12th IEEE International Conference on Control & Automation (ICCA)
Kathmandu, Nepal, June 1-3, 2016

System Integration of a Vision-Guided UAV for Autonomous Landing on Moving Platform

Xudong Chen[1], Swee King Phang[1], Mo Shan[1], and Ben M. Chen[2]

Abstract— **This manuscript describes a UAV implemented with vision and laser based localization algorithm to track and land on a moving platform. Specifically, a pre-designed marker is installed on the moving platform and a downward facing monocular camera is mounted on the UAV for pose estimation. For a robust and precise UAV height estimation, a LiDAR scanning range finder is utilized to determine accurate height and to estimate descent velocity during landing. The proposed algorithm is realized in a quadrotor UAV and its performance is validated in actual flight experiments.**

I. Introduction

In recent years, unmanned aerial vehicles (UAVs) are widely used in many practical applications in both military and civilian fields, such as surveillance, exploration, agriculture, search and rescue and border patrol. The evolution of advanced robotic technologies has allowed UAV become more autonomous in many new fields of applications. A comprehensive survey focusing on control, guidance and navigation of UAVs can be studied in [1].

Automation of landing as the one of the most critical phase of UAV fight, is a challenging problem especially onto a moving platform. Under sub-optimal weather conditions, even for manned aircraft, landing is known as one of the risky operation to pilots, as it is required to approach landing sites with accurate attitude and velocity [2].

To aid the landing process, global positioning system (GPS), LiDAR and radar sensors have been widely used for relative pose estimation between the aircraft and the ground (or landing platform). Aircrafts adopting GPS information for precise guidance and landing are presented in [3], [4], [5], [6]. Besides using GPS signals, a common automatic recovery system uses millimeter-wave radar on the platform for pose estimation with a transponder to send data onto the UAV. Such systems usually require extra devices on the landing sites or platforms, and data link between UAV and landing platform needs to be established and maintained during flight. In [7], a rotorcraft with visual tracking and LiDAR sensors is presented. It is capable to provide both pose and orientation information of the ship deck.

Vision aided autonomous aerial systems are recently becoming popular due to rich information provided by the images. To-date, many monocular camera and stereo vision based algorithms are developed for autonomous landing of UAVs [8], [9], [10], [11]. These algorithms usually require pre-known marker to be attached on the landing sites. Many works has been documented in [12], [13], [14], where the UAV is capable to hover and land on a moving platform by using optical flow algorithm with a monocular camera. In [15], a monocular camera based landing system with unknown dimensions of the landing pad is developed. Of all these implemented marker-based autonomous landing algorithm, the most challenging part is the difficulty of acquiring accurate height measurement while camera loses visual sight, or being too far to the marker. To achieve smooth touchdown, it is crucial to keep the descent speed to a constant value in a rate-based controller.

In this manuscript, motivated by the above-mentioned constrains, an autonomous UAV landing on moving platform solution is proposed by utilizing both vision and LiDAR sensors. A monocular vision camera is primarily used to estimate the horizontal position of the target, while the LiDAR sensor can accurately extract the height information of the UAV. On top of that, it can be used to detect and filter outliers introduced by the vision algorithm effectively. This paper is divided into the following sections: Section II will briefly mention the hardware and sensors used in this implementation. Section III focuses on the vision algorithm to estimate horizontal position of the marker, while section IV introduce the height estimation algorithm to robustly detect the actual ground. Flight controller will be introduced in Section V follows by actual flight experiment results in Section VI. Lastly, conclusion remarks will be made.

II. Hardware Platform

A self-customized quadcopter UAV codenamed T-Lion is utilized as the bare platform to implement the proposed autonomous landing on moving vehicle systems. T-Lion series of UAV is developed by the Control Science Group of Temasek Laboratories at the National University of Singapore. Due to its suitable size and high payload, it is widely used in many indoor and outdoor applications.

Flight controller, powerful processors and several essential sensors are installed onto T-Lion to achieve the purpose of autonomous landing on a moving platform. They will be discussed individually in this sections.

