

Vision-based Autonomous Landing Using an MPC-controlled Micro UAV on a Moving Platform*

Alireza Mohammadi[†], Yi Feng[†], Cong Zhang[†], Samir Rawashdeh[†], and Stanley Baek[‡]

Abstract— Autonomous landing of micro unmanned aerial vehicles (UAVs) on moving targets has the potential to resolve many limitations of small-scale UAVs, such as uninterrupted flight tasks, rapid deployment and recovery of multiple UAVs, and extended operational ranges through mobile recharging stations. In this work, we present and experimentally verify a new vision-based method that enables a micro UAV to land autonomously on a mobile landing platform. Our method, which can be implemented on small-scale UAVs with limited payload capabilities and computational resources, incorporates model predictive control, vision-based localization, and extended Kalman filter for path following, navigation, and guidance. Our method uses a closed-loop controlled gimbaled camera for visual navigation and relative localization of the landing platform, a sensor fusion technique based on extended Kalman filters for target localization, and a model predictive control scheme for autonomous landing of the UAV under system uncertainties and wind disturbances. We demonstrate flight experiments of autonomous landing with an average error of 39 cm from the center of a mobile platform.

I. INTRODUCTION

During the past few decades, micro aerial vehicles (MAVs) have been extensively employed in various applications, including surveying and mapping, rescue operation in disasters, building inspection, geophysics exploration, traffic monitoring, animal protection, and agricultural crop monitoring, to name a few [1]. Despite their numerous potential applications, there are several obstacles for making MAVs ubiquitous such as insufficient onboard computing power and limited operational range, which is mainly due to limited payload and battery capacity. To overcome the limited onboard computing power and operational range issues, researchers have suggested power support using tethers [2] for limited operation range, solar photovoltaic (PV) panels [3] for fixed-wing unmanned aerial vehicles (UAVs), and stationary charging stations [4].

An alternative solution for sustained MAV missions over long flight ranges and duration is based on takeoff from and landing on mobile platforms that are equipped with charging or battery swapping stations. Such a mobile landing platform can also be used to reconfigure the UAV depending

*Distribution A. Appeared for Public Release: Distribution Unlimited. The views expressed in this article are those of authors and not necessarily those of the U.S. Air Force Academy, the U.S. Air Force, the Department of Defense, or the U.S. Government. PA#: USAFA-DF-2020-68

[†]A. Mohammadi, Y. Feng, C. Zhang, and S. Rawashdeh are with the Department of Electrical and Computer Engineering, University of Michigan-Dearborn, Dearborn, MI 48128, USA

[‡]S. Baek is with the Department of Electrical and Computer Engineering, United States Air Force Academy, USAF Academy, CO 80840, USA

on the mission. For instance, emergency vehicles, delivery trucks, and marine or ground carriers could be used not only for deploying UAVs between desired locations but also for mobile charging UAVs [4]–[8]. In large-scale UAV operations, autonomous takeoff and landing can happen on moving aircraft carriers or naval ships [9]–[11].

The main challenges in autonomous landing on mobile platforms are twofold. First, the relative pose between the landing platform and the UAV should be accurately estimated. Second, robust trajectory following under uncertainties and disturbances should be achieved. To overcome the aforementioned difficulties, the following three approaches, which are based on computer vision, state estimation, and closed-loop trajectory tracking control have been proposed by various researchers.

With regards to computer vision-aided autonomous landing, optical flow methods for target following have been extensively used to calculate the linear and angular velocity of UAVs to realize indoor autonomous landing [12]. Lee *et al.* [13] have presented an image-based visual servoing (IBVS) system for indoor UAVs to track a landing pad. Along this line of research, Serra *et al.* [14] have adopted dynamic IBVS along with transitional optimal flows for velocity measurement. As an alternative to IBVS-based methods, Brommer *et al.* [15] have integrated AprilTags [16], which is a visual fiducial system, to autonomously land on a stationary recharging station for automated energy replenishment. Beul *et al.* [17] have employed two cameras for achieving high-frequency pattern detection along with an adaptive yawing algorithm to land autonomously on a golf cart moving with a speed of approximately 4.2 m/s. Although computer vision-aided target tracking is a promising solution, to follow a cooperative landing platform under various wind speeds and lighting conditions, a sensor fusion technique using a platform-mounted GPS and a gimbaled camera onboard UAVs can significantly increase the robustness of target localization.

With regards to state estimation-aided autonomous landing, extended Kalman filter (EKF) and particle filter (PF) have proven to be effective in autonomous landing problem (see, e.g., [18]–[22]). Whereas particle filter can approximate multi-modal probability distributions, EKFs do not require the expensive re-sampling computation associated with PFs. In this work, we consider landing platforms, which can be considered as cooperative agents, that move on linear or slightly curved paths without aggressive motions. Consequently, EKF is a better choice for position estimation of

the landing platform.

With regards to trajectory tracking approaches for autonomous landing, Erginer and Altug [23] have combined an attitude control PD scheme with a vision-based tracking method. Their proposed method enables quadcopters to land autonomously on stationary landing pads. Voos and Nourghassemi have presented a control system including an inner attitude control loop, an outer velocity and altitude proportional control loop, and a 2D-tracking controller for autonomous landing of a quadrotor UAV on a mobile platform based on feedback linearization [24]. Borowczyk *et al.* [25] have integrated a proportional-integral-derivative (PID) controller with a visual fiducial system, an IMU, and a GPS receiver. Ahmed and Pota have proposed an extended backstepping nonlinear control for landing of rotary-wing UAVs that are attached to their mobile platforms via tethers [26]. In addition to conventional nonlinear control schemes for autonomous landing, learning-based and intelligent control methods such as fuzzy logic-based controllers [27]–[29] and adaptive neural networks [30]–[33] have also been employed to achieve optimal control policies under nonlinear aerodynamic effects, uncertainties, and disturbances. However, a control loop with a model-predictive controller (MPC) would be a better choice for this application because it is fairly easy to implement in real-time from a computational perspective, and yet has been shown to be exceptionally robust with respect to exogenous disturbances.

Model predictive control (MPC), which belongs to a class of optimal-control techniques called open-loop optimal-feedback control, has also been used for solving the autonomous landing problem in [32]–[35]. MPC-based controllers can iteratively compute the open-loop optimal control input to MAVs by using repeated feedback updates while satisfying multiple linear constraints on velocity and attitude of MAVs. Furthermore, the prediction for the MAV behavior provides a feed-forward compensation during the landing maneuver. Templeton *et al.* [34] proposed an MPC-based terrain mapping and analysis system in order to autonomously land a helicopter on an unprepared terrain. Samal *et al.* [35] proposed a neural network-based model predictive controller to handle disturbances and parameter uncertainties for altitude control of unmanned helicopters. Tian *et al.* [36] have presented an MPC combined with genetic algorithm (GA) to solve a cooperative search problem for UAVs. Wu *et al.* [37] have implemented a model predictive controller for obstacle avoidance and route planning for carrier aircraft launching. Raffo *et al.* [38] employed MPC along with an integral action for steady state error correction and disturbance rejection in UAVs. In [39], linear MPC (LMPC) schemes are demonstrated to be capable of achieving target-following performance that are comparable with nonlinear MPC (NMPC) schemes while relieving the computational burden associated with NMPC controllers.

In our prior work [40], we have designed a family of LMPC controllers for solving the UAV autonomous landing problem and verified our proposed solution using a high-fidelity hardware-in-the-loop (HIL) simulation environment.

