System Development and Flight Experiment of Vision-Based Simultaneous Navigation and Tracking

Yoko Watanabe*, Charles Lesire†, Alain Piquereau‡ and Patrick Fabiani§
Department of Systems Control and Flight Dynamics
ONERA, Toulouse 31055, France

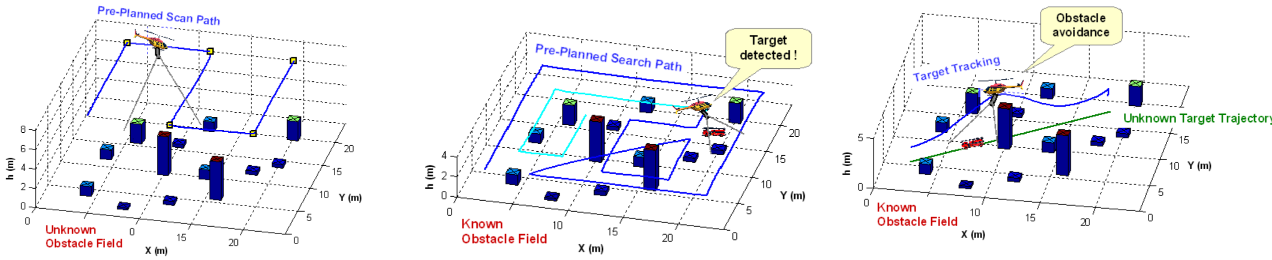
Martial Sanfourche¶ and Guy Le Besnerais‖
Department of Modeling and Information Processing
ONERA, Chatillon 92322, France

This paper outlines a UAV onboard vision-based system for simultaneous navigation and tracking of a moving ground target in a GPS-denied environment. The system consists of i) an image processor which detects the target and estimates the apparent motion of the groud surface, ii) a navigation filter which localizes both the own-ship UAV and the target by fusing onboard inertial sensor measurements and the image processing results, and iii) a guidance law which makes the UAV pursue the target by using the navigation results. The entire system including those algorithms is implemented as a single component of the Orocos robotic architecture, integrated with the auto-pilot system of the ONERA ReSSAC unmanned helicopter, and evaluated through closed-loop simulations and flight experiments.

I. Introduction

This paper addresses a UAV air-to-ground target tracking mission in an unknown urban environment. The mission is divided into three different operational phases: ($\phi 1$) global obstacle mapping, ($\phi 2$) target search and ($\phi 3$) target tracking. Figure 1 summarizes the entire mission scenario. First, the UAV explores the operation site at a sufficiently high and safe altitude to gather information which is post-processed to construct a 3D obstacle map ($\phi 1$). Then, the UAV searches a ground target at a lower altitude along a search path pre-planned based on the obstacle map ($\phi 2$). Once the target is detected, the UAV starts localizing and pursuing it while avoiding obstacles ($\phi 3$). The first phase can be achieved using the vision-based mapping algorithm developed in [1]. This paper focuses on developing the autonomous visual target search and tracking system. One of the challenges in this system is to enable an accurate global localization of both the own-ship UAV and the target while operating in a GPS degraded/denied environment. Two suggested approaches are used in combination. In case of GPS loss, the integrated vision/inertial navigation which utilizes sparse optical flow measurements to complement the UAV velocity information is applied [2]. Furthermore, since the vision-based navigation performance significantly depends on the camera motion relative to objects of interest (such as the target and the ground surface), the guidance law is designed to improve the UAV and the target localization accuracies while achieving the tracking mission. Such an idea is called dual control or observer trajectory optimization, and was first treated by Speyer in a bearing-only target interception problem [3]. This paper adopts the one-step-ahead optimization approach proposed in [4] so that the resulting guidance law creates motions preferable to the vision-based navigation.

*Post-Doctoral Researcher, Email: Yoko.Watanabe@onera.fr
†Research Engineer, Email: Charles.Lesire@onera.fr
‡Research Engineer, Email: Alain.Piquereau@onera.fr
§Director, Email: Patrick.Fabiani@onera.fr
¶Research Engineer, Email: Martial.Sanfourche@onera.fr
‖Research Engineer, Email: Guy.Le_Besnerais@onera.fr



(ϕ1) Obstacle Mapping (ϕ2) Target Search (ϕ3) Target Tracking

Figure 1. Air-to-Ground Target Tracking Mission Scenario

Another challenge addressed in this paper is real-time implementation and in-flight evaluation of the suggested visual navigation and tracking system. The onboard system of the ONERA ReSSAC unmanned helicopter consists of the basic auto-pilot system including a flight controller and a GPS/INS navigation filter and the decision architecture developed based on Orocos (Open RObot COntrol Software) [5]. All the algorithms in the system are integrated and implemented as a single component within this Orocos architecture. The implemented system is first tested through closed-loop simulations by connecting the Orocos-based architecture with the OpenRobots simulator that has been collaboratively developed at CNRS-LAAS and ONERA [6]. Then, the system is embedded in the onboard system of the ReSSAC helicopter and evaluated in its actual flights. The system performance is validated by achieving a closed-loop flight of purely vision-based target tracking.

