Indoor Autonomous Navigation of Low-Cost MAVs Using Landmarks and 3D Perception

Ludovic APVRILLE
Institut Mines-Telecom
Telecom ParisTech, CNRS/LTCI
Campus SophiaTech, 450 Route des Chappes,
06410 Biot, FRANCE
Email: ludovic.apvrille@telecom-paristech.fr

Jean-Luc Dugelay
EURECOM
Campus SophiaTech, 450 Route des Chappes,
06410 Biot, FRANCE
Email: jean-luc.dugelay@eurecom.fr

Benjamin Ranft
FZI Research Center for Information Technology
Haid-und-Neu-Str. 10-14,
76131 Karlsruhe, GERMANY
Email: ranft@fzi.de

***Abstract*—We present an implementation of autonomous navigation for *Micro Air Vehicles* which is well-suited for very inexpensive models: It relies on a bottom and a front camera, and few additional on-board sensors to solve the challenges of flight planning and collision avoidance. Artificial landmarks may be used in specific places (lines in narrow corridors) and in places with an ambiguous further flight path, such as corridor crossings or junctions. For the latter case, they thus provide topological localization, which enables our system to perform tasks like way point following. Even without any 3D sensor, our system is also able to reconstruct metric distances from its monocular camera via two complementary methods: An oscillating motion pattern is superimposed to regular flight to reliably estimate up-to-date 3D positions of sparse image features. As an alternative, a specific flight maneuver can virtually create a vertical stereo camera to provide depth information densely across most pixels at single points in time.**

***Keywords*—Drones; UAV; 3D perception;**

I. Introduction

Drones, also knwon as UAV (Unmanned Aerial Vehicle) are flying machines which are remotely controlled. They are generally used for surveillance purpose and to sense information. Drones are currently used for oudoor applications, and, apart from the user interaction, their guidance is GPS-based. Since they are naturally much smaller than regular drones, mini drones are more likely to be used for indoor applications. Unfortunately, the GPS signal can not be efficiently used inside buildings, and so, mini-drones must permanently be remotely controlled, which strongly reduces their interest.

Thus, the EURECOM/Telecom ParisTech project called *Drone4u* intends to provide mini-drones with automated environment analysis capabilities. The objective of this analysis is to allow mini-drones to autonomously navigate inside buildings. Many "new" applications are targetted by autonomous navigation: guidance of persons in complex buildings (assistance to person), inspection of a building after a disaster (e.g., an earthquake), surveillance activities e.g. to know whether the building of a company is empty before locking it during the night.

Another objective of Drone4u is also to demonstrate that autonomous navigation can also be performed on low-cost mini drones (e.g., a few hundred euros), therefore with limited video processing capabilities, and cheap sensors. In that context, our work is mainly to invent and implement environment recognition algorithms. That recognition is applied in three contexts:

- To Follow of a colored line located on the floor.
- To identify landmarks located at crossings.
- To capture the environment of the drone in 3 Dimensions, with no 3D sensor.

First, the paper presents the general system architecture in next section. Then, each above-mentionned recognition schemes is detailed.

II. System architecture

The complete system is based on the monocular bottom or forward-facing camera, and supplementary on-board sensors of the inexpensive *Parrot AR.Drone 2.0* quadcopter [1] shown in fig. 1. Regarding infrastructure, our system requires a standard PC for remote processing and control, colored lines put on floor - if the line following part is necessary -, and landmarks for topological localization at crossings or junctions, but no external cameras or radio beacons. A full demonstration video is available at [2]. The remote PC connects to the quadcopter via WiFi. At this, our base system employs *Robot Operating System* [3], a widely-used open-source middleware. A supplementary driver [4] conveniently provides the *AR.Drone*'s camera images and sensor measurements, and accepts normalized control commands $\in [-1, 1]$ for each of the quadcopter's four degrees of freedom.

