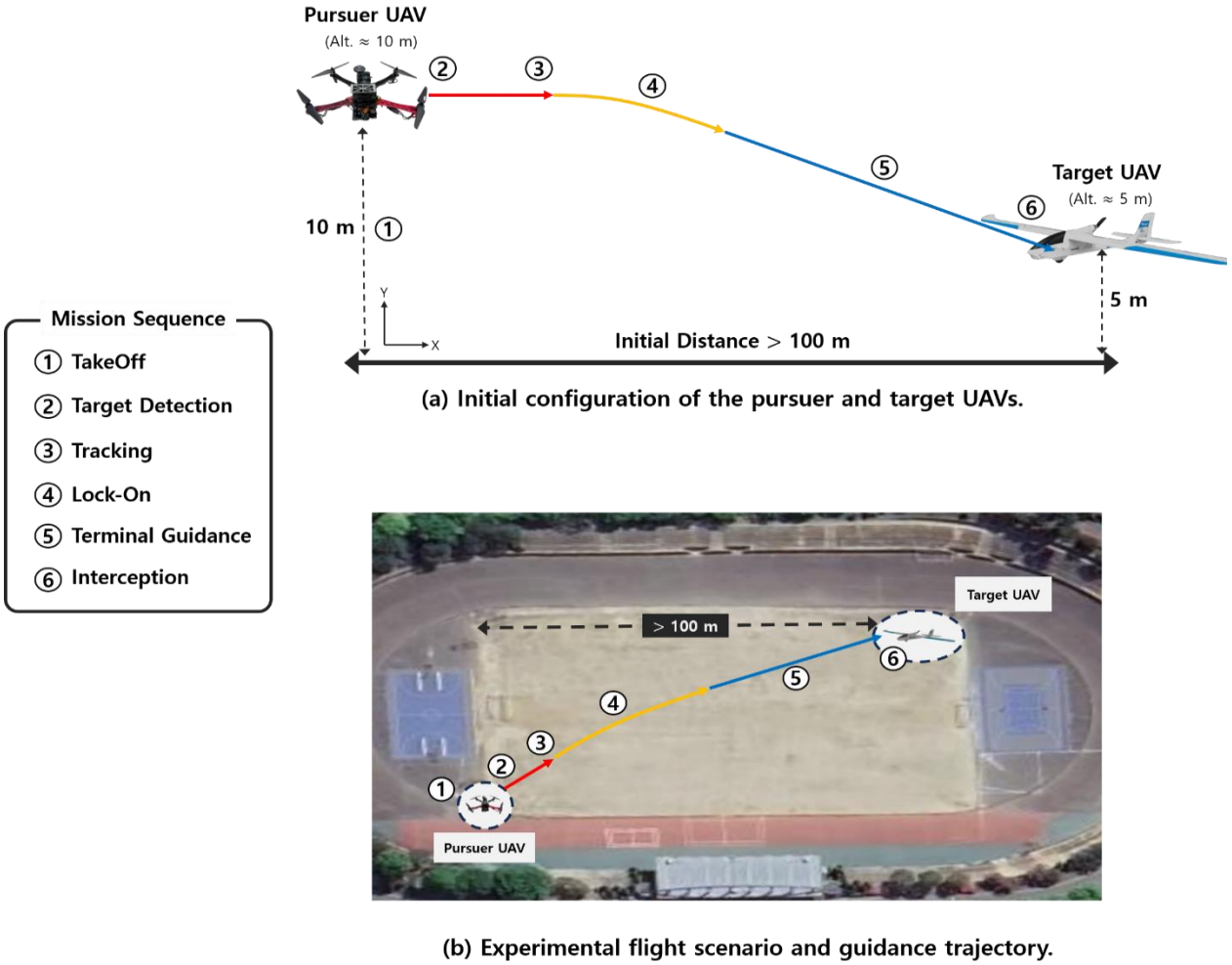
Table 1 - Flight Test System Configuration

Component	Model
Airframe	Tarot TBS-500 Discovery
Flight Controller	Holybro Pixhawk 6C Mini
Onboard Computer	Jetson Orin NX (16 GB)
Carrier Board	Seeed Studio A603
Camera	IMX219

3.2 Flight Test Scenario

In this study, flight tests were conducted to verify the operational feasibility of the proposed image-based terminal guidance system.

Figure 6 illustrates the flight test environment and the terminal guidance scenario.



(a) Initial configuration of the pursuer and target UAVs.