This paper is organized in the following manner: Section II describes the visual air-to-ground target tracking system. Section III describes the onboard system of the ONERA ReSSAC helicopter and presents the flight experiment results. Section IV includes concluding remarks.

II. Vision-Based Navigation and Target Tracking System

Figure 2 illustrates the onboard system architecture of the vision-based air-to-ground target tracking. As shown in the figure, the system can be divided into three different components: i) image processor, ii) navigation filter, and iii) guidance and control system. This section details algorithms in each of these components.

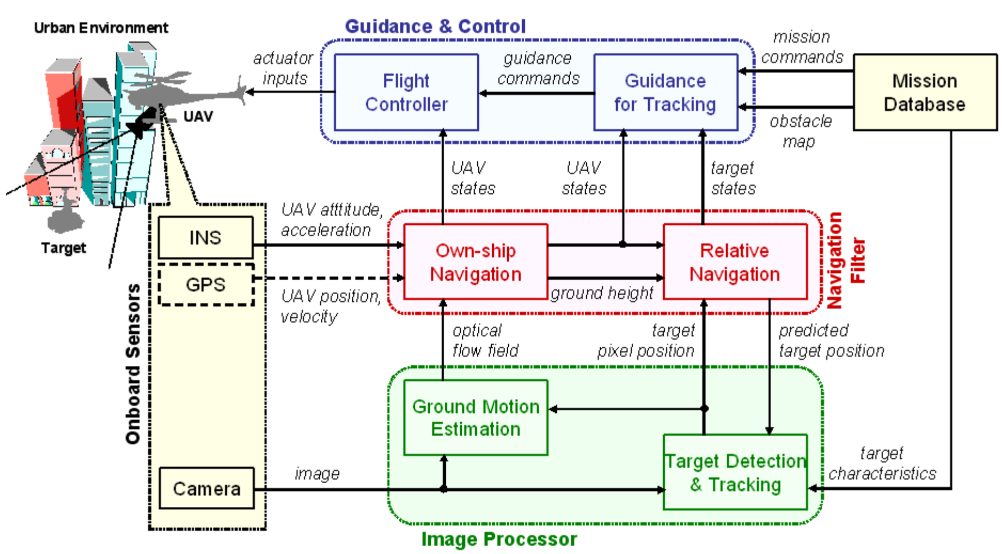
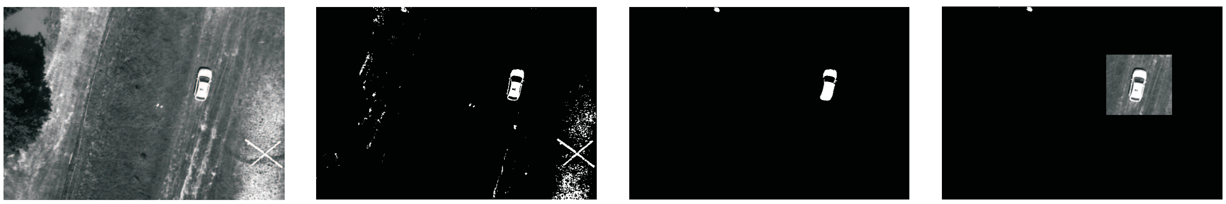


Figure 2. Vision-Based Navigation and Target Tracking System

A. Image Processor

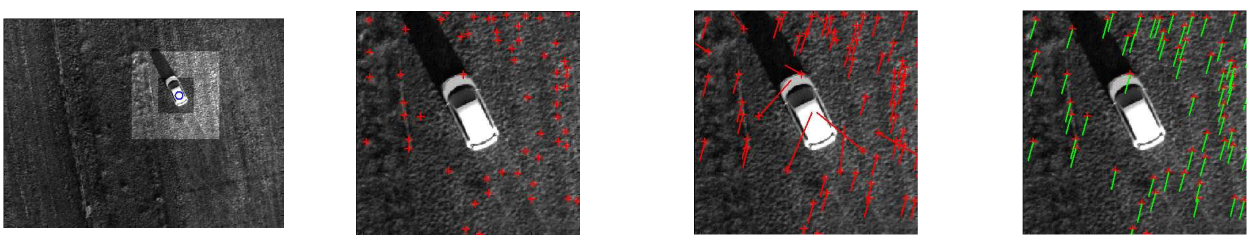
Two main tasks are devoted to the embedded image processor: ground motion estimation and target detection and tracking. Both tasks lead to problems of very different difficulty depending on the imaged scene: typically the urban environment is quite challenging. In urban areas, ground motion estimation uses optical flow estimation [7], but it is necessary to reject image regions which belong to superstructures (buildings, trees, etc.) and to moving objects. Starting from the output of a general-purpose point tracker one can design a process to robustly fit a parametric motion model over a group of frames, so as to maintain the temporal consistency of the registration ground plane. This video registration process is the basic block of a video tracking technique dedicated to aerial image-sequencing over urban areas described in [8].

