AUTONOMOUS LANDING OF A UAV ON A MOVING VEHICLE FOR THE MBZIRC*

L. CANTELLI, D.C. GUASTELLA, C.D. MELITA, G. MUSCATO
Dipartimento di Ingegneria Elettrica Elettronica e Informatica, Università di Catania, viale A. Doria 6, 95125, CATANIA, Italy

S. BATTIATO, F. D'URSO, G.M. FARINELLA, A. ORTIS, C. SANTORO
Dipartimento di Matematica e Informatica, Università di Catania, viale A. Doria 6, 95125, CATANIA, Italy

MBZIRC is an important robotic competition that will be held in Abu Dhabi in March 2017. The main objective of Challenge 1 is the autonomous landing of an UAV on a moving vehicle. This paper shows the architecture of the system developed by the team of the University of Catania, the different modules implemented and achieved preliminary results.

1. Introduction

The Mohamed Bin Zayed International Robotics Challenge 2017 (MBZIRC) is a robotic competition that will be held in Abu Dhabi in March 2017. The team of the University of Catania was selected to participate to the Challenge 1 and this paper reports an overview and some details on the developed system. The Challenge 1 consists in the autonomous landing of an Unmanned Aerial Vehicle (UAV) on a moving platform [1].

Autonomous landing on a moving vehicle is an important problem that has been investigated by different research groups worldwide [2]-[4]. Our group, in previous activities, has worked on the cooperation between UAVs and Unmanned Ground Vehicles (UGVs) in particular to help humanitarian demining operations [5]-[7] and for aerial monitoring [8], [9].

In the following sections, first an overview of the challenge will be described, than the developed system and its components will be presented. Focus will be given to the main modules related to the dynamical estimator and the vision system. Finally some results of simulations and of the on-field trials will be presented and commented.

* The work has been supported by the Khalifa University in Abu Dhabi via the sponsorship for participation in the MBZIRC challenge, for which our team was selected.

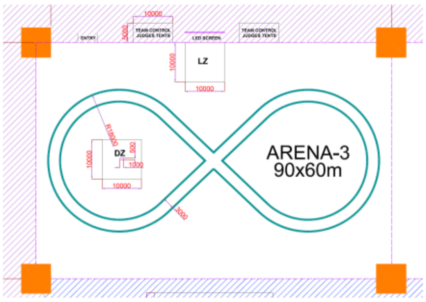

Figure 1. MBZIRC Challenge 1 Arena.

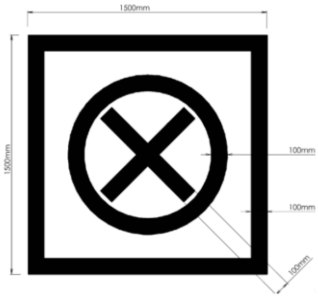

Figure 2. Pattern for the landing zone.

2. MBZIRC Challenge 1

According to MBZIRC rules, Challenge 1 requires a UAV to locate, track and land on a moving ground vehicle. The competition is performed in an open arena where a ground vehicle moves following an eight shaped trajectory, as shown in Figure 1. On top of the vehicle, the landing area is a square of dimensions 1.5m x 1.5m indicated by a given target (Fig. 2). The UAV has to take-off from a given position and autonomously land, in the shortest time, on the target on the moving vehicle. The speed of the vehicle is of 15 km/h for the first 8 minutes and then is reduced to 5 km/h for other 7 minutes. The maximum score will be given if the UAV lands on target fully autonomously, when the vehicle is moving at maximum speed.

3. Strategy and system architecture

Challenge 1 requires carefully taking into account control aspects, computer vision algorithms and the development or integration of suitable hardware needed to perform the autonomous tasks.

Our aim will be to complete the mission autonomously in a very short time. The basic overall approach we followed consists in taking off and reaching the center of the path by using a precise RTK-DGPS at an altitude suitable for a global view of the environment, for a preliminary detection and localization of the target from a depth estimation of the scene. A visual tracking procedure is capable of estimating the position of the target and to generate a suitable trajectory for the UAV. A dynamic estimator has been implemented that merges the measurements from the vision algorithm with the inertial and positioning measurements of the UAV and with the estimated trajectories of the UGV; then the estimator, on the basis of the UAV dynamic, generates in real time the optimal trajectory to reach the target.