A. Flight Controller

All T-Lion series of UAVs are equipped with an in-house flight controller. The flight controller consists of dual inertial measurement units (IMUs) with smart switching capability when anyone of them is behaved abnormally. It is

[1]Xudong Chen, Swee King Phang, Mo Shan are with Temasek Laboratories, National University of Singapore, Singapore. {tslcx,skphang,shanmo}@nus.edu.sg

[2]Ben M. Chen is with Department of Electrical and Computer Engineering, National University of Singapore, Singapore. bmchen@nus.edu.sg

978-1-5090-1738-6/16/$31.00 ©2016 IEEE

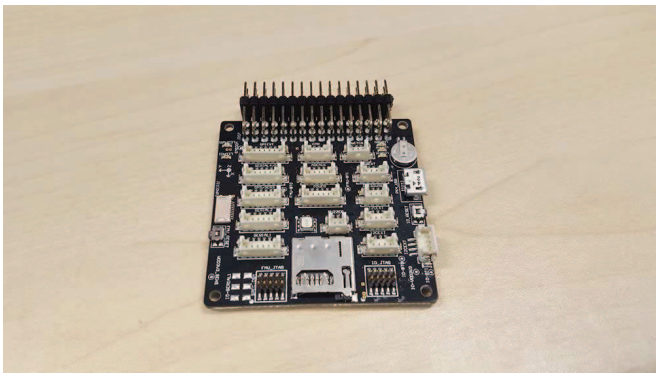

Fig. 1. In-house flight controller board

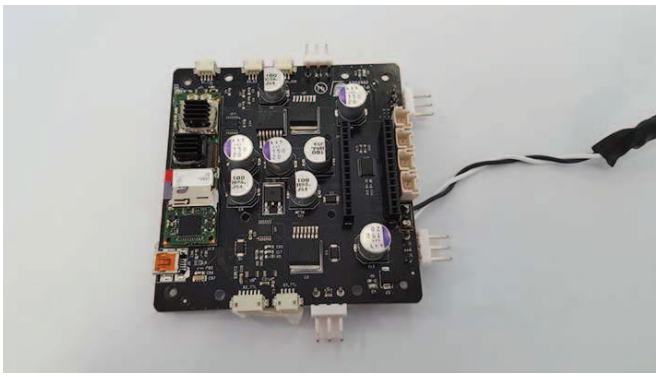

Fig. 2. Self customized circuit board LionHub

mainly used for flight orientation stabilization. It controller is also capable of providing GPS coordinate of the UAV for autonomous position and velocity control. However, GPS will not be used in this implementation as the UAV will only self-localize and track with on vision algorithm. The flight controller board can be visualized in Fig. 1.

B. Laser Range Finder

To be robust enough to take care of different landscapes and structures where the moving vehicle travels, a 30-meter range laser scanner, Hokuyo UTM-30LX, is utilized to estimate flying height of the UAV. It is mounted facing downwards and located at the back of T-Lion.

C. Gimbal Controlled Camera System

Monocular camera is employed for visual image capturing during flight and landing. In this implementation, a Point-Grey BlackFly camera is used. This camera is required to face vertically downwards at all time during flight.

In order to achieve a stabilized visual image capturing, a 2-axis gimbal system is considered. Arris Zhaoyun 2-axis Brushless Gimbal is adopted to mount and stabilize the camera. The camera is controlled in both pitching and rolling direction. To accommodate the choice of camera, minor modification is carried out on the mounting plate of the gimbal.

D. Onboard Computer

In General, image processing is computational intensive. The current on-board ARM-based processor for flight controller alone will not be able to take up the task. Hence, an Intel NUC mini-computer with i7 processor is utilized to perform vision algorithm using images captured by the on-board camera, and also to process the laser range finder data for height estimation algorithm.

All image processing procedures will be implemented and run in this vision computer. A robust height estimation algorithm will also be included.

E. Hardware Integration

The components need to be properly connected such that

1) They will be properly powered. Intel NUC mini-computer, Zhaoyun Gimbal and Hokuyo laser range finder need 12 v supply, while the rest of the components need 5 v supply.
2) The components will be connected via various communication ports, such as USB ports and serial ports without excessive wiring.