In this paper, the target detection and tracking problem is made simpler by assuming a-priori knowledge of the target color and size. For example, the automatic detection illustrated in Figure 3 is performed based on the fact that the target's graylevel is significantly higher than the background. Given the detected target position, the ground motion can be more easily estimated in its neighborhood by making the hypothesis that ground surface around the target is flat. After a step of sparse optical flow estimation, an affine model is robustly fit to approximate the flow field (Figure 4). This fast process is used in the flight experiment described in Section III.



a) Original Image b) Binarized Image c) Object Extraction d) Target Detection

Figure 3. Automatic Target Detection

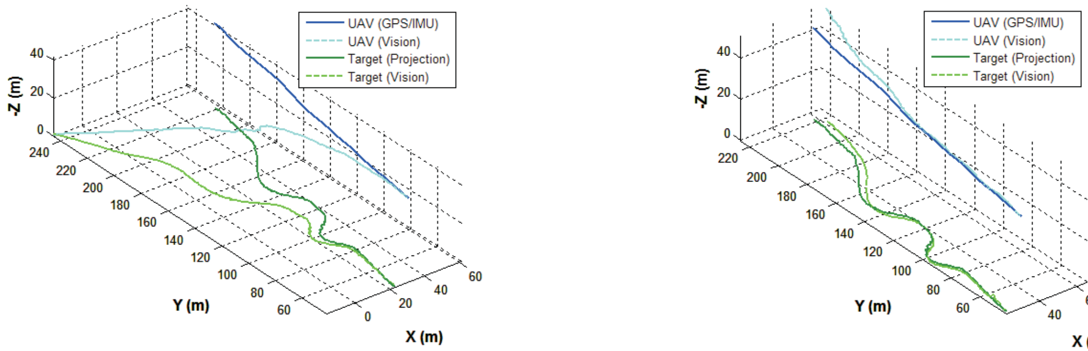


a) Target Detection b) Feature Point Selection c) Feature Point Matching d) Affine Approximation

Figure 4. Ground Motion (Sparse Optical Flow) Estimation

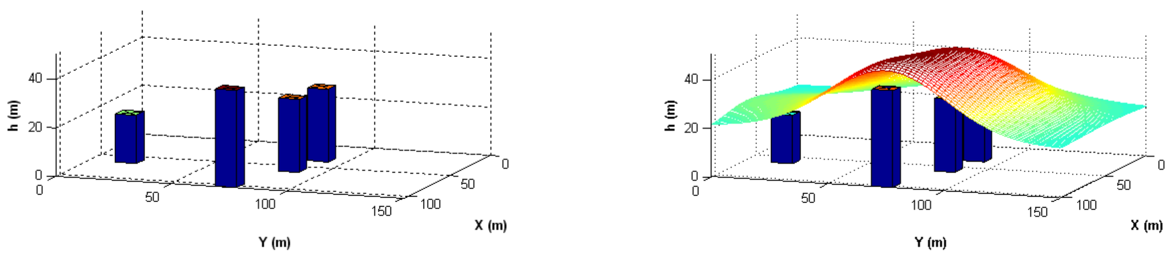
B. Navigation Filter

The navigation filter design applies an extended Kalman filter (EKF) to simultaneously estimate the global position and velocity of the UAV and those of the target by fusing the onboard sensor measurements with the image processing outputs. The GPS/INS navigation in general gives a very accurate estimate of the UAV state, while the INS-only navigation solution diverges very quickly due to an accumulation of the measurement bias. In order to prevent such a navigation divergence in case of occasional GPS loss, the ground motion estimation result of the image processor is utilized as a complementary velocity measurement of the UAV. This additional measurement also aides the target state estimation by providing ground height (i.e. the target vertical position) information. The details of this integrated vision/INS navigation design is described in [2]. Its performance has been verified through offline simulations using the actual vehicle state data synchronically recorded with the onboard camera images in open-loop flight of the ONERA ReSSAC helicopter. Figure 5 compares the self- and target-localization results of the INS-only navigation and the suggested vision/INS navigation when assuming an absence of GPS measurements. It is clear from the result that the optical flow measurement effectively complements the GPS information and eliminates the divergence in the INS-only navigation solution.