(b) Experimental flight scenario and guidance trajectory.

Figure 6 – Flight test environment and terminal guidance scenario.

The flight tests were conducted at an initial distance of approximately 100 meters or more. The target was positioned at a height of approximately 5 meters above ground level, and the pursuing aircraft was configured to take off and fly up to an altitude of approximately 10 meters. This setup simulated a long-range detection and approach scenario for an aerial target.

The flight test was designed so that, after the interceptor took off, the phases of target detection, tracking, lock-on, and terminal guidance were performed sequentially, and the system's normal operation was verified at each stage.

4. Flight Test Result

The flight test was conducted at an initial distance of approximately 100 m or more, and a fixed-wing target with a wingspan of 2.3 m was detected using forward camera imagery. The test results showed that a bounding box was generated for the target in the camera imagery, and it was confirmed that object detection continued continuously even during flight. During the tracking phase, roll and yaw commands were generated based on the position error relative to the image center, and it was confirmed that tracking was performed such that the image center converged toward the target center. Additionally, after target tracking was maintained for approximately 3 seconds or more, the system transitioned to the lock-on phase, during which increased roll control gain and throttle compensation based on vertical direction error were applied.

Figure 7 shows the results of target detection, tracking, and terminal guidance conducted during flight testing, confirming that each phase was performed sequentially.

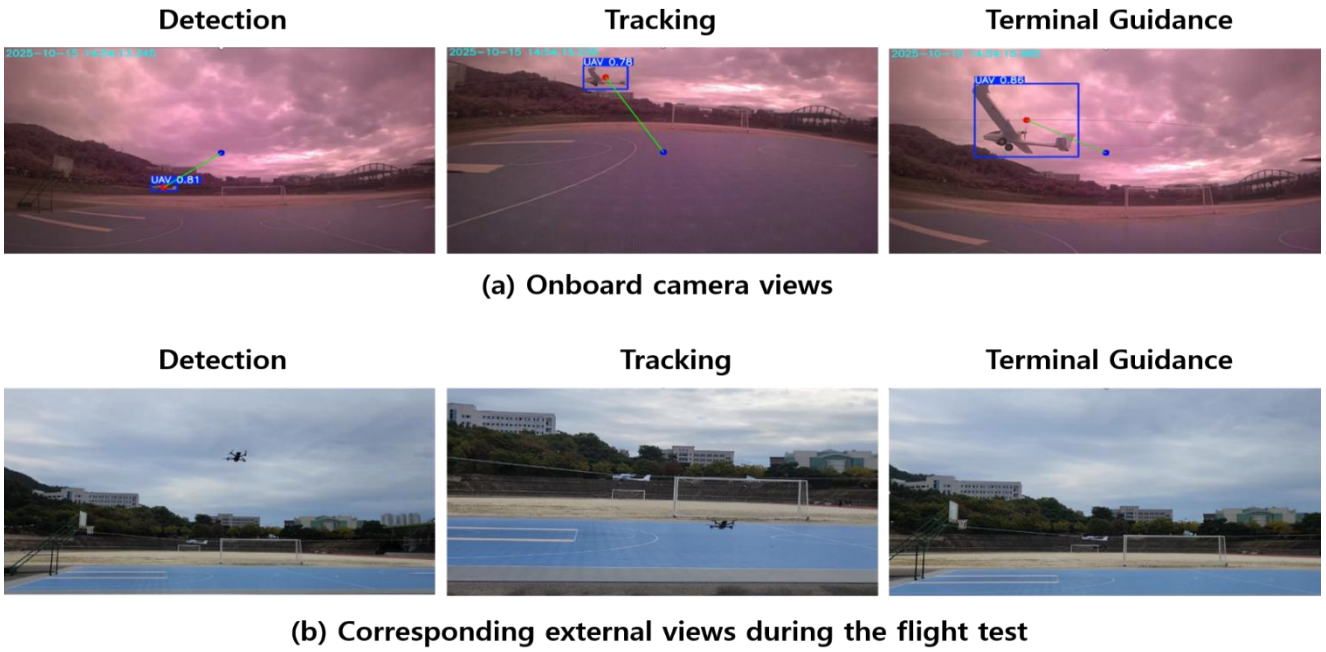


Figure 7 – Representative flight test result of the proposed vision-based terminal guidance system.

The flight test results showed that the interceptor followed a flight path toward the target, and the feasibility of the proposed image-based terminal guidance system was ultimately confirmed through a collision with a stationary aerial target.