Once in the proximity of the target, a computer vision algorithm is adopted for a precise estimation of the 3D position of the target to be used for safe landing. Once landed, all motors are switched off.

The emphasis has been put on the use of lightweight hardware platforms: to this aim, the computer vision and control algorithms will be optimized to run effectively on a lightweight high performance embedded system.

A key aspect is in the use of a high-resolution depth sensing camera, which will help the mission strategy software in recognizing the moving UGV: indeed, by using depth information, the background can be identified and subtracted from the visual data in order to detect the UGV; then, in the identified region, the landing target can be searched more easily.

4. Hardware and Software used

For the competition the chosen multirotor frame is the “Spreading Wings S900” by DJI, characterized by high payload and stability.

For the autopilot, we will use the PixHawk, a high-performance system able to deal with both the stabilization and the navigation of the UAV. This simple but powerful system can be connected to an on-board companion computer that, by running the high-level navigation algorithms, can easily drive the UAV.

The “eyes” of the multirotor are represented by a ZED camera, a 3D Camera for Depth Sensing and Motion Tracking developed by Stereolab. It gives as output both HD video and high frame-rate 3D Video Capture.

The image processing algorithm is executed by a Jetson TX1, an embedded system developed by NVIDIA for visual computing which provides a high performance GPU. The GPU-accelerated parallel processing uses the visible and depth information coming from the ZED camera to implement detection and tracking algorithms. The computed target position is used by the high level control algorithms, running also on the TX1, to give the correct commands to the Pixhawk autopilot by means of the Mavlink protocol.

The accuracy in the localization of the multirotor is ensured by an on-board RTK-DGPS system, receiving the corrections from a base station.

In Figure 3 the whole hardware platforms selected are shown.

5. Software Architecture

The control software runs on the TX1. The software architecture is designed as a multithread C/C++ application and it is executed on a Linux environment. Furthermore, for simulation purpose, the software is able to run inside a SITL (Software In The Loop) environment, using Gazebo as physics engine.

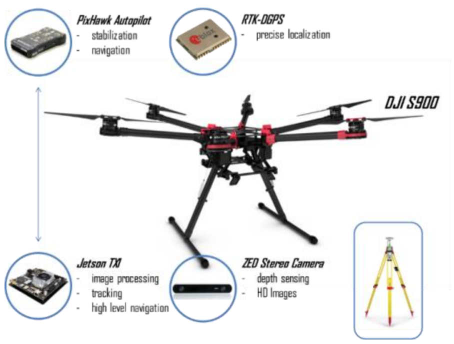

Figure 3. Hardware platforms used.

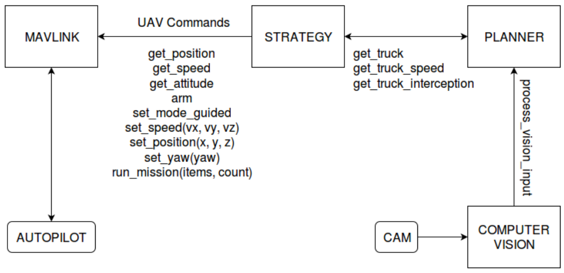

Figure 4. Software architecture.

The multithread process is made up of four threads, as shown in the figure 4.

MAVLINK, PLANNER and COMPUTER VISION are the threads that provide support to the STRATEGY one:

- The MAVLINK thread is used as an interface between the process and the autopilot. It allows translating messages from and to the autopilot through the Mavlink protocol.
- The COMPUTER VISION thread can acquire and analyse images from the ZED camera and gives the coordinates of the target as output to the PLANNER thread.
- The PLANNER thread is the interface between the main thread (STRATEGY) and a Finite State Machine (FSM). It receives coordinates from COMPUTER VISION in order to update the FSM and gives the position of the target over time as output to the main thread.
- The STRATEGY is the main thread and represents the decision-making module of the overall system. It has a continuous acknowledge of the state variables of both the system and the target. Its aim is to choose, in each condition, the best strategy to optimally achieve the result.

