AUTONOMOUS LANDING OF A QUADROTOR ON A MOVING TARGET WITH OCCLUSION AVOIDANCE MANEUVER

Dabin Kim[1] & H. Jin Kim[2]

[1,2]Department of Aerospace Engineering, Seoul National University

Abstract

The landing phase represents one of the most safety-critical tasks in quadrotor flight, especially when the quadrotor is tasked with landing on a dynamic target, as in applications like truck-drone delivery systems. The potential for occlusions from surrounding obstacles adds complexity to these scenarios. In this paper, we address the challenge of occlusion during autonomous landing through a two-stage solution. Firstly, a feedback motion planning approach is implemented to adjust the path towards the dynamic target, particularly in regions susceptible to occlusion. Secondly, a receding-horizon controller, derived from the reference generated by the feedback motion planner, produces control inputs that track the given reference while simultaneously maximizing visibility of the target. Simulation results demonstrate the effectiveness of the proposed method in generating trajectories that minimize occlusion, ensuring the quadrotor follows feasible controls for a safe landing on the moving target.

Keywords: Quadrotor Control, Autonomous Landing, Feedback Motion Planning, Model Predictive Control

1. Introduction

Quadrotors have been applied in various applications of industry, such as search and rescue [1], transportation [2], and aerial cinematography [3]. Among various maneuvers during flight, landing is one of the most safety-critical tasks, where a small error might result in a crash. However, it becomes more difficult when the quadrotor should land on a dynamic target. For example, in a package shipment with a truck-drone delivery system [4], which is suggested to overcome payload limitations and small battery capacity of the quadrotor, the quadrotor might land on the moving truck after the delivery is finished. Especially, generating an actuator input for safe landing is crucial. Therefore, trajectory planning and control methods are required to safely land the moving quadrotor onto the moving target.

To control the quadrotor for safe landing on the dynamic target, various controllers are designed based on nonlinear control [5], visual servoing methods [6] for direct image feedback, model-predictive control approach [7], and more recently proposed deep reinforcement learning-based approach [8]. Recent works are trying to extend controller design to consider more realistic conditions. [9] applied model predictive control for landing on the ship where wind and wave disturbances are present. [10] designed a robust controller to overcome large wind disturbances.

However, prior research on autonomous landing often assumed an obstacle-free environment, neglecting the realistic possibility of occlusions. The occlusion is particularly pertinent in applications like truck-drone delivery, where urban environments with abundant human-made structures pose a high risk of occlusions during flight. The occlusion is a potential risk for autonomous landing in two aspects. First, for vision-based landing control, losing the target in the image measurement would result in a failure to find proper control input. Second, occlusion hinders communication between the

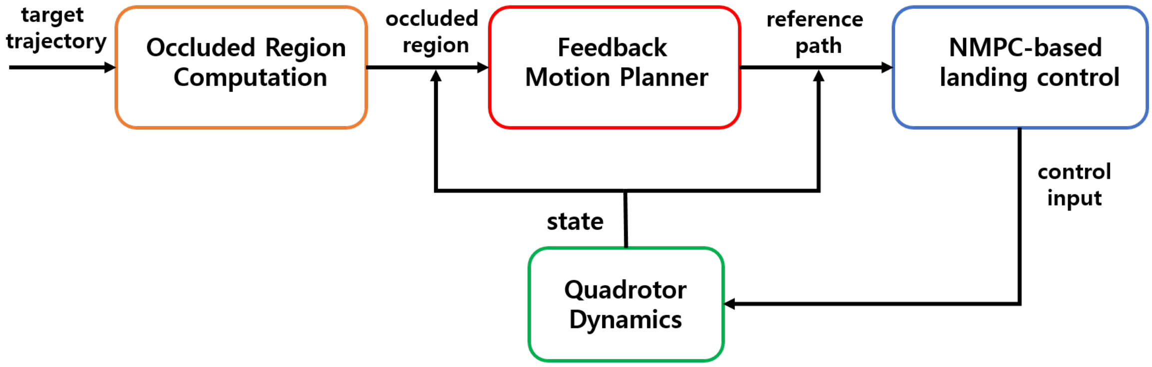


Figure 1 – Planning and control pipeline of the proposed method.

quadrotor and the landing target, where target information is crucial for safe landing control. Therefore, avoiding occlusion while landing on the dynamic target is crucial for safety.

In this paper, we present a planning and control pipeline designed for the autonomous landing of a quadrotor on a moving target, taking into account potential occlusions by obstacles. Initially, we define and compute the occluded region, which is the space where occlusions may occur. The feedback motion planner is proposed, which generates a position path toward the dynamic landing target, strategically modulating the trajectory in proximity to the identified occluded region. After the position path is generated, nonlinear model predictive control (NMPC) is designed to generate dynamically feasible control input that tracks the given reference and safely lands on the target.

2. Method

The method we propose for occlusion-handling autonomous landing on a moving target consists of a two-stage solution. The overall pipeline is depicted in Figure 1. The first module computes the occluded region based on the target trajectory and obstacle geometry. From the computed occluded region, the feedback motion planner generates references to avoid occlusion with a moving platform. The last module is an NMPC-based landing control to follow the given reference and land on the platform.