Figure 8 shows the collision scene following a successful terminal guidance maneuver and the condition of the aircraft after the collision. Structural damage was observed on both the pursuing aircraft and the target aircraft after the collision, confirming that the proposed image-based terminal guidance system is capable of approaching a target and colliding with it in a real-world flight environment.

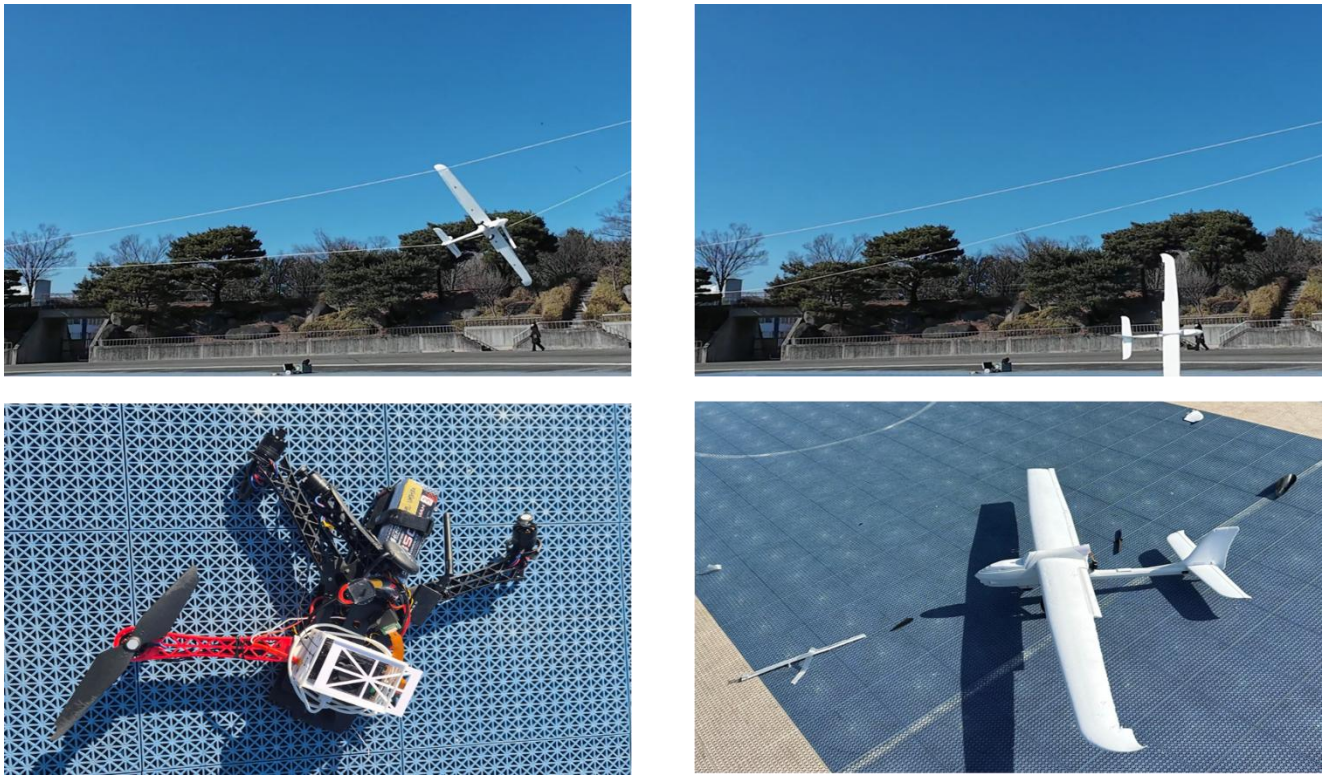


Figure 8 – Flight test results of the proposed vision-based terminal guidance system. The upper images show the collision between the pursuer UAV and the target UAV, while the lower images show the post-collision of both vehicles.

5. Conclusion

In this study, we implemented a vision-based terminal guidance system that utilizes YOLOv8-based object detection results and error information in the image coordinate system to perform terminal guidance. Additionally, we integrated object detection, guidance command generation, and flight control functions by applying a communication architecture based on ROS 2 and MAVLink. Through flight tests using a multicopter platform, we verified the feasibility of target detection, tracking, and terminal guidance functions. The flight test results confirmed that image-based object detection results can be linked with flight control for use in terminal guidance, and the operational feasibility of the proposed system was verified through collisions with static aerial targets. This confirmed that a closed-loop terminal guidance architecture using only image information is applicable in actual flight platform environments.