6. Computer Vision

6.1. *Detection module*

The scene is perceived with a ZED camera and the depth of the scene is reconstructed through stereo vision algorithms [10]. The reconstructed depth is useful to measure the distance between the camera and the viewed objects within a range of about 100 cm to 20m, at 100FPS.
To detect the target, we will use a simple algorithm to subtract the background based on the estimated depth and the information of the height drone. For each frame the system captures a points cloud through the ZEDcam. Each point is associated to a depth value. The depth map is dynamically thresholded considering the estimated distance of the target to the drone. This provides the region of interest of the image in which is located the target. The result of this process is a bounding box which is then fed to the target tracking module.

6.2. *Tracking module*

This module performs the tracking of the target on the vehicle [11]. To track the target, we exploit an algorithm based on the popular FERN features [12]. Specifically we build on the approach suggested in [13] . A pre-trained model is used as starting point for the detection of the target. The algorithm is able to improve the model by learning in real-time during the tracking of the target. To obtain a robust target model to follow, the algorithm learns from both positive and negative sample patches of the processed frames. For each frame, the ZEDcam framework provides a points cloud with the 3D coordinates for each image point. The 3D coordinates are expressed with respect to the ZEDcam local coordinate reference system. Then these coordinates are converted to global coordinates. The coordinates related to the tracked bounding box are used to estimate the position with respect to the point of view of the drone.

7. Dynamical Estimation (Planner)

Two algorithms have been implemented: the first allows to estimate the target position in real time and to predict its future position and the other one to evaluate the best rendezvous point between the UAV and the ground vehicle.
When the target is detected by the computer vision block, the state switches from IDLE (any input from computer vision) to two possible states:

- C POINT: the input coordinates belong to a neighbourhood of the centre of the track;

- NC POINT: the input coordinates are far from the centre of the track.

This distinction was needed because there is the possibility that the vision detects the target on the wrong straight segment: in this case a NC point should be detected to switch to the ACTIVE PREDICTION state.

To evaluate the reliability of the estimation process, a parameter, called Confidence, was adopted, that increases if the input from computer vision and the estimated point are consistent, otherwise it decreases; if Confidence is equal to one, the estimation process is wrong and is necessary to return to the previous state (C or NC point). The filtered data were used not only to evaluate the target position but also to estimate the real velocity and the direction of the vehicle.

When the state ACTIVE PREDICTION is reached, the coordinates of the target are known and it is then possible to estimate the future target positions and consequently the optimal rendezvous point.

8. Results

8.1. *Simulations*

Several simulations have been executed to test both the software architecture and the sub-blocks. PLANNER block has been extensively simulated in MATLAB/Simulink environment. The mission strategy has been improved by further simulations in both Gazebo (Fig. 5) and MATLAB (Fig.6) environments by introducing the dynamical estimation of the target, to generate in real time the optimal trajectory to reach the target.

The whole Software architecture has been initially simulated in Gazebo environment by using an ARUCO marker as target.

8.2. *On field trials*

Several on field tests have been performed to acquire real images and data; moreover target tracking and landing on the mobile platform have been executed. Initially the videos have been acquired by using a Phantom 3 DJI UAV, and then the ZEDCAM was mounted on an ASCTEC Firefly. Finally, the stereo-camera and the TX1 board have been installed and tested onboard the selected DJI S900 platform. The software architecture has been preliminary tested on a Raspberry PI board communicating to the Pixhawk autopilot and installed on two smaller UAVs (DJI F450, DJI F550). Many different trials have been also performed on the field arena concerning autonomous take-off, navigation and landing. Some results are also published on the video https://youtu.be/sP90fDK3flo .

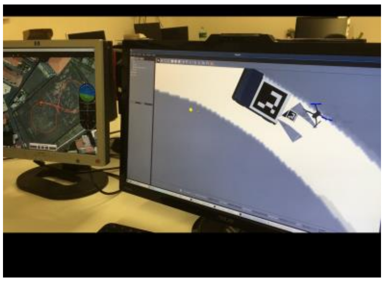

Figure 5. GAZEBO simulations.