In a similar work by Fu *et al.* [41], a combination of visual servoing and GPS precision system has been employed to demonstrate the feasibility of autonomous landing of UAVs on unmanned ground vehicles (UGVs) in simulation carried out in ROS and V-REP environments. A similar autonomous landing system has also been discussed in [42].

In this paper, we extend our prior results in [40] and provide a complete solution, *with experimental verification*, to the autonomous MAV landing problem on moving platforms. In particular, we present a new system architecture that can be implemented on MAVs that have insufficient payload capabilities and onboard computational resources for navigation, perception, guidance, and sensor fusion. To this end, we propose using (i) an EKF-based optimal sensor fusion algorithm for target localization using visual perception with a closed-loop controlled gimbaled camera and (ii) an MPC-based guidance and control system for an MAV to land autonomously on a moving landing platform under dynamic uncertainties and external disturbances such as wind gusts. We have validated the efficacy of our EKF-MPC-based sensor fusion and path following control approach with numerous hardware-in-the-loop simulations and flight experiments where autonomous landing with an average error of 39 cm from the center of a mobile platform has been achieved under the influence of wind disturbances in outdoor environments.

The rest of the paper is organized as follows. In Section II we present a complete system architecture including the hardware and software systems. Section III presents (i) vision-based target position measurement, combined with extended Kalman filter for optimal target position estimation, (ii) dynamic model of the UAV, (iii) model predictive control for guidance of the UAV and integral control for robustness. Section IV presents flight experiments results to validate our approach. Finally, Section V concludes the paper with some additional remarks and future research directions.

II. SYSTEM OVERVIEW

A. Hardware Setup

In this section, we provide an overview of our UAV autonomous landing hardware setup. The UAV used in this work is a DJI Matrice 100 quadrotor, modified to have long distance Wi-Fi communication capability (see Fig. 1). It is equipped with a flight control system (autopilot), an onboard computer called DJI Manifold (which houses a quad-core ARM processor and an NVIDIA Kepler GPU), a Ubiquiti Picostation for Wi-Fi communication, and multiple sensor modules including a gimbaled camera, a 9-axis inertial measurement unit (IMU), a DJI Guidance navigation module, and a GPS receiver.

The role of the onboard computer is to perform sensor fusion, localization, path planning and control. Since the load carrying capability of the UAV is limited, the image processing module is installed in the ground station computer, which is carried by the mobile landing platform. As shown in Fig. 2, a high performance computer in the ground station, namely, an Intel NUC6i7, receives all the flight data for aircraft

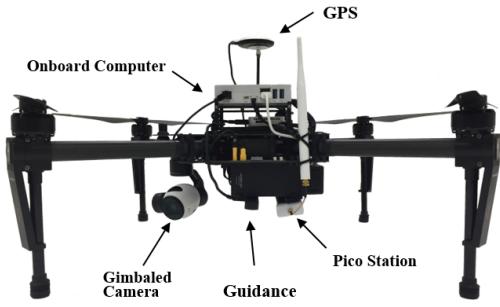


Fig. 1. DJI M100 quadcopter with a 3-axis gimbaled camera, a 9-axis IMU, a Guidance vision-based navigation module, an NVIDIA onboard processor, and a Pico Station wireless module. Guidance module is used for height measurements.

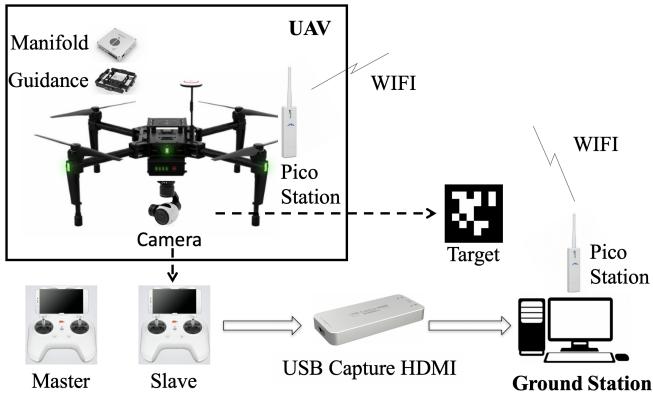


Fig. 2. Hardware system configuration

monitoring and real time video streaming through a slave remote controller and video encoder for image processing. The camera can provide 4K videos at frame rates of up to 30 fps or 1080P at up to 60 fps which enables us to track the target visually. DJI Guidance is an obstacle detection sensor module consisting of a vision processing core, stereo cameras, and ultrasonic sensors. It was used to accurately measure the distance between the UAV and the landing platform to precisely determine whether the UAV has landed or not.

The mobile landing platform used in this work is equipped with an embedded system interfaced with a GPS receiver, an IMU, and a WiFi module that can transmit the position and velocity of the landing platform to the UAV at 10 Hz.

B. Coordinate Systems

The coordinate systems used in this work are shown in Fig. 3. The coordinates of the MAV are denoted by $\{M\}$ and they are defined as the \hat{X}^m axis pointing through the nose of the UAV, the \hat{Y}^m axis pointing to the right of the \hat{X}^m axis, and the \hat{Z}^m axis pointing down through the bottom of the UAV. The coordinates of the gimbaled camera are denoted by $\{G\}$ and they are defined as \hat{X}^m pointing to the right (camera view), the \hat{Y}^m axis up, and the \hat{Z}^m axis

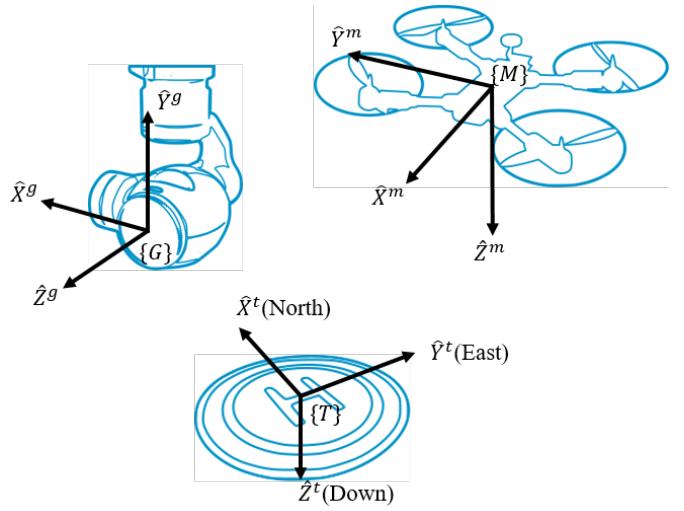


Fig. 3. Coordinate frames of the UAV, gimbaled camera, and the landing pad

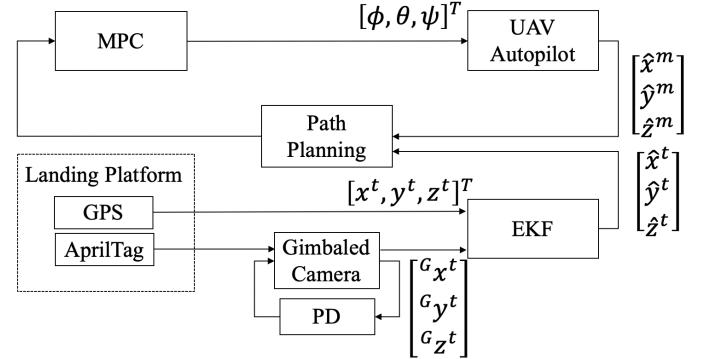


Fig. 4. The block diagram of the control system architecture.

toward the target. The coordinates of the landing platform are defined as $\{T\}$ and they are defined as the \hat{X}^t axis pointing to the north, the \hat{Y}^t axis pointing to the east, and the \hat{Z}^t axis pointing down.