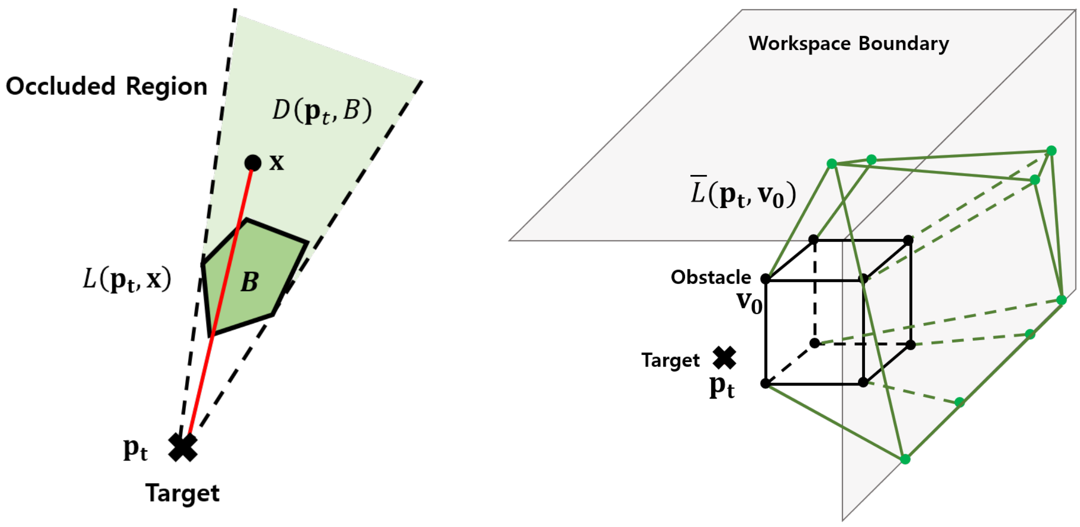
2.1 Feedback Motion Planner for Occlusion Avoidance

2.1.1 Occluded Region

In this section, we elucidate the definition of the occluded region. The occluded region $D(\mathbf{p}_t, B)$ of the target position $\mathbf{p}_t \in \mathbb{R}^3$ and the obstacle $B \subset \mathscr{W} \subset \mathbb{R}^3$ is defined as:

$$D(\mathbf{p}_t, B) = \{\mathbf{x} \in \mathbb{R}^3 | L(\mathbf{p}_t, \mathbf{x}) \cup B \neq \emptyset\}, \tag{1}$$

where $L(\mathbf{p}_t, \mathbf{x}) = \{(1-\gamma)\mathbf{p}_t + \gamma\mathbf{x} | \gamma \in [0,1]\}$ is the line-of-sight (LoS) vector between the target and the quadrotor. We assume that the obstacles are convex polyhedra. We acknowledge that the assumption of convex obstacles may limit applicability, but it is feasible to approximate non-convex obstacles into convex shapes using the convex hull algorithm. For 2D and 3D environments, the occluded regions are graphically depicted in Figure 2. For numerical computation, the extended LoS vector $\overline{L}(\mathbf{p}_t, \mathbf{v}_o) = \{(1-\gamma)\mathbf{p}_t + \gamma\mathbf{v}_o | \gamma \geq 1\}$ is extended from the target to the vertex of the obstacle $\mathbf{v}_o$, and the intersection between the extended LoS vector and the workspace can be found for each vertex. For each vertex of the obstacle B, the intersection between the corresponding LoS vector and the workspace is added as a vertex of the occluded region $D(\mathbf{p}_t, B)$. The resulting occluded region forms a polyhedron but is not necessarily convex, depending on the shape of the workspace boundary. Therefore, we utilize the convex hull of the occluded region as an approximation for efficient computation.



(a) (b)

Figure 2 – (a) Occluded region $D(\mathbf{p}_t, B)$ for a 2D environment. (b) Occluded region computation for a 3D obstacle. From the target (cross), the extended LoS vector (green line) is computed for each vertex (black dot) on the obstacle. The intersection points (green dot) between the extended LoS vectors and the workspace boundary (shaded plane) construct the occluded region with the obstacle.

In the case of a dynamic target, the occluded region varies as the target moves. For safe occlusion avoidance, instead of handling the different occluded regions at each time instance, we consider the union of the occluded regions for the short future time horizon T. We sample a finite number of intermediate times up to the horizon T and compute the occluded region for each sampled time. The union of the sampled occluded regions is non-convex, but explicitly computing non-convex polyhedra might result in a high computational load and difficulties in motion planning. Therefore, we instead construct the over-approximation as the convex hull of all sampled occluded regions. This might result in conservative behavior in motion planning, but since the speed of the landing target is usually small compared to the quadrotor, it is not critical in practice. Also, we can reduce the conservativeness by adjusting the time horizon T. Note that even though a vision-based estimator can predict the future state of the non-cooperative landing target, we assume that the position and intention of the future movement of the landing target are known to the quadrotor based on communication.

2.1.2 Feedback Motion Planner

The Feedback Motion Planner (FMP) is a continuous motion planning method where the path is generated from a vector field rather than discrete waypoints or a continuous function. FMP is designed based on single-integrator kinematics as:

$$\dot{\mathbf{p}} = \mathbf{u} \tag{2}$$

where $\mathbf{p}, \mathbf{u} \in \mathbb{R}^3$ represent position and velocity inputs, respectively. The objective of the FMP is to find a feedback policy $\mathbf{u} = \pi(\mathbf{p}, \mathbf{p}_g)$ that leads the position to the goal position $\mathbf{p}_g$ while satisfying task-specific constraints such as collision avoidance. In this work, the constraint that the FMP needs to satisfy is avoiding occlusion of the landing target.