III. Following a line

Following a line is ensured by using the bottom camera only. The line is particularly useful in "difficult" environment, typically in a narrow corridor where a corkscrew flight would be delicate. The line may be considered as non intrusive in industrial building (e.g., a factory), but of course too intrusive

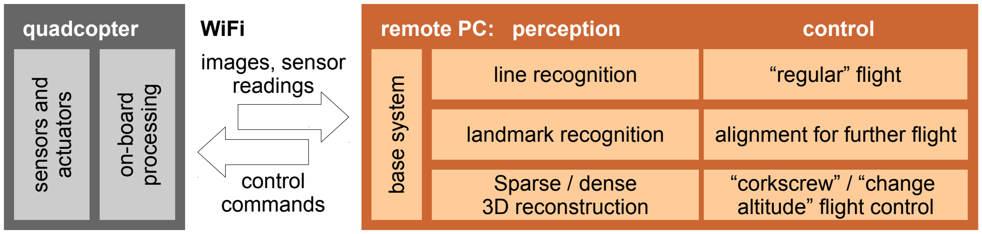

Figure 2. Overview of our system's modules: The perception methods may be run concurrently, but only one of their associated control strategies is selected based on their respective results.

Figure 1. *AR.Drone 2.0* quadcopter: Its 4 degrees of freedom during flight are indicated by arrows.

in a house. If a line is not pre-positionned, then the drone simply relies on 3D perception only.

IV. Landmark Recognition

Artificial landmarks are required for taking navigation decisions, e. g. for taking previously defined turns at corridor crossings and junctions. Fig. 3 shows an example of the currently employed wall-mounted markers, which have been developed in a student project [5]. They are detected and recognized via thresholding in the HSV color space: Detection only involves the green corners' relative positions, from which the landmark's distance and point of view can be derived as well. The 14 cyan/black dots along the edges allow recognizing $2^{12} = 4096$ individual landmarks, while two remaining bits are reserved for error detection via checksums.

V. 3D Perception

Two techniques have been defined for 3D perception. These techniques have been introduced so as to reconstruct the 3D environment of the drone using only the front camera.

- **Sparse 3D reconstruction** may be used continuously during regular flight and therefore is our preferred method of perception. It usually yields the spatial locations of few hundreds of distinct image points, see Figures 4 and 5. Their accuracy largely depends on the quadcopter's motion: Vertical and sideways movements are particularly beneficial, which is why our associated control strategy superimposes an oscillation in those directions, hereby creating a corkscrew-shaped flight trajectory.
- **Dense 3D reconstruction** can alternatively provide an estimated distance for most of the 265.000 pixels of an image, but in return requires exclusive flight control to virtually create a vertical stereo camera through a change in altitude (see Figure 6). Because regular flight needs to be interrupted for this maneuver, results are dense in space but sparse in time.

More information on those two techniques are throroughfully explained in [6].

VI. Conclusion and Future Work

We have briefly presented each module of a complete system which has proven capable of autonomous indoor navigation. Despite the absence of any inherent 3D sensor on our quadcopter, it is able to perform 3D reconstructions mainly based on a monocular camera. Even though it has been sufficient for our application, the quality of sparse 3D reconstruction and visual odometry can be improved by applying bundle adjustment techniques to longer feature tracks. Because of their computational complexity, it might be a challenge to embed the within the drone. The localization and prediction of moving objects using a monocular camera requires resolving individual scale ambiguities. Furthermore, visual place recognition such as [8] offers the opportunity to avoid the need for specific landmarks or lines, and to make our system fully independent from any infrastructure.

References

[1] P. SA, "Ar.drone 2.0. parrot new wi-fi quadricopter- specifications," *ardrone2.parrot.com/ardrone-2/specifications*, 2013.

[2] L. Apvrille, "Autonomous navigation of micro drones," *youtube.com/watch?v=tamYpmGvzRw*, 2013.

[3] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng, "Ros: an open-source robot operating system," in *IEEE International Conference on Robotics and Automation, Workshop on Open Source Software*, 2009.

[4] M. Monajjemi, "Autonomylab/ardrone_autonomy," *github.com/AutonomyLab/ardrone_autonomy*, 2013.

[5] J. de Campredon, "Report: Early image processing tests on uav (cont.)," *Student Project at EURECOM, Sophia Antipolis, France*, 2013.

[6] B. Ranft, J.-L. Dugelay, and L. Apvrille, "3d perception for autonomous navigation of a low-cost mav using minimal landmarks," in *International Micro Air Vehicle Conference and Flight Competition (IMAV'2013)*, Toulouse, France, Sept 2013.

[7] H. Hirschmüller, "Stereo processing by semiglobal matching and mutual information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 328–341, 2008.