C. System Configurations

In this section, we provide an overview of the control architecture of our system. As shown in Fig. 4, an MPC and a path planner form a closed-loop system for path planning and navigation. An EKF has been used to fuse the measurements from gimbaled camera and GPS to obtain the optimal estimates of location of the landing platform. We have used robot operating system (ROS) as middleware for the software architecture and communication framework.

To estimate the position and the orientation of the landing platform, we have adopted an efficient and robust fiducial marker called, AprilTags [16], which is securely fixed on the landing platform (see Fig. 5). The mobile landing platform with an AprilTag (21.5 cm \times 21.5 cm) is shown in Fig. 5 b. We have used a ROS software package to obtain the position and orientation of the tag with respect to the camera frame, with the calibrated intrinsic parameters of the camera and the size of AprilTag. With the onboard DJI manifold computer,

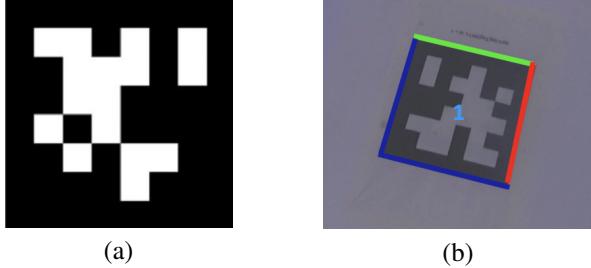


Fig. 5. An example of AprilTags. (a) the AprilTag we used in this work; (b) the AprilTag detected during a flight test. The red-green-blue box indicates AprilTag detection and the number inside the box represents the identification number of the AprilTag.

the AprilTag detection runs at 4.5 Hz. We improved the frame rate to 30 Hz by processing the images on our moving ground station and transmitted the pose estimation result back to the UAV using socket programming.

Once an AprilTag is detected, the reference gimbal angles are calculated and sent to a proportional derivative (PD) controller in Gimbal Control module. The PD control law will then make the gimbaled camera to track the landing platform in a timely fashion.

The flight control system (Autopilot) has an embedded EKF, which provides the position, velocity, and acceleration of the UAV at 50 Hz. After fusing the telemetry data from Guidance we can monitor and control the UAV flight behavior (position control, velocity control, waypoint control, etc.) with the provided API functions at frequencies of up to 50 Hz.

III. MODELING

This section describes the dynamic model of the UAV used in this work, the EKF formulation, and the MPC controller. The EKF provides the optimal estimate of the target position to the MPC module for computing the optimal control inputs.

A. Target Localization

As the UAV approaches the landing platform from a long distance, the UAV enters the Approach state, where our target localization algorithm relies on the position data transmitted from the landing platform. The position data are measured by the GPS receiver integrated with the landing platform. Once the distance between the UAV and the landing platform becomes less than a certain threshold (around 7 m) for the landing platform to be observed by the onboard gimbaled camera, the camera is used to calculate the position of an AprilTag on the landing platform relative to the UAV. This measurement occurs on a frame-by-frame basis with some noise. To optimally fuse, estimate, and predict the location of the mobile landing platform, we have formulated an extended Kalman filter (EKF).

The equations of motion of the landing platform are given

by

$$p_{k+1}^{t,n} = p_k^{t,n} + \Delta T v_k^t \cos \theta_k^t + \frac{1}{2} \Delta T^2 a_k^t \cos \theta_k^t \quad (1a)$$

$$p_{k+1}^{t,e} = p_k^{t,e} + \Delta T v_k^t \sin \theta_k^t + \frac{1}{2} \Delta T^2 a_k^t \sin \theta_k^t \quad (1b)$$

$$v_{k+1}^t = v_k^t + \Delta T a_k^t \quad (1c)$$

$$\theta_{k+1}^t = \theta_k^t + \Delta T \omega_k^t \quad (1d)$$

where $p_k^{t,n}$ and $p_k^{t,e}$ are the discrete-time position of the landing platform in the north and east direction, respectively, at time k ; v_k^t , θ_k^t , ω_k^t and a_k^t are, respectively the speed, heading, angular velocity, and acceleration of the landing platform in the inertial reference frame; and ΔT is the time step. We assume that the angular velocity and the acceleration of the landing platform remain constant for a short period of time, ΔT .

For the optimal localization of the landing platform, we have formulated an EKF as follows

$$\mathbf{x}_{k+1}^t = \mathbf{f}(\mathbf{x}_k^t) + \mathbf{w}_k^t \quad (2a)$$

$$\mathbf{z}_k^t = \mathbf{h}(\mathbf{x}_k^t) + \mathbf{v}_k^t \quad (2b)$$

where \mathbf{x}^t is the state vector defined by $\mathbf{x}^t = [p^{t,n} \ p^{t,e} \ v^t \ \theta^t]^\top$; \mathbf{z}^t is the observation vector; \mathbf{f} is the state-transition model, representing the kinematics of the landing platform in the discrete time domain; \mathbf{h} is the observation model; $\mathbf{w}_k^t \sim \mathcal{N}(0, Q^t)$ is a zero mean Gaussian random vector with covariance Q^t , representing the uncertainties in the system model; and $\mathbf{v}^t \sim \mathcal{N}(0, R^t)$ is a zero mean Gaussian random vector with covariance R^t , representing measurement noise. According to (1), the state-transition model is given by

$$\mathbf{f}(\mathbf{x}_k^t) = \begin{bmatrix} p_k^{t,n} + \Delta T v_k^t \cos \theta_k^t \\ p_k^{t,e} + \Delta T v_k^t \sin \theta_k^t \\ v_k^t \\ \theta_k^t \end{bmatrix} \quad (3)$$

and the observation model, assuming that only position measurements are available, is given by

$$\mathbf{h}(\mathbf{x}_k^t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \mathbf{x}_k^t \quad (4)$$

The acceleration a_k^t and angular velocity ω_k^t in (1) are unknown in target tracking, and therefore we consider them as unmodeled system uncertainty (or external disturbances). Then, the disturbance in the EKF equations in (2a) is given by

$$\mathbf{w}_k^t = \begin{bmatrix} \frac{1}{2} \Delta T^2 \bar{a} \cos \theta_k^t \\ \frac{1}{2} \Delta T^2 \bar{a} \sin \theta_k^t \\ \Delta T \bar{a} \\ \Delta T \bar{\omega} \end{bmatrix} \quad (5)$$

where \bar{a} and $\bar{\omega}$ are uncorrelated zero mean Gaussian random variables with variance of σ_a^2 and σ_ω^2 , respectively. Consequently, the system uncertainty covariance matrix $Q_k^t \in \mathbb{R}^{4 \times 4}$ can be formulated by $Q_k^t = \mathbb{E}[\mathbf{w}_k^t (\mathbf{w}_k^t)^\top]$, where $\mathbb{E}[\cdot]$

denotes the expectation operator. Finally, the measurement uncertainty covariance matrix, $R^t \in \mathbb{R}^{2 \times 2}$ can be given by

$$R^t = \begin{bmatrix} (\rho^n)^2 & 0 \\ 0 & (\rho^e)^2 \end{bmatrix} \quad (6)$$

where $(\rho^n)^2$ and $(\rho^e)^2$ are variances of position measurement error of the landing platform in the north and east direction, respectively.