In [11], the FMP method to avoid an obstacle B is designed as:

$$\pi(\mathbf{p}, \mathbf{p}_g) = M(\mathbf{p}, B) f(\mathbf{p}, \mathbf{p}_g) \tag{3}$$

where $M(\mathbf{p}, B)$ is the modulation matrix that locally modifies the attraction field $f(\mathbf{p}, \mathbf{p}_g) = \mathbf{p}_g - \mathbf{p}$[1]. The modulation-based feedback motion planner can be adopted to avoid the occluded region D, as

[1] The detailed derivation of the modulation matrix M can be found in [11] and will be described in the full paper.

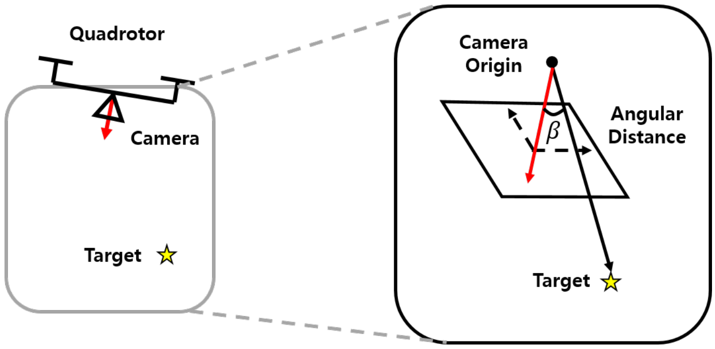


Figure 3 – For the downward-facing camera attached to the quadrotor body, the angular distance β with respect to the target is defined as the angle between the optical axis (red) and the bearing vector pointing toward the target.

the occluded region can be treated as an enlarged obstacle. Since goal convergence and collision avoidance with the feedback law (3) are guaranteed with convex obstacles [11], we can also conclude no occlusion occurs for the static target case. However, when applying the feedback law to the dynamic target case, since the occluded region varies with time, there might be a time instance $k > 0$ where the current position $\mathbf{p}(k)$ is inside the occluded region. In this case, to avoid possible future occlusion, the feedback control is designed to guide the position toward the closest point on the surface of the occluded region D, which is the projection from the current position to the surface of the occluded region, denoted as $\mathbf{p}_{proj}(D)$. The resulting feedback law is defined as

$$\pi(\mathbf{p},\mathbf{p}_g) = \begin{cases} v_n \eta M(\mathbf{p},D) f(\mathbf{p},\mathbf{p}_g) & \text{if } \mathbf{x} \notin D \\ v_n \eta (\mathbf{p}_{proj}(D) - \mathbf{p}) & \text{otherwise} \end{cases} \tag{4}$$

where $\eta > 0$ is a normalizing coefficient and v_n is a nominal velocity.

2.2 NMPC-based landing control

The proposed FMP can generate a continuous path that minimizes occlusion; however, the quadrotor may fail to track the path because it is generated based on simple kinematics. Therefore, to enhance the feasibility of the control command, we have developed a receding horizon control method based on NMPC formulation that simultaneously tracks the reference and generates dynamically feasible control input.

The full state of the quadrotor is denoted as $\mathbf{x} = [\mathbf{p}^T, \mathbf{q}^T, \mathbf{v}^T, \omega^T]^T \in \mathbb{R}^{13}$, where $\mathbf{q} \in \mathbb{R}^4$ represents the quaternion of the quadrotor, $\mathbf{v} \in \mathbb{R}^3$ is the velocity, and $\omega \in \mathbb{R}^3$ is the angular velocity with respect to the quadrotor body frame. The input $\mathbf{w} \in \mathbb{R}^4$ is a vector of rotor thrust. The nonlinear discrete-time dynamics of the quadrotor is expressed as

$$\mathbf{x}(k+1) = f(\mathbf{x}(k), \mathbf{w}(k)). \tag{5}$$

The details of the function $f : \mathbb{R}^{13} \times \mathbb{R}^4 \to \mathbb{R}^{13}$ can be found in [12].

For the MPC horizon $H \in \mathbb{N}$ and the initial position $\mathbf{p}(0)$, the reference path $\{\mathbf{p}_r(k)\}_{k=1:H}$ is generated from the FMP (4) by forward integration of an ordinary differential equation, where $\mathbf{p}_r(k) = [\mathbf{p}_r^T(k), \mathbf{q}_r^T(k), \mathbf{v}_r^T(k), \omega_r^T(k)]^T$. Specifically, the velocity component of the reference path $\mathbf{v}_r$ is given from (4) and the position component $\mathbf{p}_r$ is computed from forward integration of the velocity. Attitude component $\mathbf{q}_r$ is unspecified from the FMP and set as stable equilibrium $\mathbf{q}_r = [1,0,0,0]^T$, and the angular velocity component as $\omega_r = [0,0,0]^T$.