a) INS-Only Navigation b) Vision/INS Navigation (with Optical Flow)

Figure 5. Navigation Results with and without Optical Flow Measurement



a) 3D Obstacle Map b) Safety Altitude Map

Figure 6. An Example of Safety Altitude Map

C. Guidance Design

A UAV guidance objective of the target search operational phase is to follow a pre-planned search path (say $\boldsymbol{X}_{ref}$), and that of the target tracking phase is to track the target while avoiding obstacles. There is a large body of research in UAV urban navigation, including an optical flow-based biomimetic approach to obstacle avoidance [9]. For simplicity, it is suggested performing obstacle avoidance by maintaining a position-dependent safety altitude $h_d(X, Y)$ that is determined based on an obstacle map obtained during the first operational phase. Figure 6 illustrates an example of such an altitude map. Then, the UAV guidance problem becomes a position tracking problem where the desired position is given by

$$\boldsymbol{X}_d(t) = \begin{cases} \boldsymbol{X}_{ref}(t), & \text{during target search} \\ \left[\, X_t(t) \quad Y_t(t) \quad -h_d\left(X_v(t), Y_v(t)\right) \right]^T, & \text{during target tracking} \end{cases}$$

(X_t, Y_t) and (X_v, Y_v) are global horizontal position of the target and the UAV respectively. As shown in Figure 2, the guidance input is calculated by using the estimated state of the UAV and the target. The following linear feedback law is the simplest and most commonly used approach to determine the UAV acceleration input for position tracking.

$$\boldsymbol{a}_{com}(t) = -K_P(\hat{\boldsymbol{X}}_v(t) - \hat{\boldsymbol{X}}_d(t)) - K_D(\hat{\boldsymbol{V}}_v(t) - \hat{\dot{\boldsymbol{X}}}_d(t)) + \hat{\ddot{\boldsymbol{X}}}_d(t)$$

For example, the control gains K_P and K_D can be obtained by solving a LQR problem; However, this guidance can cause a large tracking error when the estimation error is large.

In vision-based navigation and control problems in general, the separation principal does not hold between estimation and control. It means that the navigation performance highly depends on relative motion of a camera with respect to objects of interest (such as the target and the ground surface). Particularly in bearing-only relative navigation, the depth information becomes unobservable when there is no lateral relative motion [10]. Therefore, this paper suggests introducing an extra input $\Delta \boldsymbol{a}_{com}$ to create motions that enhance the estimation performance and minimize the expected position tracking error. Because of its realtime applicability, the one-step-ahead (OSA) optimization approach proposed in [4] is applied to obtain $\Delta \boldsymbol{a}_{com}$. This approach performs the optimization under an assumption that there will be only one more measurement at the one time step ahead. Figure 7 compares simulation results of visual target tracking when using the nominal linear guidance law and the OSA optimal guidance law. When using the linear guidance, the UAV ends up crashing due to a drift in the height estimation. On the other hand, the optimal guidance generates a small lateral motion to maintain the depth observability and prevents the crash from occuring. It has been also verified through simulations that the proposed guidance law is effective for the optical flow-based self-navigation proposed in the previous subsection. Figure 8 shows simulation results in which an absence of GPS signals during the target tracking is assumed. In this situation, the OSA optimal guidance law takes into account the effect of the camera motion both on the optical flow-based self localization and on the vision-based target localization. Figure 8 plots the resulting UAV trajectories when using the nominal and the optimal guidance laws, compared with the desired path. When using the nominal guidance, the UAV fails to avoid the obstacle due to the error in its self localization. The UAV horizontal position is overestimated, and hence it decends to the commanded altitude before flying over the obstacle. This collision is prevented by enhancing the UAV self localization accuracy by creating the additional camera motion relative to the ground surface on which optical flow is calculated.