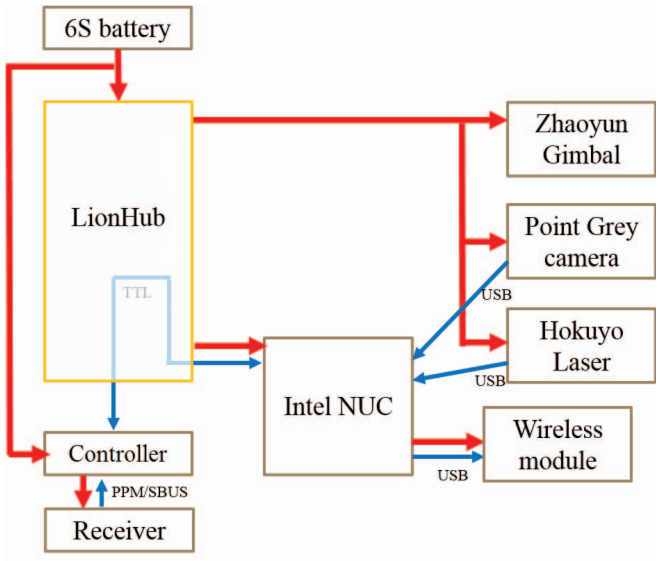

Fig. 3. Connection between components

To accommodate the sensors and hardware, a customized circuit board codenamed LionHub (see Fig. 2) together with various voltage regulators is designed. It is specially designed for T-Lion series UAV such that all avionic components used can be mounted on it directly. Fig. 3 shows the connections between the components in block diagram. All the peripheral components connected to Intel NUC via USB ports, while the data from the processor unit is sent to flight mission controller via TTL serial port. An additional GPS receiver is included in the system only for data comparison purpose during the stage of algorithm development.

III. Vision-based X-Y Position Estimation

In this section, the onboard vision system, which is responsible for providing the relative pose between the UAV and the landing platform, will be investigated. To facilitate the identification of the platform, a customized marker is

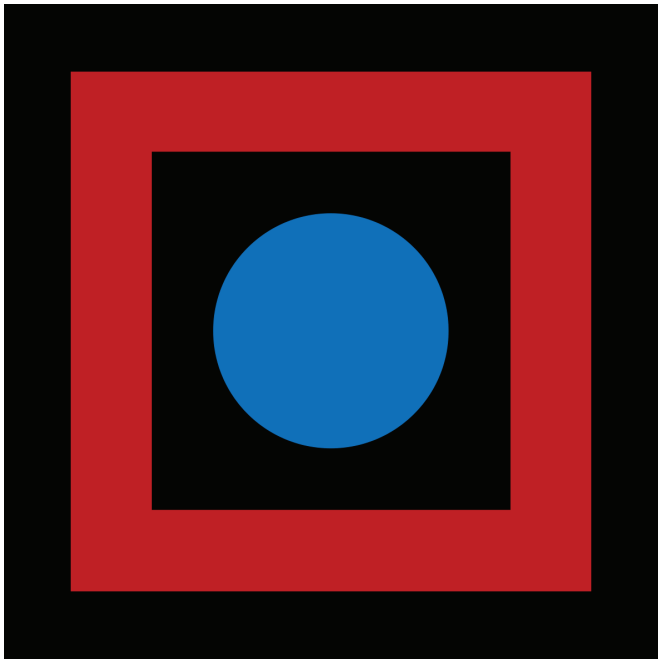

Fig. 4. Marker placed on moving platform

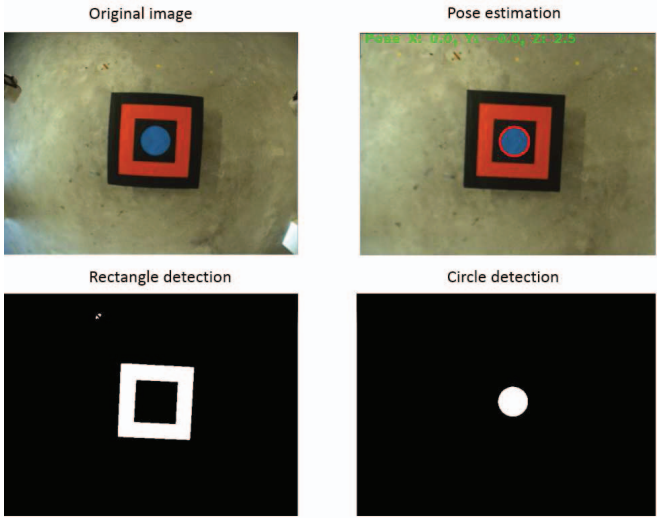

Fig. 5. Results of onboard vision algorithm

designed with a red square border and a blue circle in it (Fig. 4).