With the given reference path $\{\mathbf{p}_r(k)\}_{k=1:H}$, the optimization problem for the receding-horizon con-

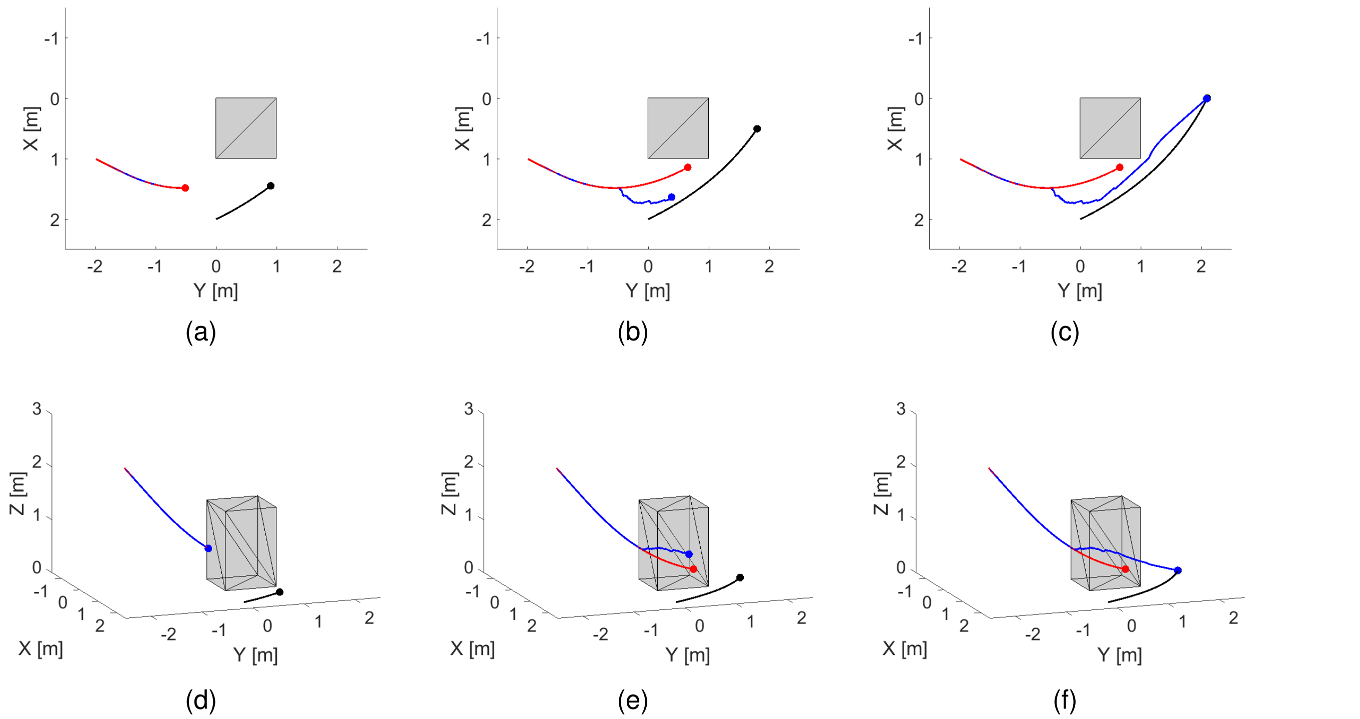


Figure 4 – Snapshots of the simulation results at three different time instances (a-d), (b-e), and (c-f), respectively. The proposed FMP (blue line) tracks the dynamic target (black line) without occlusion, whereas the attraction field (red line) fails due to a lack of consideration about occlusion.

trol is formulated as,

$$\min_{\{\mathbf{w}(k)\}_{k=1:H}} \sum_{k=1}^{H} J_t(\mathbf{x}(k), \mathbf{w}(k), \mathbf{p}_r(k)) + J_{fov}(\mathbf{x}(k)) \tag{6}$$

$$\text{such that} \quad \mathbf{x}(k+1) = f(\mathbf{x}(k), \mathbf{w}(k)), \quad \text{for } k \in \{1, \cdots, H\} \tag{7}$$

where the tracking cost J_s and the Field-of-View (FoV) cost J_{fov} are defined as explained in the following subsections.

2.2.1 Tracking cost

The tracking cost J_t is defined for tracking of a given reference with minimal thrust as:

$$J_t(\mathbf{x}(k), \mathbf{w}(k), \mathbf{p}_r(k)) = k_p\|\mathbf{p}(k) - \mathbf{p}_r(k)\|^2 + k_q d(\mathbf{q}, \mathbf{q}_r)^2 + k_v\|\mathbf{v}(k)\|^2 + k_\omega\|\omega(k)\|^2 + k_w\|\mathbf{w}(k)\|^2, \tag{8}$$

where $\mathbf{q}_r$ is the quaternion at the hovering state, $d(\mathbf{q}, \mathbf{q}_r)$ is the geodesic distance on the quaternion space, and $k_p, k_q, k_v, k_\omega, k_w > 0$ are weight coefficients.