III. Onboard System and Flight Experiment

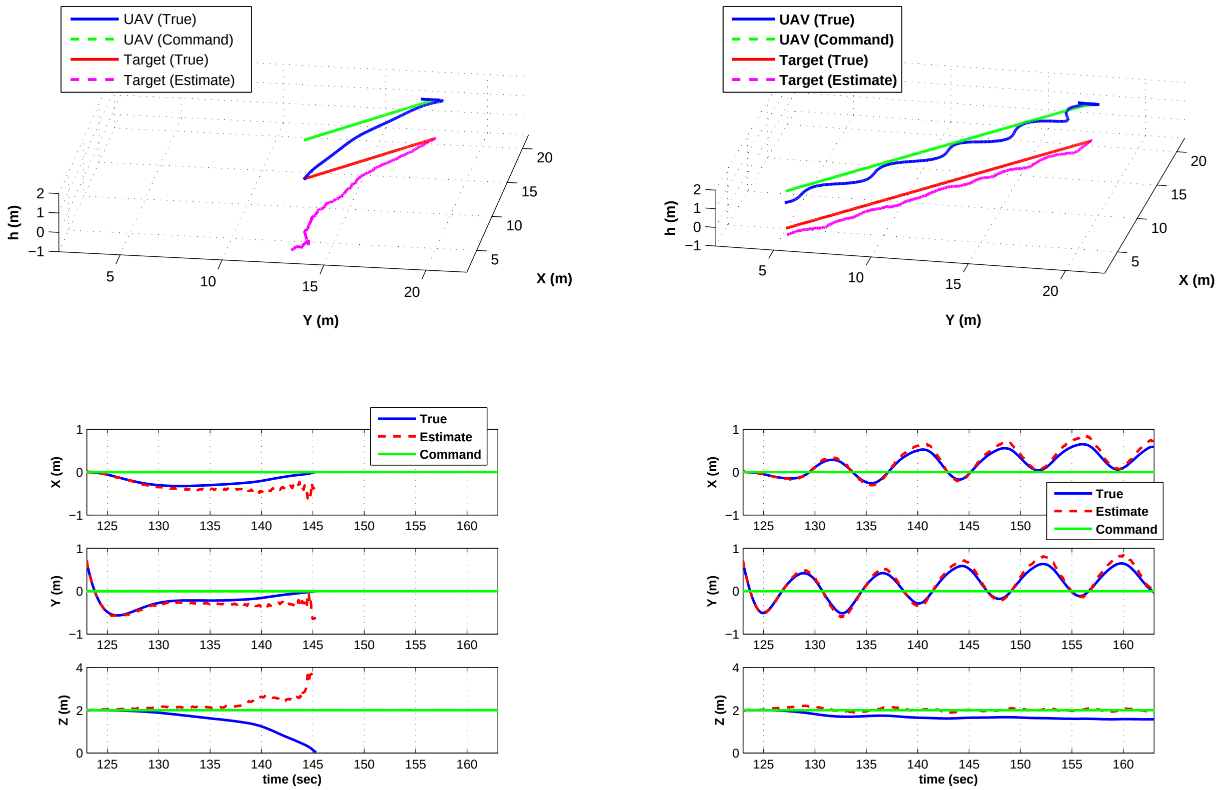
An ultimate goal of this work is to implement the vision-based navigation and target tracking system designed in Section II onboard on the ONERA ReSSAC unmanned helicopter, and to evaluate the system in its flights. This section explains the embedded architecture of the ReSSAC UAV and presents some flight experiment results.

A. The ONERA ReSSAC Unmanned Helicopter

The ONERA ReSSAC helicopter is an experimental platform that has been developed based on an industrial unmanned helicopter YAMAHA RMax. Table 1 summarizes its specifications. The onboard system of the ReSSAC helicopter is composed of two processors. The first one is dedicated to a basic auto-pilot system including the GPS/INS navigation and the flight controller that have been described in previous publication [11]. The second one is for the decision architecture which is in charge of mission management, decision-making and supervision. The visual target tracking system proposed in this paper is implemented on this decision architecture. The PIP11 (MPL) hardware unit which incorporates the embedded Pentium M with 1.8 GHz and communication ports of RS-232, Firewire and Ethernet is used for the second processor. The onboard camera is connected to the PIP11 via Firewire, and it is set to send images of size 640×480 pixels at 10 Hz. The two onboard processors interact and communicate through two RS-232 serial connections. One allows the decision architecture to obtain the UAV estimated state from the auto-pilot system, and the other to send a guidance command request. The Ethernet/Wifi bridge is used to connect the decision architecture to the ground control station.