The onboard image is first converted from color to grayscale, and then processed by adaptive thresholding. Contour detection is performed in the resulting binary image to identify the rectangle. Once it is found, the original colored image is cropped such that only the region inside the rectangle is kept, where circle will be detected. Using the cropped image instead of the entire image could reduce the amount of outliers for circle detection. In case that the rectangle could not be found, circle detection will be performed in the whole image.

The cropped image is then converted from RGB colorspace to HSV, and the circle is detected by thresholding the hue value. The range of hue is set to be 0 to 50, corresponding to blue hue. The outliers are pruned by setting thresholds for the circle areas. Based on the detected circle, the pose of the UAV can be estimated by the OpenCV function *solvePnP* as described in [16]. Since the pose estimation is marker based, the marker should be visible to the camera during the entire landing process. The result of the vision algorithm is shown in Fig. 5.

There are two considerations in marker design to increase the robustness of pose estimation. Firstly, the colors used for rectangle and circle are different, to maximize their separation in hue. Secondly, circle is chosen because the pose could still be estimated even if the circle is only partially seen, which happens when the UAV is close to the platform during landing. This advantage increases the precision of autonomous landing.

IV. Height Estimation

Height measurement is critical for autonomous landing on the moving target, as mentioned above. It could also be used to estimate the relative position between the UAV and the moving target in x- and y-positions. Therefore, to obtain accurate height measurement, a high-end scanning laser range finder, namely the Hokuyo laser scanner, is installed on-board of T-Lion. A simple yet robust estimation of UAV height is designed and the idea will be explained below.

For each complete scan, the Hokuyo laser scanner generates 1081 integer values, which represent the measured distances in millimeter from its starting point on the right to the ending point on the left, covering a total of 270° angle. These data can be seen as in polar coordinates since each distance is associated with its own angle direction. To convert the raw measurement date from polar coordinates (r_i, θ_i) to Cartesian coordinates (x_i, y_i), the following transformation can be applied

$$x_i = r_i \cos\theta_i, \quad (1)$$

$$y_i = r_i \sin\theta_i, \quad (2)$$

where $i \in \{1, 2, 3, ..., 1081\}$ is the index of the laser scanner measurements. This array of 2D points can be grouped into clusters of points belonging to individual line segments by the *split-and-merge* algorithm. The main steps of the *split-and-merge* algorithm is summarized here with Fig. 6 giving a graphical illustration.

1) Connect the first point A and the last point B of the input data by a straight line.
2) Find point C among all data points that has the longest perpendicular distance to line AB.
3) If this longest distance is within a threshold, then a cluster is created containing points in between A and B.
4) Else, the input points will be split into two subgroups, A-C and C-B. For each group, the *split-and-merge* algorithm will be applied recursively.

Thereafter, clusters of points could be created, and then least-square line fitting algorithm can be applied to all points