2.2.2 Field-of-View cost

In this work, we assume that a downward-facing camera is attached to the quadrotor body. The FoV cost is designed to control the quadrotor state to position the target inside the FoV of the camera. This is encouraging, as in vision-based landing, the detection of the target is more stable when the target is located near the center of the image plane [13]. The angular distance β between the target in the camera frame and the optical axis of the camera is used as a metric to indicate how near the target is to the center of the image plane, as shown in Figure 3. The FoV cost J_{fov} is defined as:

$$J_{fov}(\mathbf{x}(k)) = k_{fov}(1 - \cos\beta)^2 \tag{9}$$

where $k_{fov} > 0$ is a coefficient.

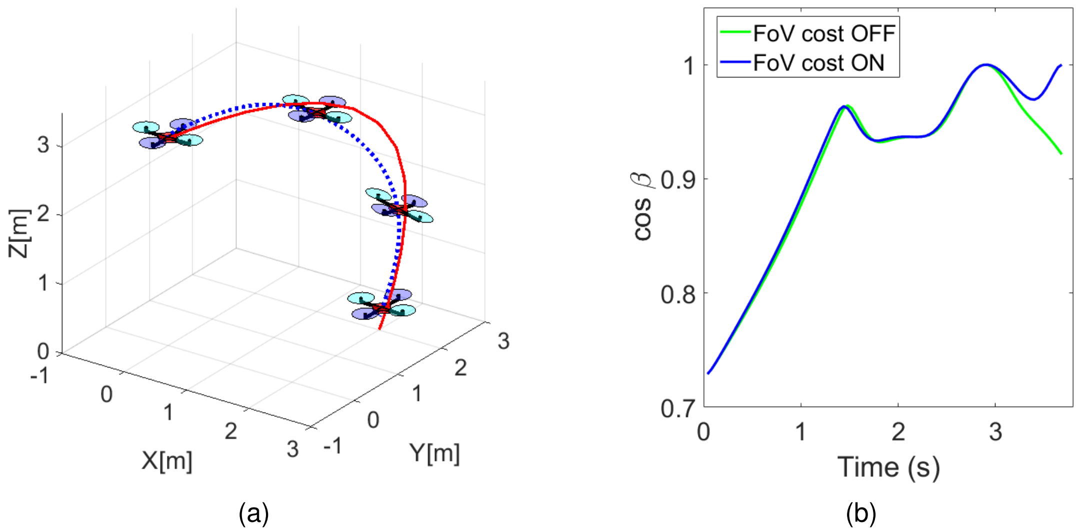


Figure 5 – (a) The trajectory generated by the landing MPC (blue line) tracking the reference trajectory (red line). (b) Cosine of the angular distance with respect to the landing target from the trajectory of the landing MPC, comparing cases with and without the FoV cost.

3. Evaluation

In this section, results from the implementation of the proposed method are presented to validate its efficacy for autonomous landing on a dynamic target. First, each module—the FMP and the NMPC-based landing control—was individually validated, and the demonstration of integrated planning and control is provided.

3.1 Feedback Motion Planning

To validate the proposed feedback motion planner, we apply the feedback law (4) in an environment with a cuboid obstacle, and the target follows a predefined trajectory. We assume that position and future trajectory data are accessible when the target is not occluded by the obstacle. The results of the numerical simulation are depicted in Figure 4. The single-integrator agent starts at $\mathbf{p} = [0, -2, 2]^T$, and the target follows a polynomial trajectory from $\mathbf{p}_t = [2, 0, 0]^T$ to $\mathbf{p}_t = [0, 2.1, 0]^T$, with a nominal velocity $v_n = 0.9\ m/s$. The proposed FMP is compared with the FMP derived from an attraction field (AF) toward the target. When the target is visible from the agent, both FMPs show the same maneuver. However, as the target approaches the obstacle, the proposed FMP avoids possible occlusion by deviating from a simple attracted direction, while the AF results in occlusion. The AF fails to reach the target and stops since it cannot obtain target information due to occlusion, whereas the proposed FMP succeeds in rendezvousing with the target.

3.2 Landing with NMPC

The implementation of landing control with NMPC is applied to track a continuous trajectory toward the static landing target, as illustrated in Figure 5(a). The reference trajectory is generated from the starting point ($\mathbf{p} = [0, 0, 3]^T$) to the landing target ($\mathbf{p}_t = [2, 2, 0]^T$). The landing MPC generates input in a receding-horizon manner for every time step, resulting in a trajectory that successfully lands on the target while tracking the reference trajectory.

Additionally, we compared the MPC trajectory with and without the FoV cost to assess the effect of adding the FoV cost to the optimization problem. The cosine value of the angular distance of the target is measured to indicate how close the target is located to the center of the image. As shown in Figure 5 (b), using the FoV cost enables a smaller angular distance, especially near the landing target, where the angular distance can abruptly change as the distance between the quadrotor and the target decreases.