Table 1. Specifications of the ONERA ReSSAC Helicopter

Model	YAMAHA RMax
Length (including main rotor)	3.63 (m)
Empty Weight	60 (kg)
Payload	20 (kg)
Onboard Sensors	GPS, INS, Compass, Barometer, Camera, LRF etc.



a) Nominal Linear Guidance b) One-Step-Ahead Optimal Guidance

Figure 7. Vision-Based Relative Navigation and Target Tracking Results: UAV and target trajectories (top), true and estimated relative positions (bottom)

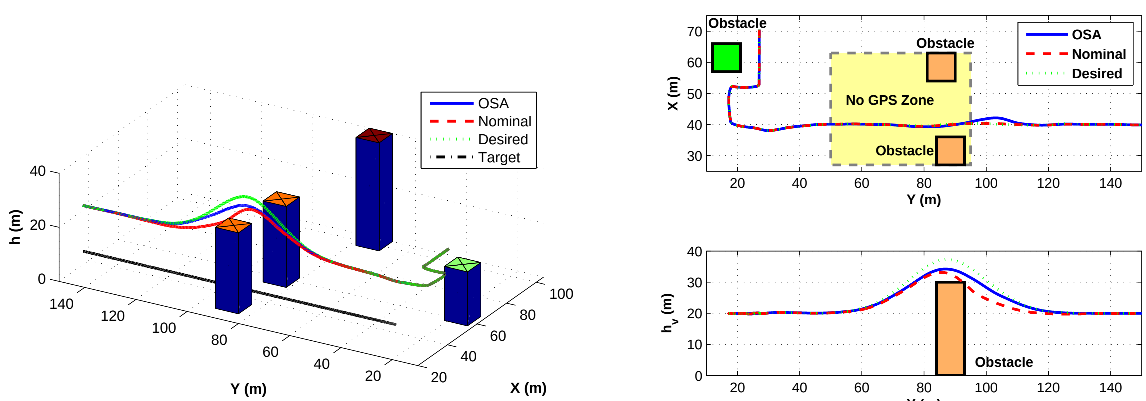


Figure 8. Obstacle Avoidance with and without Optimal Guidance

B. Orocos-Based Decision Architecture

The decision architecture is executed on a GNU/Linux Debian system and is based on Orocos middleware [5]. Orocos is an open source robotic framework, which offers a real-time toolkit (RTT) that manages interactions and execution of components that are defined and developed by a user. A standard Orocos component interface is shown in Figure 9. Each component has data ports and services to provide. It can react to events, process service requests and execute scripts in real-time. Such a component may be connected to hardware devices (camera, controller etc.) or it may integrate processes. All the algorithms in the visual target tracking system including the image processor, the navigation filter and the guidance law are implemented in a single Orocos component in C++ based on the Orocos RTT library. Besides this main "Target Tracking" component, there are components which are connected to the onboard camera and to the primary processor, and also components which perform data recording. The entire system is built by connecting and activating these components as illustrated in Figure 10. Execution of each Orocos component is monitored and controlled by a special component, called *Deployer*. Deployer is considered as a central component in terms of control flow.

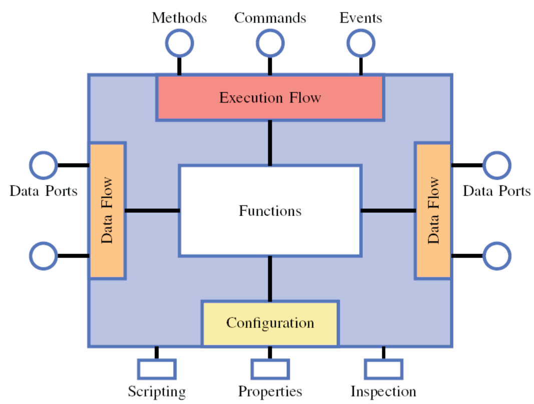


Figure 9. Orocos Component Interface

C. Closed-Loop Simulation with OpenRobots Simulator

After implemented in the Orocos architecture, the visual target tracking system is first verified through the software-in-the-loop (SITL) simulation. The SITL simulation uses the OpenRobots simulator that has been collaboratively developed at CNRS-LAAS and ONERA.[6] The OpenRobots simulator is built based on Blender and Python script language, and it is able to simulate multiple mobile robots in a 3D dynamic environment. It can also emmulate onboard sensor measurements (such as GPS, inertial sensors and camera) and communication link between the robots. Figure 11 shows the interface of the OpenRobots simulator while performing the closed-loop target tracking simulation. In this simulation, motion of the ground 'target' robot was given manually via keyboard. The left-top window appeared in Figure 11 is the emmulated camera image. The Orocos-based architecture is connected to the OpenRobots simulator using Yarp. It receives the simulated UAV onboard sensor measurements and camera images from the simulator, processes those data, and sends back the UAV velocity command to the simulator. The SITL simulation is very beneficial in debugging the implemented system before conducting flight experiments using the actual vehicles.