However, this study was conducted in a static aerial target environment and employed a simple proportional control structure based on the image coordinate system. Therefore, in the future, we plan to conduct flight tests in a maneuvering aerial target environment and apply target motion prediction techniques and sophisticated guidance control algorithms to improve terminal guidance performance and system robustness.

6. Acknowledgement

This paper is research supported by the Ministry of Trade, Industry and Resources (A major activities of 2026 industry-academia cooperation: aviation and defense SW professional education).

7. Contact Author Email Address

If you have any questions, please send an email to: ‘inar3y@cju.ac.kr’ (the first author) or ‘jhmoon@cju.ac.kr’ (the corresponding author).

8. Copyright Statement

The authors confirm that they, and/or their company or organization, hold copyright on all of the original material included in this paper. The authors also confirm that they have obtained permission, from the copyright holder of any third party material included in this paper, to publish it as part of their paper. The authors confirm that they give permission, or have obtained permission from the copyright holder of this paper, for the publication and distribution of this paper as part of the ICAS proceedings or as individual off-prints from the proceedings.

References

[1] S. Wang, F. Jiang, B. Zhang, R. Ma, and Q. Hao, "Development of UAV-Based Target Tracking and Recognition Systems," IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 8, pp. 3409-3422, Aug. 2020, doi: 10.1109/TITS.2019.2927838.

[2] P. Sun, B. Zhu, and S. Li, "Vision-Based Prescribed Performance Control for UAV Target Tracking Subject to Actuator Saturation," IEEE Transactions on Intelligent Vehicles, vol. 9, no. 1, pp. 2382-2389, Jan. 2024, doi: 10.1109/TIV.2023.3302689.

[3] M. Y. Arafat, M. M. Alam, and S. Moh, "Vision-Based Navigation Techniques for Unmanned Aerial Vehicles: Review and Challenges," Drones, vol. 7, no. 2, Art. no. 89, 2023, doi: 10.3390/drones7020089

[4] G. Tang, J. Ni, Y. Zhao, Y. Gu, and W. Cao, "A Survey of Object Detection for UAVs Based on Deep Learning," Remote Sensing, vol. 16, no. 1, Art. no. 149, 2024, doi: 10.3390/rs16010149

[5] Y. Zheng, Z. Chen, D. Lv, Z. Li, Z. Lan, and S. Zhao, "Air-to-Air Visual Detection of Micro-UAVs: An Experimental Evaluation of Deep Learning," IEEE Robotics and Automation Letters, vol. 6, no. 2, pp. 1020-1027, Apr. 2021, doi: 10.1109/LRA.2021.3056059.

[6] S. Duan, T. Wang, T. Li, and W. Yang, "M-YOLOv8s: An Improved Small Target Detection Algorithm for UAV Aerial Photography," Journal of Visual Communication and Image Representation, vol. 104, Art. no. 104289, 2024, doi: 10.1016/j.jvcir.2024.104289.

[7] P. Sun, B. Zhu, Z. Zuo, and M. Basin, "Vision-Based Finite-Time Uncooperative Target Tracking for UAV Subject to Actuator Saturation," Automatica, vol. 130, Art. no. 109708, 2021, doi: 10.1016/j.automatica.2021.109708

[8] H. Choi and Y. Kim, "Vision-Based UAV Guidance and Estimation for Aerial Target Tracking," in Proc. 28th Congress of the International Council of the Aeronautical Sciences (ICAS), Brisbane, Australia,

2012, pp. 3290-3298.

[9] F. Goodarzi, D. Lee, and T. Lee, "Geometric Nonlinear PID Control of a Quadrotor UAV on SE(3)," in Proc. European Control Conference (ECC), Zurich, Switzerland, 2013, pp. 3845-3850, doi: 10.23919/ECC.2013.6669644.

[10] Z. Liu, P. An, Y. Yang, S. Qiu, Q. Liu, and X. Xu, "Vision-Based Drone Detection in Complex Environments: A Survey," Drones, vol. 8, no. 11, Art. no. 643, 2024, doi: 10.3390/drones8110643

[11] Ultralytics, "YOLOv8 Documentation," Accessed: Jun. 15, 2026. [Online]. Available: https://docs.ultralytics.com/models/yolov8/