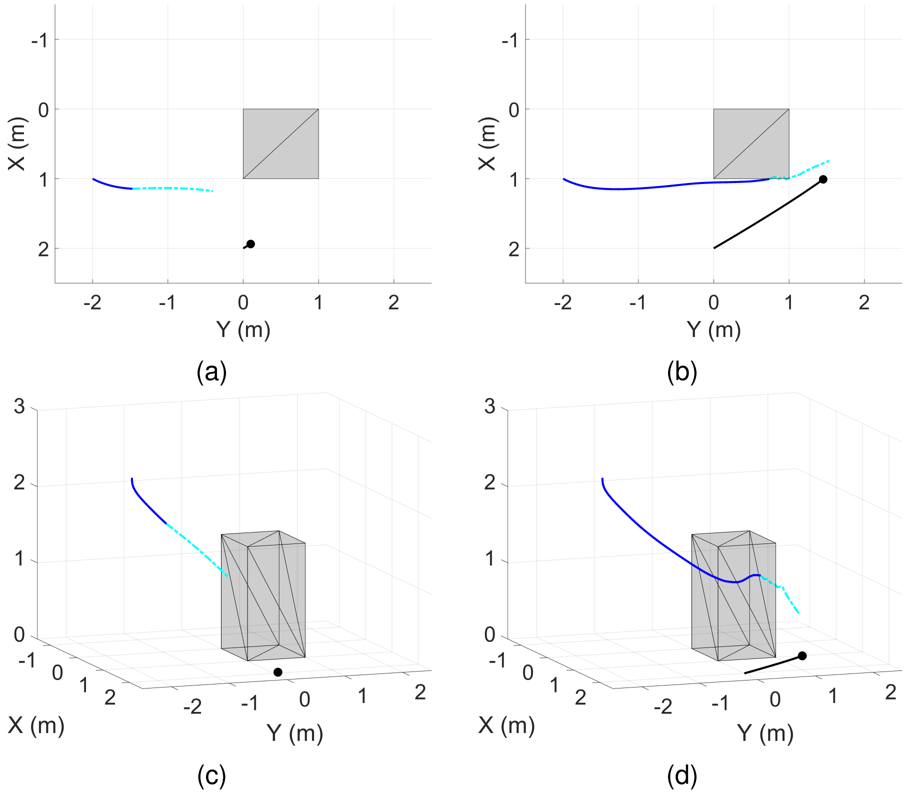


Figure 6 – Snapshots of the simulation results for the baseline control without consideration of occlusion constraints at two different time instances: (a-c) and (b-d). The multirotor trajectory (blue line) tracks the reference path (dashed cyan line) generated by the feedback motion planner, which follows the target (black dot) and avoids the obstacle. However, due to the absence of occlusion constraint consideration, it encounters occlusion and fails to reach the target.

3.3 Integrated Planning & Control

We have developed an integrated planner and controller using the FMP and the landing MPC. In our landing scenario, the FMP generates a reference path while the MPC computes control inputs to track it, allowing the quadrotor to navigate the environment and track a moving target simultaneously. This integrated algorithm effectively guides the quadrotor to land safely on the moving target, even in the presence of obstacles.

For communication between the quadrotor and the target, we assume that the quadrotor receives the current position, velocity, and future trajectory of the target. However, in cases of occlusion, where the target is not visible, the quadrotor cannot receive this information, leading to a failure to reach the target.

The reference position path is generated based on the proposed feedback motion planner, which predicts the future position of the target and attempts to reach it. We set the nominal velocity for the feedback motion planner as $v_{nom} = \min(1.0, |\mathbf{p} - \mathbf{p}_t|^2)$, aiming to lower the quadrotor's velocity for a safe landing while reaching the target. For the MPC parameters, we use coefficients defined in (8) as $k_p = 30.0$, $k_q = 0.1$, $k_v = 10.0$, $k_w = 0.1$, and $k_u = 10.0$. Additionally, the coefficient for the FoV cost is set as $k_{fov} = 20.0$, with a prediction horizon of $H_T = 1.0s$ and the number of sampling nodes as $H = 15$.

We compare our proposed method with a baseline controller that does not utilize the occlusion-aware feedback motion planner. Instead, it relies on obstacle avoidance using the obstacle polyhedron defined in Sec. 2.1.1 but does not consider target occlusion. Additionally, the FoV cost coefficient for the baseline controller is set to $k_{fov} = 0$. The results of the baseline controller are presented in Fig. 6. While the feedback motion planner successfully generates a collision-free path, it fails to reach the target due to a lack of consideration for target occlusion.

On the other hand, the proposed method, as shown in Fig. 7, utilizes the occlusion-aware feedback motion planner to generate a reference path that helps the MPC avoid target occlusion. This results