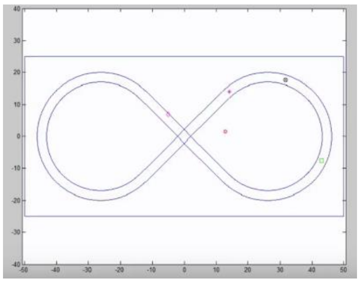

Figure 6. MATLAB/SIMULINK simulations.

Figure 7. S900 platform during the field trials.

9. Conclusions

This paper reported preliminary results of the system developed for the participation to the MBZIRC 2017 competition. The final paper will show the final system and the final results.

Acknowledgements

These results have been possible thanks to the great work performed by our students G. Galvagna, A. Mazzaglia, A. Scandurra, P. Statello, S. Saccone, G. Giammello, F. Leggio, F. Farinella, G. Sutera, L. Guarnera, F. Ragusa.
We are also grateful to all the companies that helped us during the work: STMicroelectronics, Micron, Etnamatica, Kevin Schurter and AAT Oranfresh.

References

1. MBZIRC website www.mbzirc.com .
2. P. Serra, R. Cunha, T. Hamel, D. Cabecinhas & C. Silvestre. “Landing of a Quadrotor on a Moving Target Using Dynamic Image-Based Visual Servo Control”, IEEE Transactions on Robotics, 32(6), 1524-1535, 2016.

3. S. Jin, J. Zhang, L. Shen & T. Li . "On-board vision autonomous landing techniques for quadrotor: A survey". In IEEE 35th Chinese Control Conference (CCC), pp. 10284-10289, July 2016.
4. O. Amidi, "An autonomous vision-guided helicopter", 1996. PhD Thesis. Carnegie Mellon University Pittsburgh, PA.
5. L. Cantelli, P. Laudani, C.D. Melita, G. Muscato, " UAV/UGV Cooperation to Improve Navigation Capabilities of a Mobile Robot in Unstructured Environments", Proceedings of CLAWAR 2016, London, September 2016.
6. L. Cantelli, M. Mangiameli, C.D. Melita, G. Muscato, "UAV/UGV cooperation for surveying operations in humanitarian demining", 11th IEEE International Symposium on Safety Security and Rescue Robotics, Oct 21-26, 2013 Linkoping, Sweden.
7. L. Cantelli, M. Lo Presti, M. Mangiameli, C.D. Melita, G. Muscato, "Autonomous Cooperation between UAV and UGV to improve navigation and environmental monitoring in rough environments", 10th International Symposium on Humanitarian Demining coupled with the 11th IARP WS HUDEM'2013, 23 April 2013, Šibenik ,Croatia.
8. De Benedetti M, D'Urso F, Messina F, Pappalardo G, Santoro C. "UAV-based Aerial Monitoring: A Performance Evaluation of a Self-Organising Flocking Algorithm", Proceedings of 2015 IEEE International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), 2015.
9. De Benedetti M, D'Urso F, Messina F, Pappalardo G, Santoro C, "Self-Organising UAVs for Wide Area Fault-tolerant Aerial Monitoring", WOA2015 CEUR WORKSHOP PROCEEDINGS, p. 142-145, ISSN:1613-0073, Napoli, 17-19 Giugno 2015.
10. R. I. Hartley, and A. Zisserman, "Multiple View Geometry in Computer Vision", Cambridge University Press, 2004.
11. S. Battiato, G. M. Farinella, A. Furnari, G. Puglisi, A. Snijders, J. Spiekstra, "An Integrated System for Vehicle Tracking and Classification", Expert Systems with Applications, Elsevier, ISSN 0957-4174, Vol. 42, N. 21, pp.7263-7275, doi:10.1016/j.eswa.2015.05.055, 2015
12. Ozuysal, M., Calonder, M., Lepetit, V., Fua, P.: Fast Keypoint Recognition Using Random Ferns. IEEE Transactions on Pattern Analysis and Machine Intelligence 32, 448–461 (2010).
13. Z. Kalal, K. Mikolajczyk and J. Matas, "Tracking-Learning-Detection," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, no. 7, pp. 1409-1422, July 2012.