B. Dynamic Model

The UAV dynamics are governed by the Newton-Euler equations

$$m\mathbf{a} = \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} + R_B^W \begin{bmatrix} 0 \\ 0 \\ -T \end{bmatrix} + \mathbf{f}_D \quad (7)$$

where m , \mathbf{a} , and g denote the UAV mass, the UAV acceleration, and the gravitational acceleration, respectively. Furthermore, R_B^W is the rotation matrix from the north-east-down (NED) reference frame to Body Frame, T is the generated total thrust due to the UAV rotors, and \mathbf{f}_D is the air damping force [14].

We denote the roll, pitch, and yaw angles of the UAV by ϕ , θ and ψ , respectively. Therefore, in Equation (7), the rotation matrix R_B^W is given by

$$R_B^W = \begin{bmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi - c_\phi c_\psi & c_\phi s_\theta s_\psi + s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{bmatrix} \quad (8)$$

where the shorthand notation $c_x \equiv \cos x$ and $s_x \equiv \sin x$ is used. Using the roll-pitch-yaw attitude parameterization, Equation (7) can be written as

$$\begin{aligned} m \begin{bmatrix} \ddot{p}^n \\ \ddot{p}^e \\ \ddot{p}^d \end{bmatrix} &= \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} - \begin{bmatrix} c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\phi s_\theta s_\psi + s_\phi c_\psi \\ c_\phi c_\theta \end{bmatrix} T \\ &\quad - k_d \begin{bmatrix} \dot{p}^n |\dot{p}^n| \\ \dot{p}^e |\dot{p}^e| \\ \dot{p}^d |\dot{p}^d| \end{bmatrix} \end{aligned} \quad (9)$$

where p^n , p^e , and p^d denote the position in north, east, and down, respectively; their first and second derivatives are, respectively, velocities and accelerations. In this paper, we have modeled the air damping force to be proportional to the signed squared velocity with friction coefficient k_d . Finally, the inputs acting on the UAV are given by thrust (T), roll (ϕ), pitch (θ), and yaw (ψ) in the model given by (9).

To achieve (nearly) constant flight altitude, i.e., $\dot{p}^d \approx 0$ and $\ddot{p}^d \approx 0$, a proportional-derivative (PD) controller is used to regulate the longitudinal motion of the UAV in the pitch plane. By applying a thrust force given by $T = mg/c_\phi c_\theta$ for hovering at constant altitude and assuming $\psi = 0$, Equation (9) is simplified to

$$m \begin{bmatrix} \ddot{p}^n \\ \ddot{p}^e \end{bmatrix} = mg \begin{bmatrix} -\tan \theta \\ \tan \phi / \cos \theta \end{bmatrix} - k_d \begin{bmatrix} \dot{p}^n |\dot{p}^n| \\ \dot{p}^e |\dot{p}^e| \end{bmatrix} \quad (10)$$

Under the assumption of non-aggressive maneuvers, the states of the UAV in this work remain near the equilibrium

point $(\phi = 0, \theta = 0)$. Linearizing the dynamics given by (10) about this equilibrium point, we obtain

$$m \begin{bmatrix} \ddot{p}^n \\ \ddot{p}^e \end{bmatrix} = mg \begin{bmatrix} -\theta \\ \phi \end{bmatrix} - k_d \begin{bmatrix} \dot{p}^n \\ \dot{p}^e \end{bmatrix} \quad (11)$$

Hence, we have

$$\begin{bmatrix} \ddot{p}^n \\ \ddot{p}^e \end{bmatrix} = -k_d/m \begin{bmatrix} \dot{p}^n \\ \dot{p}^e \end{bmatrix} + g \begin{bmatrix} -\theta \\ \phi \end{bmatrix} \quad (12)$$

Therefore, the state space representation of the linearized dynamics of the UAV can be written as

$$\dot{\mathbf{x}}^m = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -k_d/m & 0 \\ 0 & 0 & 0 & -k_d/m \end{bmatrix} \mathbf{x}^m + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ -g & 0 \\ 0 & g \end{bmatrix} \mathbf{u} \quad (13)$$

where $\mathbf{u} = [\theta \ \phi]^\top$ is the input to the system; and $\mathbf{x}^m = [p^{m,n} \ p^{m,e} \ \dot{p}^{m,n} \ \dot{p}^{m,e}]^\top$ represents the state vector of the UAV dynamics.

C. Model Predictive Control of UAVs

During non-aggressive maneuvers such as hovering or step reference tracking, linear and nonlinear MPC control schemes result in more or less the same performance for rotary wing MAVs that operate under the influence of disturbances (see the in-depth investigation in [43]). On the other hand, during aggressive maneuvers, nonlinear MPC schemes outperform their linear counterparts throughout the whole operational envelope of MAVs. Furthermore, as demonstrated in [44], under certain conditions, nonlinear MPC becomes equivalent to linear MPC. Specifically, when sequential quadratic programming (SQP) is used to obtain the NMPC-based control policies, the first step of a full-step Gauss-Newton SQP results in a linear MPC control solution (see Lemma 3.1 in [44]). In this paper, we consider the case where the mobile platform is moving on a straight line and the center of mass of the drone is moving on an approximate straight line, while the orientation of the drone remains almost constant. Consequently, a linear MPC control scheme can be effectively used for controlling the motion of the drone.

To design a linear MPC scheme for the UAV, whose dynamics are given by (13), we discretize the dynamical system with $\Delta T = 0.025$ s. In the rest of the paper, unless otherwise stated, \mathbf{x} refers to the discrete time state $\mathbf{x}_k = [p_k^{m,n} \ p_k^{m,e} \ \dot{p}_k^{m,n} \ \dot{p}_k^{m,e}]^\top$. Subsequently, the predicted state with M calculated future control inputs are given by

$$\begin{aligned} \hat{\mathbf{x}}_{k+P} &= A^P \mathbf{x}_k + A^{P-1} B \hat{\mathbf{u}}_k + \dots + B \hat{\mathbf{u}}_{k+M-1} \\ &= A^P \mathbf{x}_k + A^{P-M} \sum_{i=1}^M A^{i-1} B \hat{\mathbf{u}}_{k+M-i} \end{aligned} \quad (14)$$

where $\hat{\mathbf{x}}_{k+P}$ is the P -step prediction of the state at time k with $P > M$. Hence, the augmented state predictions are

$$\hat{\mathcal{X}} = \mathcal{S}^x \mathbf{x}_k + \mathcal{S}^u \hat{\mathbf{u}}_M \quad (15)$$

where

$$\begin{aligned}\hat{\mathcal{X}} &= [\hat{\mathbf{x}}_{k+1} \ \cdots \ \hat{\mathbf{x}}_{k+P}]^\top \\ \mathcal{S}^x &= [A \ A^2 \ \cdots \ A^P]^\top \\ \mathcal{S}^u &= \begin{bmatrix} B & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{M-1}B & A^{M-2}B & \cdots & B \\ A^M B & A^{M-1}B & \cdots & AB \\ \vdots & \vdots & \ddots & \vdots \\ A^{P-1}B & A^{P-2}B & \cdots & A^{P-M}B \end{bmatrix} \\ \hat{\mathbf{u}}_M &= [\hat{\mathbf{u}}_k \ \hat{\mathbf{u}}_{k+1} \ \cdots \ \hat{\mathbf{u}}_{k+M-1}]^\top\end{aligned}$$

The quadratic cost function used in this work is formulated as follow:

$$J_k = \{q\tilde{\mathbf{x}}^\top Q\tilde{\mathbf{x}} + (1-q)\hat{\mathbf{u}}_M^\top R\hat{\mathbf{u}}_M\} \quad (16a)$$

$$\tilde{\mathbf{x}} = \mathbf{r}_P - \hat{\mathcal{X}} \quad (16b)$$

where \mathbf{r}_P and $\tilde{\mathbf{x}}$ are the reference state of the desired trajectory and the state error, respectively; Q and R represent the state and input weight matrices, respectively; and q is the weighting factor between the state cost and the input energy cost.