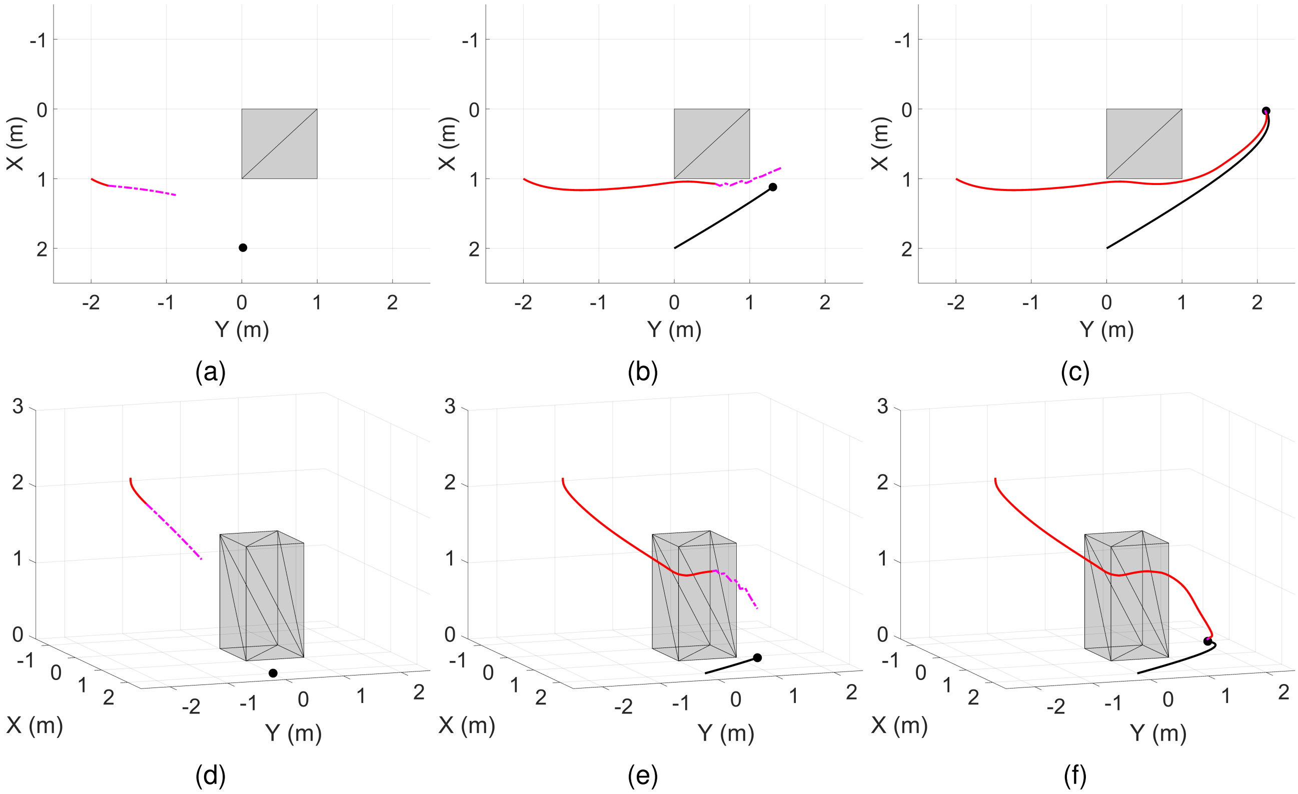


Figure 7 – Snapshots of the simulation results of the proposed controller at three different time instances: (a-d), (b-e), and (c-f), respectively. The proposed MPC generates the multirotor trajectory (red line) to track the reference path generated by the proposed FMP (dashed magenta line) and follow the dynamic target (black line) without occlusion. In contrast to the baseline controller, it successfully avoids occlusion and safely lands on the target.

in a smooth trajectory that safely lands on the target.

To evaluate the effect of the field-of-view cost (9), we compare three different values of the FoV cost coefficient: $k_{fov} = 0, 10, 20$. The results are shown in Fig. 8. While there is negligible difference in position trajectories, the attitude trajectories, especially the yaw trajectories, vary based on the FoV cost coefficient. The proposed feedback motion planner only handles the translational part of the reference path, setting the attitude reference to a constant value. As roll and pitch angles are directly correlated with acceleration, while yaw angle is not related to translation, the MPC has the freedom to adjust the yaw trajectory. A larger k_{fov} results in a larger $\cos\beta$ value (as in Fig. 8 (b)), indicating the possibility of positioning the target near the center of the image plane, which is favorable for robust target detection.

4. Conclusion

In this work, we propose a planning and control method for the autonomous landing of a quadrotor on a moving target. Unlike previous studies on autonomous landing, our approach considers occlusion from obstacles, which is crucial for successful landings in urban-like environments. We address this challenge by dividing the problem into two separate modules.

Firstly, we introduce a feedback law designed to control the agent's position towards the dynamic target while avoiding occlusion. Secondly, we propose a nonlinear model predictive control (NMPC)-based controller to track the reference path generated by the occlusion-avoiding feedback law and maximize the visibility of the landing target.

Numerical simulations demonstrate that the proposed feedback law effectively avoids occlusions caused by obstacles while converging towards the dynamic landing target. Furthermore, the NMPC-based landing control successfully tracks the given reference, ensuring a safe landing on the target.

As for future work, we plan to implement the autonomous landing control algorithm with real hardware and a vision-based detector for the landing target.