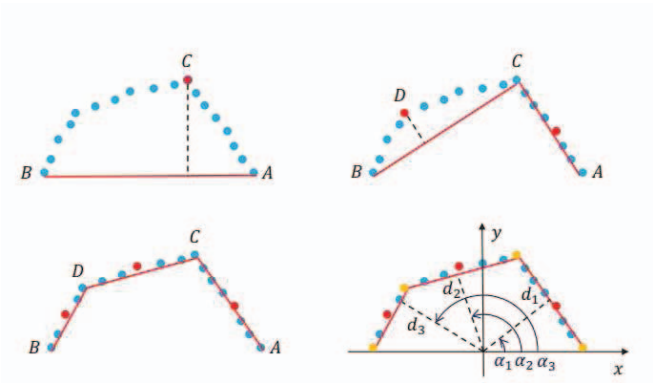

Fig. 6. Steps on *split-and-merge* algorithm

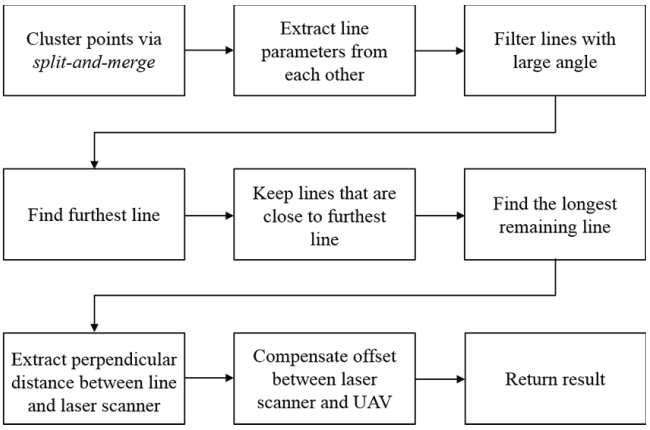

Fig. 7. Steps to compute height via laser range scanner

in each cluster to obtain the individual lines. Each line is represented by two parameters–the line's normal direction α_k and its perpendicular distance to the center of laser scanner d_k. In Fig. 6, xy-surface represents the horizontal plane of the output from the laser scanner. The normal direction of the line is defined as the angle from the x-axis to the line normal in counterclockwise direction. Next, lines with dissimilar gradient as the ground plane are filtered out. Since the obtained lines are expressed in the laser scanner frame, their directions α_k should be compensated by the UAV roll angle ϕ and then compared to the normal line of the ground plane at $\pi/2$. Hence

$$\Delta\alpha_k = \alpha_k - \phi - \pi/2. \tag{3}$$

The corresponding line is filtered out if the value of $\Delta\alpha_k$ is greater than a threshold of 5 degrees. The remaining lines are sorted by their perpendicular distances to the laser scanner and the furthest ones are kept. Among these lines, the longest one is chosen to be the true ground. At last, the perpendicular distance of this line to the laser scanner center is compensated with the UAV pitch angle θ and the offset between the laser scanner and the UAV center of gravity (CG), Δh, and thus the final height estimation to be

$$h = r\cos\theta - \Delta h. \tag{4}$$

Fig. 7 shows the flow chart of the laser scanner based height calculation algorithm. Using this method, accurate height measurement can be obtained as long as the laser scanner projects a portion of its laser beams onto the true ground. Therefore, it still works for the case when the UAV flies over protruding objects on the ground, such as the moving vehicle.

V. Flight Control

Following the usual practice in Temasek Laboratories UAV Research Group, the UAV control problem is separated into the attitude stabilization layer and the position tracking layer (see Fig. 8). The attitude stabilization layer involves the design of an inner-loop controller which ensures the UAV roll, pitch and yaw dynamics are robustly stable. Moreover, the position tracking layer involves the design of an outer-loop controller which enables the UAV to track any smooth 3-D trajectory in a responsive and precise way.

The inner-loop controller is implemented in the flight controller board, where an attitude stabilizer is implemented and tuned towards fast closed-loop dynamics with a simple software framework. As large amounts of work about quadcoptor stability control have been published in literature, the details are omitted in this manuscript.

In contrast, the design of the outer-loop controller is much more critical for this application due to the requirements on moving vehicle tracking. The relative position of the UAV to the reference marker at every instance is estimated by the aforementioned vision algorithm and the UAV needs to track the marker as closely as possible. The robust and perfect tracking (RPT) control concept from [17] perfectly fits this requirement.

The outer dynamics of the UAV in quadrotor configuration is differentially flat, similar to the case introduced in [18], which means that all its state variables and inputs can be expressed in terms of algebraic functions of flat outputs and their derivatives. A good and intuitive choice of flat outputs is

$$\sigma = [x, y, z, \psi]^{\mathrm{T}}. \tag{5}$$

It is obvious that the first three outputs, x, y, z, are totally independent. In other words, when designing its outer-loop control law and generating the position references, the UAV can be considered as a mass point with constrained velocity, acceleration and its higher derivatives in the individual axis of the 3-D global frame. Hence, a stand-alone RPT controller based on multi-layer integrator model in each axis can be designed to track the corresponding reference in that axis. To achieve a good tracking performance, it is common to include an error integral to ensure zero steady-state error.