The future M system inputs can be calculated as the optimal solution to the following constrained optimization problem, where the batch approach and quadratic programming are used to solve it [35].

$$\begin{aligned}J_k^* &= \min J_k = \min \{q\tilde{\mathbf{x}}^\top Q\tilde{\mathbf{x}} + (1-q)\hat{\mathbf{u}}_M^\top R\hat{\mathbf{u}}_M\} \\ \text{subj. to } &\begin{bmatrix} -v_{\max} \\ \vdots \\ -v_{\max} \end{bmatrix} \leqslant C_v \hat{\mathcal{X}} \leqslant \begin{bmatrix} v_{\max} \\ \vdots \\ v_{\max} \end{bmatrix} \\ &\begin{bmatrix} -u_{\max} \\ \vdots \\ -u_{\max} \end{bmatrix} \leqslant \hat{\mathbf{u}}_M \leqslant \begin{bmatrix} u_{\max} \\ \vdots \\ u_{\max} \end{bmatrix}\end{aligned} \quad (17)$$

where v_{\max} is the maximum UAV speed; and u_{\max} is the maximum tilt angle (roll and pitch) of the UAV. In this study, we have used $v_{\max} = 18$ m/s and $u_{\max} = 35^\circ$. C_v selects all velocity estimates from $\hat{\mathcal{X}}$, that is

$$C_v = \begin{bmatrix} 0 & 0 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 1 \end{bmatrix} \quad (18)$$

Substituting $\tilde{\mathbf{x}}$ and $\hat{\mathcal{X}}$ in (16b) and (15) into (17) yields

$$\begin{aligned}J_k^* &= \min \{\hat{\mathbf{u}}_M^\top (\mathcal{S}^u)^\top qQ\mathcal{S}^u + (1-q)R\hat{\mathbf{u}}_M \\ &\quad + 2(\mathbf{x}_k^\top \mathcal{S}^x)^\top qQ\mathcal{S}^u - \mathbf{r}_P^\top qQ\mathcal{S}^u\}\hat{\mathbf{u}}_M \\ \text{subj. to } &\begin{bmatrix} I \\ -I \end{bmatrix} \hat{\mathbf{u}}_M \leqslant \begin{bmatrix} u_{\max} \\ \vdots \\ u_{\max} \end{bmatrix} \\ \begin{bmatrix} I \\ -I \end{bmatrix} C_v \mathcal{S}^u \hat{\mathbf{u}}_M &\leqslant \begin{bmatrix} v_{\max} \\ \vdots \\ v_{\max} \end{bmatrix} + \begin{bmatrix} -I \\ I \end{bmatrix} C_v \mathcal{S}^x \mathbf{x}_k\end{aligned} \quad (19)$$

D. Target tracking based on gimbal

With the position of the target (or AprilTag) with respect to the camera position, we can easily obtain the desired gimbal angle for the camera to point at the target. In other words, to minimize the probability of target loss, we would like to have the target at the center of the image plane, which is captured by the camera. The desired gimbal angle can then be computed to be

$${}^G\phi_{des}^t = {}^G\phi^t + \tan^{-1}({}^Gy^t / {}^Gz^t) \quad (20a)$$

$${}^G\psi_{des}^t = {}^G\psi^t + \tan^{-1}({}^Gx^t / {}^Gz^t) \quad (20b)$$

where ${}^G\phi_{des}^t$ and ${}^G\psi_{des}^t$ are the desired pitch and yaw angle of the gimbal in Gimbal Frame, , respectively; ${}^G\phi^t$ and ${}^G\psi^t$ are, respectively, the current pitch and yaw angle of the gimbal; ${}^Gx^t$ and ${}^Gy^t$ are, respectively, the lateral and longitudinal position of the target on the image plane coordinates; and ${}^Gz^t$ is the distance between the target and the camera. Since the roll of the gimbal is limited to $\pm 15^\circ$, it is not employed for target tracking in this work.

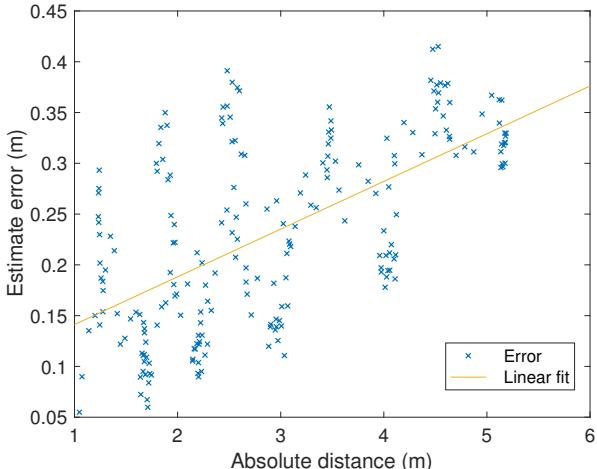
IV. EXPERIMENTS

A. Experimental Setup

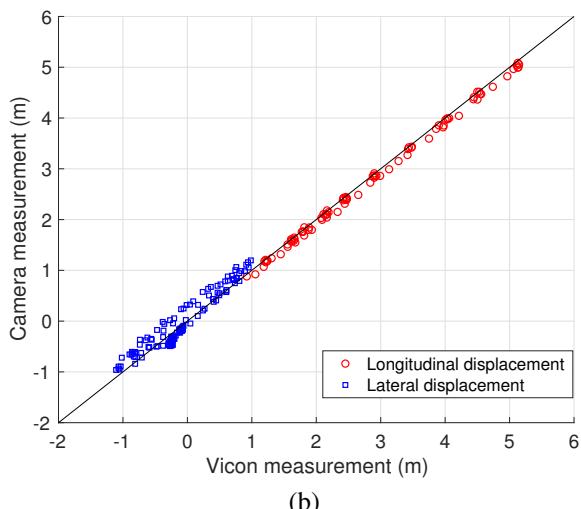
To validate the algorithms, we conducted outdoor flight experiments in two different days with the average wind speed of 5 mph and 12.9 mph.

B. AprilTag detection performance

The AprilTag was used to estimate the landing pad's pose relative to the UAV in six degrees of freedom. The pose estimate was then used in an extended Kalman filter to obtain a more reliable estimate to be used to control the UAV's approach. To understand the detection performance, we experimented with a marker that is 21.5 cm \times 21.5 cm in size, where it was held in front of the camera and moved away, increasing the distance, in steps. At each step, the marker was moved horizontally and vertically. To obtain a ground truth, we used a real-time motion capture system (Vicon) based a camera array and infra-red markers. Fig. 6 (a) shows the error in each measurements as a function of the true distance. The mean and standard deviation of the error for data collected between 1 and 5.5 m was 23.2 ± 9.1 cm. We observe that the error increases as a function of distance which is expected with the reduced resolution of the



(a)



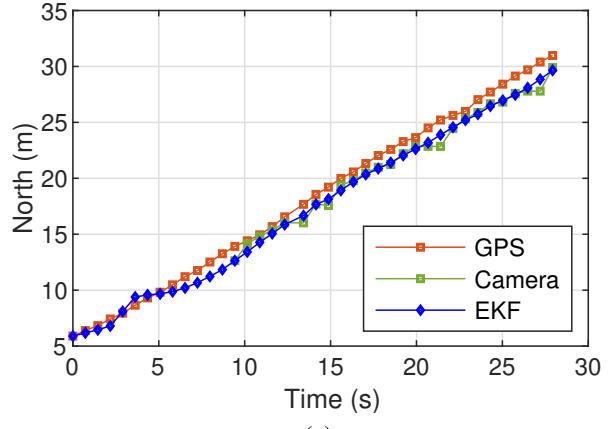
(b)

Fig. 6. AprilTag detection performance. (a) Estimation error of distance to AprilTag marker using the on-board camera. Error is below 40 cm for distances up to 5 m. The larger deviations from the linear fit associated with movements of the marker away from the center of the camera's field of view. (b) Visualization of camera estimates of the AprilTag's position compared to the ground truth. Longitudinal movement, along the camera's optical axis, are measured fairly accurately, whereas lateral movement away from the center of the field of view are less reliable.