[8] H. Lategahn, M. Schreiber, J. Ziegler, and C. Stiller, "Urban localization with camera and inertial measurement unit," in *IEEE Intelligent Vehicles Symposium*, 2013, pp. 719–724.

Figure 3. Landmark with machine-readable dot pattern and human-readable floor plan (left). Approach towards a detected yet un-recognized landmark (middle). Recognized landmark just before re-orientation for further flight (right).

Figure 4. Sparse 3D reconstructions: Blue/purple lines show optical flow vectors consistent/ conflicting with the camera's motion. The points color represents their longitudinal distance – red indicates 1 m and below, cyan for 10 m and above. A larger green circle marks the target flight direction.

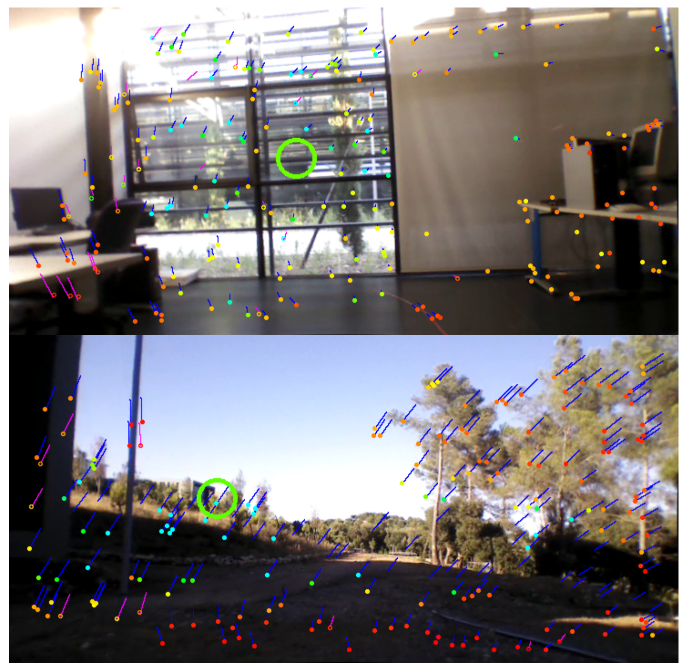

Figure 5. Imperfect sparse 3D reconstructions: A path through a window is planned because of too few correspondences (top). Difficult lighting caused erroneous estimates of camera motion and 3D point distances (bottom).

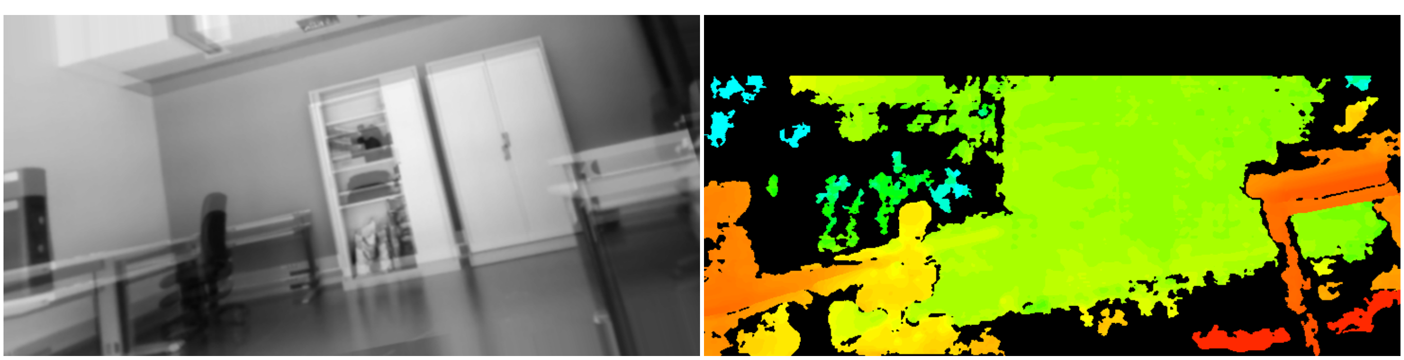

Figure 6. Dense 3D reconstruction results: The overlayed rectified images before and after the height change visualize the precision of the estimated camera motion (left). Therefore, any standard implementation for distance reconstruction, e. g. [7], may be used without modification (right).