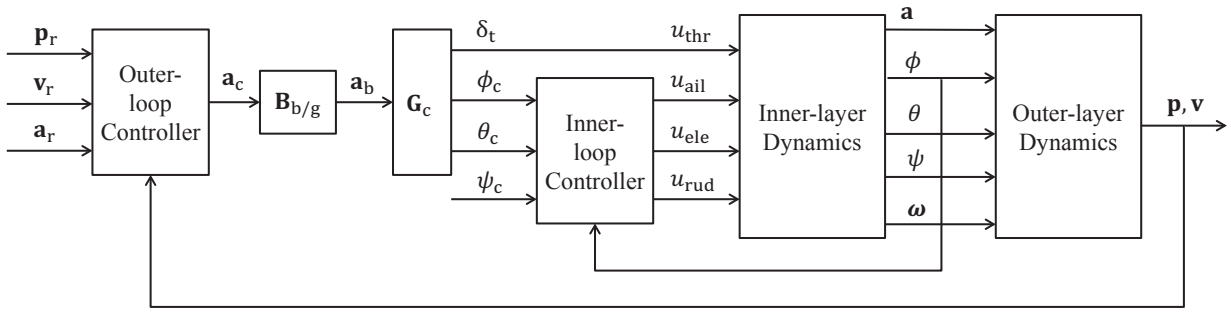

Fig. 8. Dual-loop control structure of the quadrotor

This requires an augmented system to be formulated as

$$\begin{cases} \dot{\boldsymbol{x}}_{\text{aug}} = \begin{bmatrix} 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \boldsymbol{x}_{\text{aug}} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} u_{\text{aug}} \\ \boldsymbol{y}_{\text{aug}} = \boldsymbol{x}_{\text{aug}} \\ h_{\text{aug}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \boldsymbol{x}_{\text{aug}} \end{cases}, \quad (6)$$

where $\boldsymbol{x}_{\text{aug}} = \begin{bmatrix} \int (\mathbf{p}_e) & \mathbf{p}_r & \mathbf{v}_r & \mathbf{a}_r & \mathbf{p} & \mathbf{v} \end{bmatrix}^T$, $\mathbf{p}_r$, $\mathbf{v}_r$, $\mathbf{a}_r$ are the position, velocity and acceleration references in the controlled axis, $\mathbf{p}$, $\mathbf{v}$ are the actual position and velocity and $\mathbf{p}_e = \mathbf{p}_r - \mathbf{p}$ is the tracking error of the position. Following the procedures in [19], a linear feedback control law of the following form can be acquired as:

$$u_{\text{aug}} = F_{\text{aug}} \boldsymbol{x}_{\text{aug}}, \quad (7)$$

where

$$F_{\text{aug}} = \begin{bmatrix} \dfrac{k_i \omega_n^2}{\varepsilon^3} & \dfrac{\omega_n^2 + 2\zeta\omega_n k_i}{\varepsilon^2} & \dfrac{2\zeta\omega_n + k_i}{\varepsilon} \\ & 1 & -\dfrac{\omega_n^2 + 2\zeta\omega_n k_i}{\varepsilon^2} & -\dfrac{2\zeta\omega_n + k_i}{\varepsilon} \end{bmatrix}.$$

Here, ε is a design parameter to adjust the settling time of the closed-loop system. ω_n, ζ, k_i are the parameters that determine the desired pole locations of the infinite zero structure of (6) through

$$p_i(s) = (s + k_i)(s^2 + 2\zeta\omega_n s + \omega_n^2). \quad (8)$$

Theoretically, when the design parameter ε is small enough, the RPT controller can give arbitrarily fast responses. Nevertheless, it is safer practically to limit the bandwidth of the outer loop to be much smaller than that of the inner-loop dynamics, because of the constraints of the UAV physical dynamics and its inner-loop bandwidth.

VI. Flight Test Results

T-Lion UAV implemented with the above-mentioned localization and tracking algorithm is used to carry out flight test inside an indoor building, with the aim of landing on a moving vehicle attached with the designed marker. In the flight experiment, the UAV is to take-off autonomously from the moving target, then hover and track the moving vehicle below it. Lastly, it is commanded to land on the vehicle while it is moving.

Snapshots of a few sequence of the experiment captured by the on-board camera is shown in Fig. 9, while the relative position between the UAV and the moving platform (the tracking result) is plotted in Fig. 10. In this experiment, at the time instant of 7 s, the UAV took-off from the platfrom. At time 30 s, the moving platform accelerate forward to reach a constant speed of 1 m/s. It can be seen from the plotted results that the UAV managed to accelerate and then track the platform well. At time 37 s, the platform comes to a stop and moving at the opposite direction immediately (backward). A large overshoot can be observe on the UAV as the moving platform exhibits large deceleration. Finally, the platform is commanded to move forward again and UAV is successfully landed on the moving platform starting from time 60 s. The images sequence in Fig. 9 clearly shows that the proposed algorithm robustly tracked the designed marker throughout the whole experiment, even worked well in the situation where the circular pattern of the marker is largely occluded as shown in the last image.