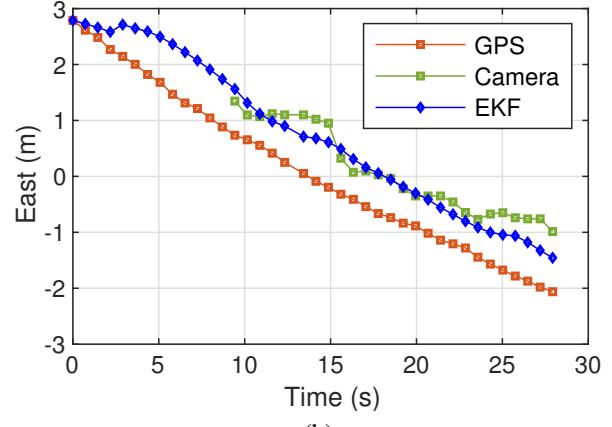
marker. The larger deviations coincide with lateral movement away from the center of the field of view. We suspect lens spherical distortion and imperfect camera calibration play a role in this. Fig. 6 (b) shows the longitudinal (along the camera's optical axis) and lateral displacements (away from center of field of view) separately, which shows that the range measurements are more accurate than lateral movements. For our intended purpose, the data shows that at close range (within 1.5 meters) as the UAV approaches the landing pad an error around 15 cm can be expected, which is sufficiently small to control and land the UAV.

C. Localization of flight experiments

Fig. 7 shows a typical flight, the estimation of extended Kalman filter is robust. At the beginning of the experiments,



(a)



(b)

Fig. 7. The estimated path through extended Kalman filter. (a) Position in the north; (b) Position in the east. The AprilTag is undetectable until $t = 9$ sec since the UAV is still flying toward the target.

the AprilTag is not able to be detected continuously and stably until 9 s and only GPS signals contribute the position estimations of EKF. After the AprilTag shows in the image plane of our camera, sensor fusion starts. Since the AprilTag has been kept 1.5 m distance to the GPS transmitter, so a curve occurred in the path after the detection of AprilTag. Notice that since the AprilTag estimate has less noise, the estimated path approaches to the AprilTag path. The UAV gets too close to the marker and lose detection of it at 28 s.

D. Experiment results

Shown in Fig. 8 (a) are the flight path of the quadcopter and the trajectory of mobile landing platform for autonomous landing. The quadcopter was following the landing platform until it reached a radial distance of 1 meter in the cylindrical coordinates from the target. Once the distance between the quadcopter and the platform was close enough to initiate landing state, which is shown in gray box, the quadcopter started descending for landing. Fig. 8 shows that the quadcopter landed on top on the landing platform around -2 m in Easting and 32 m in Northing. The outliers indicate false positives of target locations and noisy measurements of UAV positions.

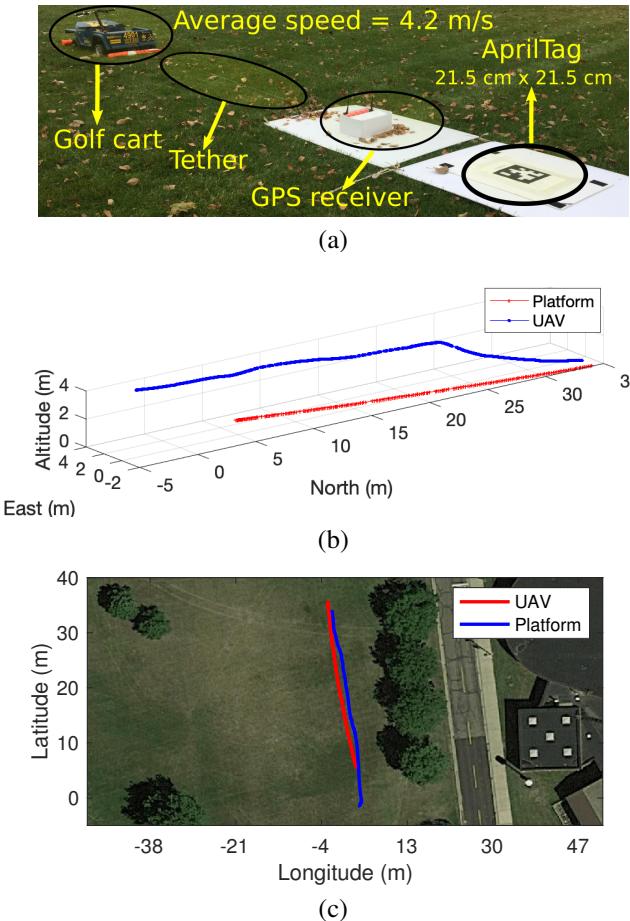


Fig. 8. The UAV autonomously landing on a target moving on a straight path. (a) Golf cart carrying the GPS receiver and the base with AprilTag; (b) 3D plot of the trajectories of the AprilTag and the UAV; (c) Satellite plot of the trajectories of the base with AprilTag and the UAV.

We demonstrate our methods in flight experiments, with various speed of target on grass. In Fig. 8, the UAV started at 8 m away from the target with no visual measurements, then detected the marker and switched to the marker, tracked and landed on the platform after the platform started moving.

As shown in Fig. 10 (a)-(c), the UAV converged with the platform in 5 s and then landed the UAV in 23 s. The average of final landing error for 6 experiments is 39 cm (see Figure 9). Fig. 10 (b) shows the position of the UAV and platform in the East. Notice how the drone followed the path in the east and keep about 1 m away until 14 s slot and entered the landing state (horizontal distance less than about 1 m). Fig. 10 (e) and (f) show the velocity of UAV during approaching and converging to the platform. Fig. 10 (g) and (h) demonstrate how the control inputs changed during approaching and converging. Because of the presence of constraints, the control inputs are limited to 20° to ensure the stability.

V. CONCLUSION

In this work, we proposed a new autonomous UAV landing system on moving platforms in the presence of uncertainties and disturbance. Our system integrates the functionality of

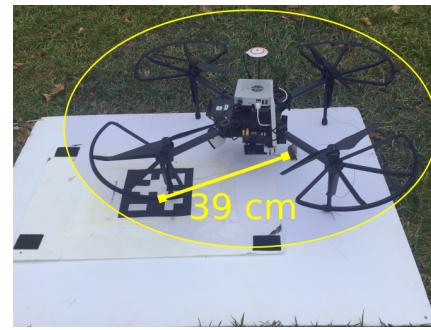


Fig. 9. Drone landed on the base with AprilTag.

a vision-based target detection method, an EKF to estimate the target position, and a linear MPC-based control scheme that controls the motion of the UAV. The flight experiments demonstrate the capability of our proposed method for autonomous landing of low-cost and lightweight UAVs on mobile platforms, in the presence of disturbances and uncertainties, with a notable precision (an average error of only 39 cm).