D. Flight Experiment

First, a simplified version of the target tracking system (assuming no obstacles and no loss of GPS signal, and hence without the optical flow estimation algorithm) has been developed and implemented in the Orocos architecture onboard the ReSSAC helicopter. An entire process including the image processor (automatic detection and target tracker), the relative navigation filter and the guidance law runs at 10 Hz. The guidance law outputs a set of the UAV horizontal velocity, height and heading angle commands based on the visually estimated relative state, and sends these commands to the flight controller which calcultes actuator inputs of the helicoter to realize the command while stabilizing the vehicle. Flight experiments of air-to-ground

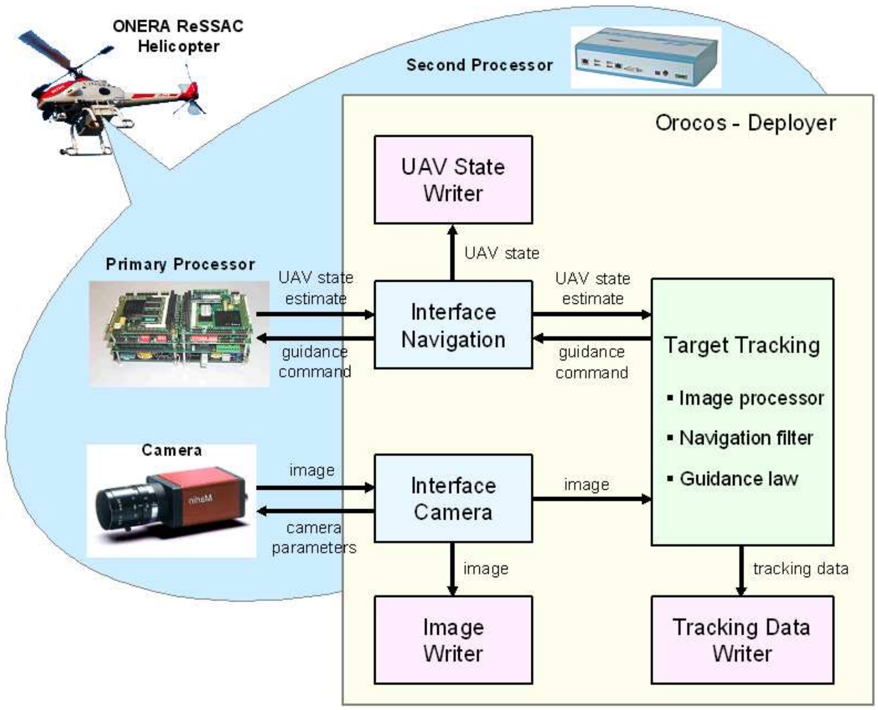


Figure 10. Orocos-Based Decision Architecture

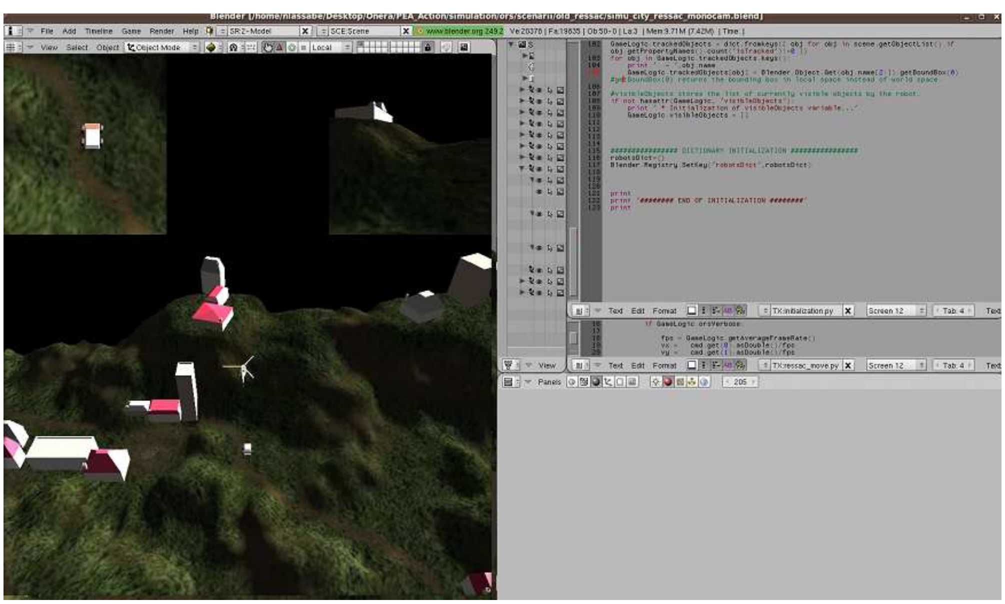
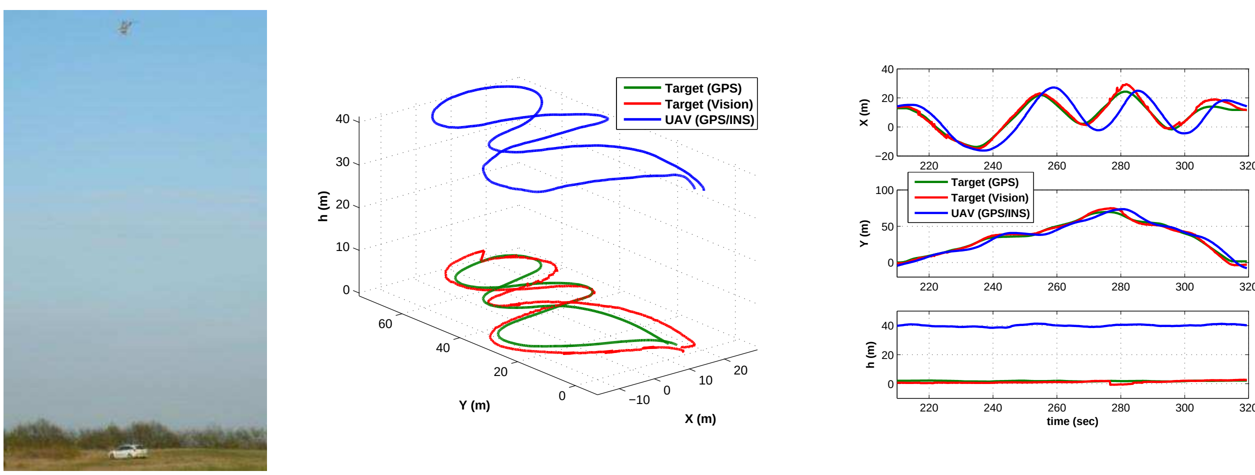