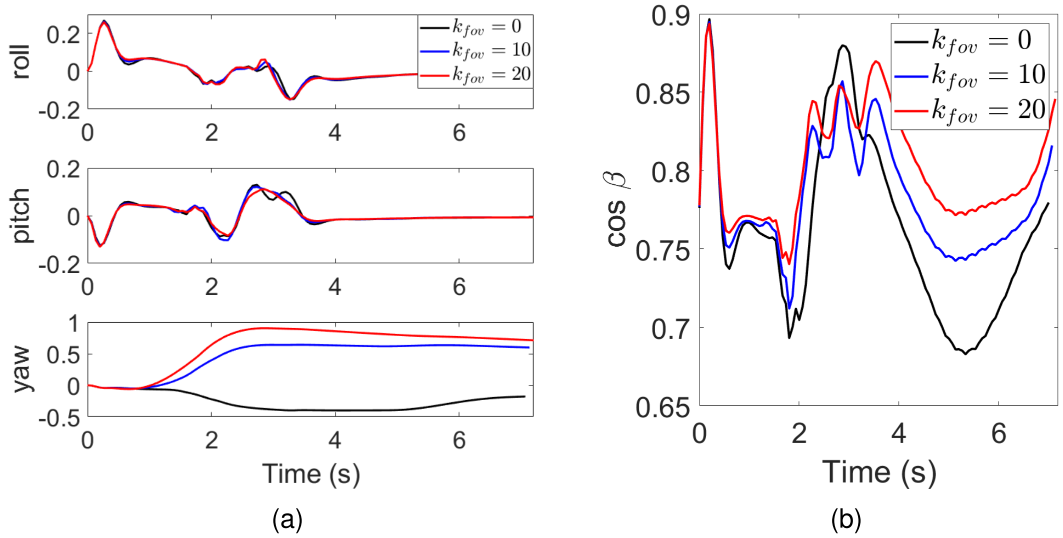


Figure 8 – Comparison results with different k_{fov} parameters: (a) attitude trajectories and (b) cosine of the angular distance relative to the landing target from the trajectory of the proposed algorithm.

5. Acknowledgement

This research was supported by Unmanned Vehicles Core Technology Research and Development Program through the National Research Foundation of Korea(NRF) and Unmanned Vehicle Advanced Research Center(UVARC) funded by the Ministry of Science and ICT, the Republic of Korea(NRF2020M3C1C1A010864)

6. Author Information

- Dabin Kim
  - Affiliation: Department of Aerospace Engineering, Seoul National University
  - Mailing address: Bldg. 300, Rm. 504, 1 Gwanak-ro, Gwanak-gu, Seoul, 08826, South Korea
  - Tel: +82-10-7151-5199
  - Email: dabin404@gmail.com, dabin404@snu.ac.kr
- H. Jin Kim
  - Affiliation: Department of Aerospace Engineering, Seoul National University
  - Mailing address: Bldg. 300, Rm. 504, 1 Gwanak-ro, Gwanak-gu, Seoul, 08826, South Korea
  - Tel. +82-2-880-1552
  - Email: hjinkim@snu.ac.kr

7. Copyright Statement

References

[1] Waharte S, and Trigoni N. Supporting search and rescue operations with UAVs. *2010 international conference on emerging security technologies*, Canterbury, UK, Vol. 1, pp 142-147, 2010.

[2] Kim H, Lee H and Kim H. J. Sampling-based Motion Planning for Cooperative Aerial Manipulation. *International Council of the Aeronautical Sciences (ICAS) Congress*, Daejeon, Korea, Vol. 1, pp 1-7, 2016.

[3] Galvane Q, Lino C, Christie M, Fleureau J, Servant F, Tariolle F. L, and Guillotel P. Directing cinematographic drones. *ACM Transactions on Graphics (TOG)*, Vol. 37, No. 3, pp 1-18, 2018.

[4] Ham A. M. Integrated scheduling of m-truck, m-drone, and m-depot constrained by time-window, drop-pickup, and m-visit using constraint programming. *Transportation Research Part C: Emerging Technologies* Vol. 91, No. 1, pp 1-14, 2018.

[5] Gautam, Alvika, Mandeep Singh, Pedda Baliyarasimhuni Sujit, and Srikanth Saripalli. "Autonomous quadcopter landing on a moving target." Sensors 22, no. 3 (2022): 1116.

[6] Wynn J, and McLain T, Visual servoing with feed-forward for precision shipboard landing of an autonomous multirotor. *2019 American Control Conference (ACC)*, Philadelphia, USA, Vol. 1, pp 3928–3935, 2019.

[7] Baca, Tomas, Petr Stepan, Vojtech Spurny, Daniel Hert, Robert Penicka, Martin Saska, Justin Thomas, Giuseppe Loianno, and Vijay Kumar. "Autonomous landing on a moving vehicle with an unmanned aerial vehicle." Journal of Field Robotics 36, no. 5 (2019): 874-891.

[8] Rodriguez-Ramos, A.; Sampedro, C.; Bavle, H.; de la Puente, P.; Campoy, P. A Deep Reinforcement Learning Strategy for UAV Autonomous Landing on a Moving Platform. J. Intell. Robot. Syst. 2018, 93, 351–366

[9] Persson L and Wahlberg B. Model predictive control for autonomous ship landing in a search and rescue scenario. *AIAA Scitech 2019 Forum*, p. 1169, 2019.

[10] Aleix P, Lopez B, and How J. Dynamic landing of an autonomous quadrotor on a moving platform in turbulent wind conditions. *2020 IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France, Vol. 1, pp 9577-9583, 2020.

[11] Huber L, Billard A, and Slotine J.-J. Avoidance of convex and concave obstacles with convergence ensured through contraction, *IEEE Robotics and Automation Letters*, Vol. 4, No. 2, pp 1462–1469, 2019.

[12] Kim D, Pezzutto M, Schenato L, Kim H. J. Visibility-Constrained Control of Multirotor via Reference Governor. *62nd IEEE Conference on Decision and Control (CDC)*, Singapore, Vol. 1, 2023.

[13] Jacquet M and Franchi A. Motor and perception constrained NMPC for torque-controlled generic aerial vehicles, *IEEE Robotics and Automation Letters*, Vol. 6, No. 2, pp 518-525, 2020.