In the future, we aim to add an outer control loop to increase the robustness of the closed-loop control system with respect to environmental uncertainties and wind disturbance. Furthermore, we aim to investigate the efficiency of our proposed method of autonomous landing of multiple UAVs on multiple moving platforms. Finally, we aim to implement an obstacle avoidance system to guarantee safety of UAV landing operation in environments that are densely covered by obstacles such as trees and where moving obstacles such as flying birds might be present.

REFERENCES

- [1] A. Al-Kaff, D. Martín, F. García, A. de la Escalera, and J. María Armengol, "Survey of computer vision algorithms and applications for unmanned aerial vehicles," *Expert Systems with Applications*, vol. 92, pp. 447–463, 2018.
- [2] A. Cherubini, A. Papini, R. Vertechy, and M. Fontana, "Airborne wind energy systems: A review of the technologies," *Renewable and Sustainable Energy Reviews*, vol. 51, pp. 1461–1476, 2015.
- [3] K. R. B. Sri, P. Aneesh, K. Bhanu, and M. Natarajan, "Design analysis of solar-powered unmanned aerial vehicle," *Journal of Aerospace Technology and Management*, vol. 8, no. 4, pp. 397–407, 2016.
- [4] A. Williams, "Persistent mobile aerial surveillance platform using intelligent battery health management and drone swapping," in *2018 4th IEEE International Conference on Control, Automation and Robotics (ICCAR)*, Auckland, New Zealand, 2018, pp. 237–246.
- [5] E. Garone, J.-F. Determe, and R. Naldi, "Generalized traveling salesman problem for carrier-vehicle systems," *Journal of Guidance, Control, and Dynamics*, vol. 37, no. 3, pp. 766–774, 2014.
- [6] K. E. Wenzel, A. Masselli, and A. Zell, "Automatic take off, tracking and landing of a miniature UAV on a moving carrier vehicle," *J Intell Robot Syst*, vol. 61, no. 221, pp. 1–7, 2011.
- [7] T. K. Venugopalan, T. Taher, and G. Barbastathis, "Autonomous landing of an unmanned aerial vehicle on an autonomous marine vehicle," in *2012 OCEANS*, Hampton Roads, VA, 2012, pp. 1–9.
- [8] T. Muskardin, G. Balmer, L. Persson, S. Wlach, M. Laiacker, A. Ollero, and K. Kondak, "A Novel Landing System to Increase Payload Capacity and Operational Availability of High Altitude Long Endurance UAV," *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 495–504, 2016.

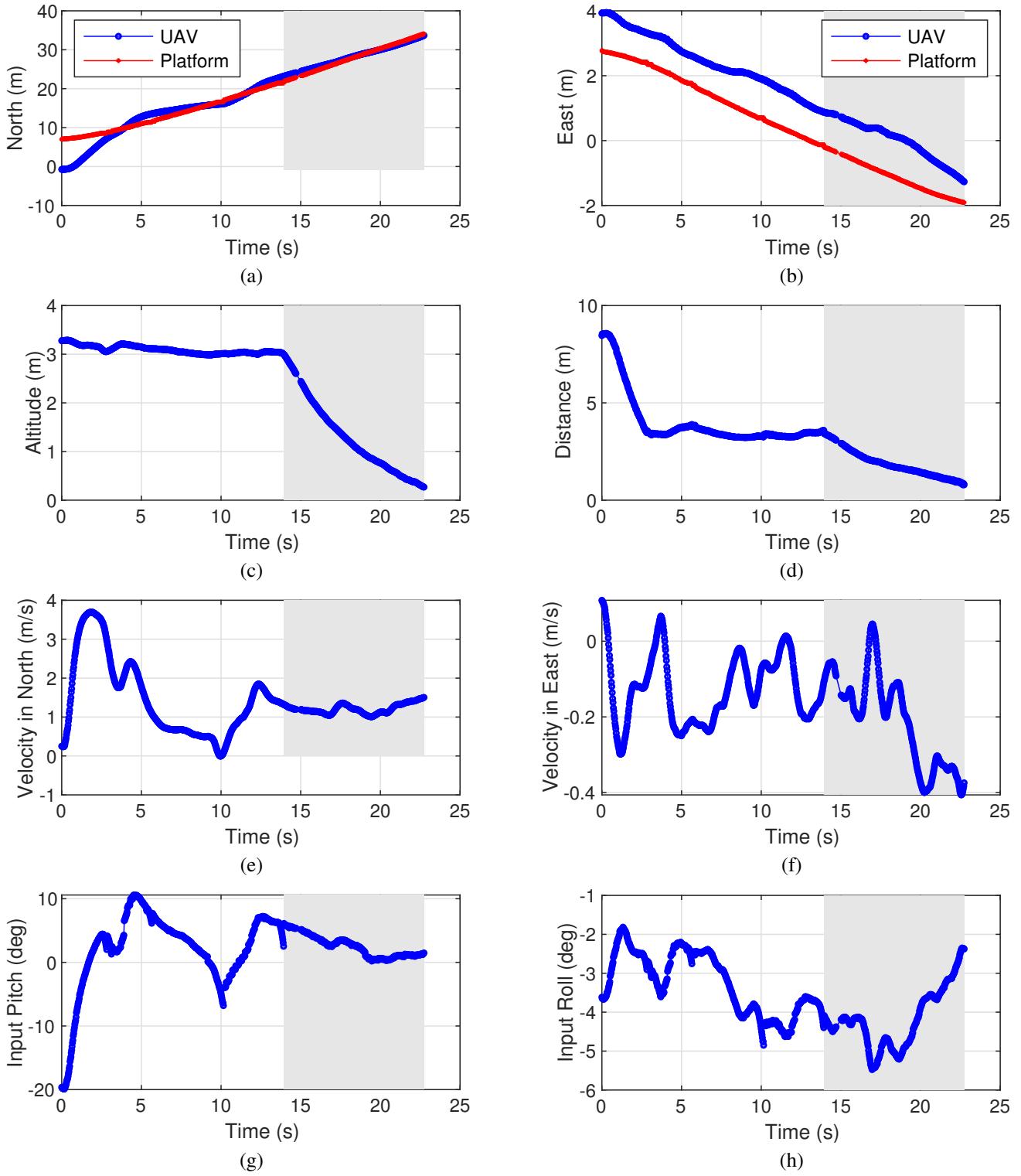


Fig. 10. The UAV autonomously landing on a target moving on a straight path in detail. (a) position in the north; (b) position in the east; (c) height; (d) distance to the target; (e) velocity in the north; (f) velocity in the east; (g) control input Pitch; (h) control input Roll. The gray regions indicate the descending stage of the UAV.

- [9] L. Coutard and F. Chaumette, "Visual detection and 3D model-based tracking for landing on an aircraft carrier," in *IEEE International Conference on Robotics and Automation*, Shanghai, China, 2011, pp. 1746–1751.
- [10] L. Coutard, F. Chaumette, and J. M. Pflimlin, "Automatic landing on aircraft carrier by visual servoing," in *IEEE International Conference*

on Intelligent Robots and Systems, San Francisco, CA, USA, 2011, pp. 2843–2848.

- [11] O. A. Yakimenko, I. I. Kaminer, W. J. Lentz, and P. A. Ghyzel, "Unmanned aircraft navigation for shipboard landing using infrared vision," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, no. 4, pp. 1181–1200, 2002.