VII. Conclusions

The proposed vision and LiDAR based localization and tracking algorithm has been implemented on actual UAV, and flight experiments are carried out to verify the algorithm. The whole system was validated by experiments with result of successful landing on a moving platform with speed of 1 m/s in a GPS-denied environment. However, a few challenges and limitations of the current system need be solved. Firstly, the quadrotor is not capable of localizing while the marker is not in the view of the on-board camera. Next, flight control strategy needs to be developed in order to achieve fully autonomous patrolling, tracking and landing. Finally, the flight controller needs to be fine tuned in order to track a vehicle with sudden acceleration or deceleration.

References

[1] F. Kendoul, "Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems," *Journal of Field Robotics*, vol. 29, pp. 315–378, 2012.

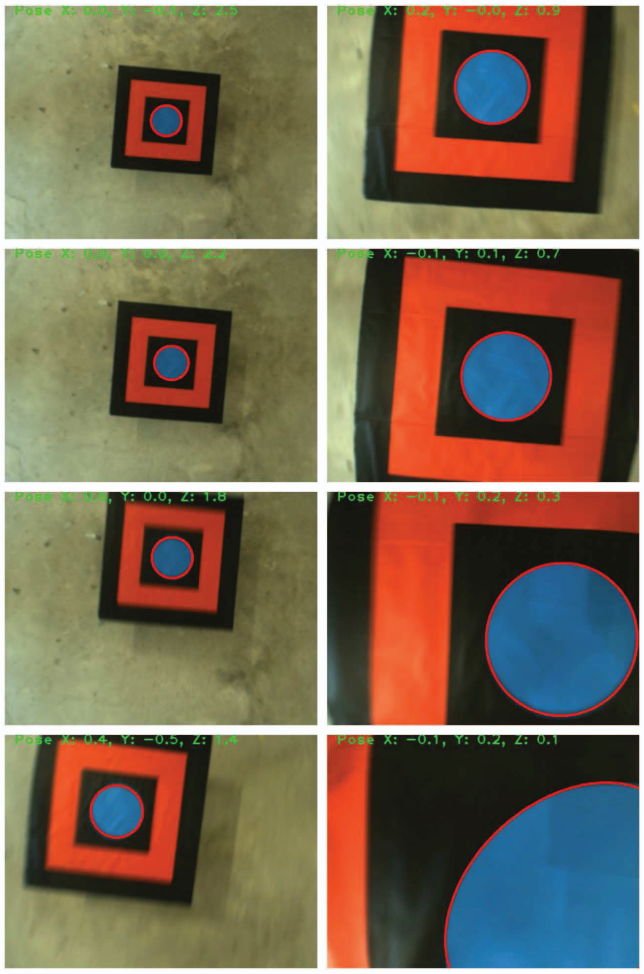

Fig. 9. Images sequence seen by on-board camera

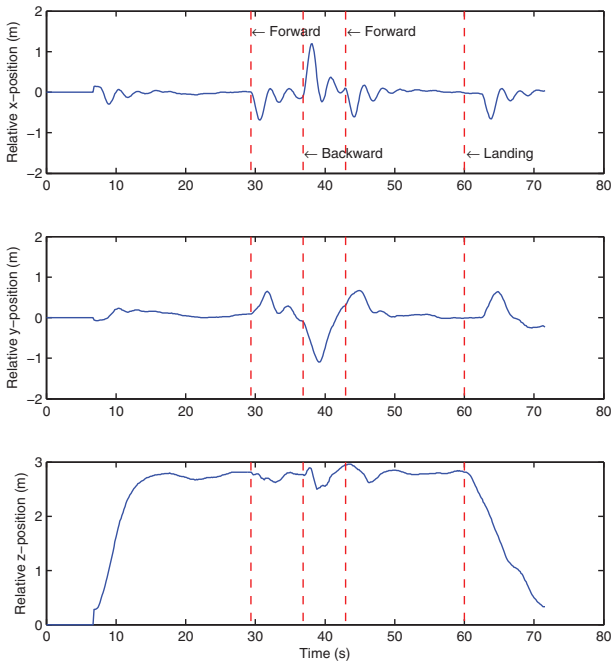

Fig. 10. Position of UAV in flight test

[2] D. Benbassat, C. Abramson, "Landing flare accident reports and pilot perception analysis," *International Journal of Aviation Psychology*, vol. 12, no. 2, pp. 137–152, 2002.