- [12] B. Herissé, T. Hamel, R. Mahony, and F. X. Russo, "Landing a VTOL unmanned aerial vehicle on a moving platform using optical flow," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 77–89, 2012.
- [13] D. Lee, T. Ryan, and H. J. Kim, "Autonomous landing of a VTOL UAV on a moving platform using image-based visual servoing," in *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, Saint Paul, Minnesota, 2012, pp. 971–976.
- [14] P. Serra, R. Cunha, T. Hamel, D. Cabecinhas, and C. Silvestre, "Landing of a quadrotor on a moving target using dynamic image-based visual servo control," *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1524–1535, 2016.
- [15] C. Brommer, D. Malyuta, D. Hentzen, and R. Brockers, "Long-duration autonomy for small rotorcraft UAS including recharging," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2018, pp. 7252–7258.
- [16] E. Olson, "AprilTag: A robust and flexible visual fiducial system," in *Proceedings - IEEE International Conference on Robotics and Automation*, Shanghai, China, 2011, pp. 3400–3407.
- [17] M. Beul, S. Houben, M. Nieuwenhuisen, and S. Behnke, "Fast autonomous landing on a moving target at MBZIRC," in *2017 European Conference on Mobile Robots (ECMR)*, Paris, France, 2017, pp. 277–286.
- [18] B. Ludington, E. Johnson, and G. Vachtsevanos, "Augmenting UAV autonomy," *IEEE Robotics and Automation Magazine*, vol. 13, no. 3, pp. 63–71, 2006.
- [19] S. Grzonka, G. Grisetti, and W. Burgard, "A fully autonomous indoor quadrotor," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 90–100, 2012.
- [20] J. Moore and R. Tedrake, "Powerline perching with a fixed-wing UAV," in *AIAA Infotech@Aerospace Conference*, Seattle, Washington, April 2009.
- [21] T. Baca, P. Stepan, V. Spurny, D. Hert, R. Penicka, M. Saska, J. Thomas, G. Loianno, and V. Kumar, "Autonomous landing on a moving vehicle with an unmanned aerial vehicle," *Journal of Field Robotics*, vol. 36, no. 5, pp. 874–891, 2019.
- [22] D. Tzoumanikas, W. Li, M. Grimm, K. Zhang, M. Kovac, and S. Leutenegger, "Fully autonomous micro air vehicle flight and landing on a moving target using visual-inertial estimation and model-predictive control," *Journal of Field Robotics*, vol. 36, no. 1, pp. 49–77, 2019.
- [23] B. Erginer and E. Altug, "Modeling and PD control of a quadrotor VTOL vehicle," in *2007 IEEE Intelligent Vehicles Symposium*, Istanbul, Turkey, 2007, pp. 894–899.
- [24] H. Voos and B. Nourghassemi, "Nonlinear control of stabilized flight and landing for quadrotor UAVs," in *7th International Workshop on Advanced Motion Control. Proceedings*, Zielona Gora, Poland, 2009, pp. 1–6.
- [25] A. Borowczyk, D.-T. Nguyen, A. P.-V. Nguyen, D. Q. Nguyen, D. Saussié, and J. L. Ny, "Autonomous landing of a multirotor micro air vehicle on a high velocity ground vehicle," *Journal of Guidance, Control, and Dynamics*, vol. 40, no. 9, pp. 2373–2380, 2016.
- [26] B. Ahmed and H. R. Pota, "Backstepping-based landing control of a RUAV using tether incorporating flapping correction dynamics," in *2008 American Control Conference*, Seattle, WA, USA, jun 2008, pp. 2728–2733.
- [27] K. Boudjitt, "Detection and target tracking with a quadrotor using fuzzy logic," in *International Conference on Modelling, Identification and Control (ICMIC)*, Algiers, Algeria, 2016, pp. 127–132.
- [28] O. Cetin, S. Kurnaz, and O. Kaynak, "Fuzzy logic based approach to design of autonomous landing system for unmanned aerial vehicles," *Journal of Intelligent Robotic Systems*, pp. 239–250, 2011.
- [29] Y.-c. Lai, K.-c. Chan, Y.-c. Liu, and F.-b. Hsiao, "Development of an automatic landing system based on adaptive fuzzy logic control for fixed-wing unmanned aerial vehicles," *Journal of Aeronautics, Astronautics and Aviation*, vol. 48, no. 3, pp. 183–194, 2016.
- [30] J.-g. Juang, L.-h. Chien, and F. Lin, "Automatic landing control system design using adaptive neural network and its hardware realization," *IEEE Systems Journal*, vol. 5, no. 2, pp. 266–277, 2011.
- [31] R. Lungu and M. Lungu, "Automatic landing system using neural networks and radio-technical subsystems," *Chinese Journal of Aeronautics*, vol. 30, no. 1, pp. 399–411, 2017.
- [32] Z. Qing, M. Zhu, and Z. Wu, "Adaptive neural network control for a quadrotor landing on a moving vehicle," in *Chinese Control and Decision Conference (CCDC)*, Shenyang, China, 2018, pp. 28–33.
- [33] S. Lee, T. Shim, S. Kim, J. Park, K. Hong, and H. Bang, "Vision-based autonomous landing of a multi-copter unmanned aerial vehicle using reinforcement learning," in *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, Dallas, TX, 2018, pp. 108–114.
- [34] T. Templeton, D. H. Shim, C. Geyer, and S. S. Sastry, "Autonomous vision-based landing and terrain mapping using an MPC-controlled unmanned rotorcraft," in *Proceedings - IEEE International Conference on Robotics and Automation*, Roma, Italy, 2007, pp. 1349–1356.
- [35] M. K. Samal, S. Anavatti, and M. Garratt, "Neural network based model predictive controller for simplified heave model of an unmanned helicopter," in *International Conference on Swarm, Evolutionary, and Memetic Computing*, Bhubaneswar, India, 2012, pp. 356–363.
- [36] J. Tian, Y. Zheng, H. Zhu, and L. Shen, "A MPC and genetic algorithm based approach for multiple UAV's cooperative search," in *International Conference on Computational and Information Science*, Xi'an, China, 2005, pp. 399–404.
- [37] Y. Wu and X. Qu, "Obstacle avoidance and path planning for carrier aircraft launching," *Chinese Journal of Aeronautics*, vol. 28, no. 3, pp. 695–703, 2015.
- [38] G. V. Raffo, M. G. Ortega, and F. R. Rubio, "Path tracking of a UAV via an underactuated control strategy," *European Journal of Control*, vol. 17, no. 2, pp. 194–213, 2011.
- [39] K. Yang, "Path following control performance comparison for an rotary wing unmanned aerial vehicle," in *2013 10th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, Jeju, South Korea, Oct 2013, pp. 313–318.
- [40] Y. Feng, C. Zhang, S. Baek, S. Rawashdeh, and A. Mohammadi, "Autonomous landing of a UAV on a moving platform using model predictive control," *Drones*, vol. 2, no. 4, p. 34, 2018.
- [41] M. Fu, "Autonomous Landing of a Quadrotor on an UGV," *2016 IEEE International Conference on Mechatronics and Automation*, pp. 988–993, 2016.
- [42] J. Janousek and P. Marcon, "Precision Landing Options in Unmanned Aerial Vehicles," *2018 International Interdisciplinary PhD Workshop (IIPhDW)*, pp. 58–60, 2018.
- [43] M. Kamel, M. Burri, and R. Siegwart, "Linear vs nonlinear MPC for trajectory tracking applied to rotary wing micro aerial vehicles," in *IFAC*, vol. 50, no. 1, Toulouse, France, 2017, pp. 3463–3469.
- [44] S. Gros, M. Zanon, R. Quirynen, A. Bemporad, and M. Diehl, "From linear to nonlinear MPC: bridging the gap via the real-time iteration," in *International Journal of Control*, vol. 7179, 2016, pp. 1–19.