[3] D. Belton, S. Butcher, G. Ffoulkes-Jones, J. Blanda, "Helicopter recovery to a moving platform using a GPS relative positioning system," *Proceedings of the 12th International Technical Meeting of the Satellite Division of the Institute of Navigation*, pp. 1769–1776, 1999.

[4] F. Wang, P. Liu, S. Zhao, B. M. Chen, S. K. Phang, S. Lai, T. Pang, B. Wang, C. Cai, T. H. Lee, "Development of an unmanned helicopter for vertical replenishment," *Journal of Unmanned Systems*, vol.3, pp. 63-87, 2015.

[5] B. Pervan, F. Chan, G. Colby, "Performance analysis of carrier-phase DGPS navigation for shipboard landing of aircraft," *Navigation*, pp. 181–191, 2003.

[6] K. Gold, A. Brown, "A hybrid integrity solution for precision landing and guidance," *Position Location and Navigation Symposium*, pp. 165–174, 2004.

[7] M. Garratt, H. Pota, A. Lambert, S. Eckersley-Maslin, C. Farabet, "Visual tracking and lidar relative positioning for automated launch and recovery of an unmanned rotorcraft from ships at sea," *Naval Engineers Journal*, vol. 121, No. 2, pp. 99–110, 2009.

[8] O. Shakernia, R. Vidal, C.S. Sharp, Y. Ma, S. Sastry, "Multiple view motion estimation and control for landing an unmanned aerial vehicle," *IEEE International Conference on Robotics and Automation*, pp. 2793–2798, 2002.

[9] D. Hubbard, B. Morse, C. Theodore, M. Tischler, T. Mclain, "Performance evaluation of vision-based navigation and landing on a rotorcraft unmanned aerial vehicle," *IEEE Workshop on Applications of Computer Vision*, 2007.

[10] S. Lange, N. Sunderhauf, P. Protzel, "A vision based onboard approach for landing and position control of an autonomous multirotor UAV in GPS-denied environments," *International Conference on Advanced Robotics*, pp. 1–6, 2009.

[11] M. Laiacker, K. Kondak, M. Schwarzbach, T. Muskardin, "Vision aided automatic landing system for fixed wing UAV," *IEEE International Conference on Intelligent Robots and Systems*, pp. 2971–2976, 2013.

[12] B. Herisse, F. X. Russotto, T. Hamel, R. Mahony, "Hovering flight and vertical landing control of a VTOL unmanned aerial vehicle using optical flow," *IEEE International Conference on Intelligent Robots and Systems*, pp. 801–806, 2008.

[13] B. Herisse, T. Hamel, R. Mahony, F. X. Russotto, "A nonlinear terrain-following controller for a VTOL unmanned aerial vehicle using translational optical flow," *IEEE International Conference on Intelligent Robots and Systems*, pp. 3251–3257, 2009.

[14] B. Herisse, T. Hamel, R. Mahony, F. X. Russotto, "The landing problem of a VTOL unmanned aerial vehicle on a moving platform using optical flow," *IEEE International Conference on Intelligent Robots and Systems*, pp. 1600–1605, 2010.

[15] Z. Yu, K. Nonami, J. Shin, D. Celestino, "3D vision based landing control of a small scale autonomous helicopter," *International Journal of Advanced Robotic Systems*, vol. 4, pp. 51–56, 2007.

[16] S. Zhao, Z. Hu, M. Yin, K. Z.Y. Ang, P. Liu, F. Wang, X. Dong, F. Lin, B. M. Chen, T. H. Lee, "A robust real-time vision system for autonomous cargo transfer by an unmanned helicopter," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 2, pp. 1210–1219, 2015.

[17] B. M. Chen, T. H. Lee, V. Venkataramanan, *Hard Disk Drive Servo Systems*, Advances in Industrial Control Series, New York: Springer, 2002.

[18] D. Mellinger, V. Kumar, "Minimum snap trajectory generation and control for quadrotors," *Proceedings of 2011 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2520–2525, 2011.

[19] B. M. Chen, *Robust and H_∞ Control*. Communications and Control Engineering Series, Springer, 2000.