Figure 11. OpenRobots Simulator Interface

target tracking have been conducted using a manually-driven car as a moving ground target, and purely vision-based closed-loop flights have been successfully achieved. Figure 12 presents the flight test results of the UAV tracking trajectory and the target GPS-measured and vision-based estimated trajectories. To ensure flight safety, constant height and heading angle commands were used in this experiment. The results prove that the target position is accurately estimated by using its image pixel coordinate information and hence the UAV pursues the target with a good precision. By using the same system, the closed-loop target tracking was also achieved in a combat training village, a more complex environment. Figure 13 depicts the resulting target and UAV trajectories from this experiment.

Then, the OSA optimal guidance policy was augmented to improve the vision-based relative navigation performance, and tested for the first time in actual flight. Figure 14-a) shows the UAV horizontal trajectory compared with the GPS-measured target trajectory. 14-b) is the relative position estimation result. As seen in the simulation result presented in Section C, the OSA optimal guidance law creates some lateral motions relative to the target in order to improve the depth observability.

Now the embedded system will be completed by the optical flow estimation, the optical flow-based self-navigation and the obstacle avoidance algorithms. The optical flow estilmation algorithm has been already implemented in the Orocos architecture, and tested in flight by running it with the target detection and tracking algorithm. It is observed that the optical flow estimation takes about 150 (msec) on the onboard processor and the target tracking system can no longer run at 10 Hz. We will try to reduce this processing time before implementing the optical flow-based self navigation filter. The ultimate goal is to perform closed-loop target tracking in the presence of obstacles and occasional loss of GPS signals. This will be the first attempt of a GPS-free automatic flight of the ReSSAC helicopter.



a) Experiment Scene b) 3D Trajectory c) Global Position Profile

Figure 12. Flight Test Result of Closed-Loop Vision-Based Target Tracking

IV. Conclusion and Future Work

This paper proposed the UAV navigation and guidance system for vision-based ground target search and tracking in a GPS-denied urban environment. It is suggested to utilize sparse optical flow to aide UAV self-navigation when GPS information is not available. Moreover, the optimal guidance law is applied to improve navigation accuracy by taking into account the inseparability between control and vision-based estimation. The embedded software architecture is developed based on Orocos in order to implement the suggested system into the onboard processor of the ONERA ReSSAC helicopter. The implemented system was first verified through the software-in-the-loop simulation by using the 3D robotic simulator. Then, closed-loop purely vision-based target tracking have successfully been achieved in flight with this architecture. The optical flow-based self localization algorithm is expected to be implemented and tested in flight shortly. For future work, we aim to augment the system with mission planning and decision making algorithms so that it can be applied to a more complex mission scenario.

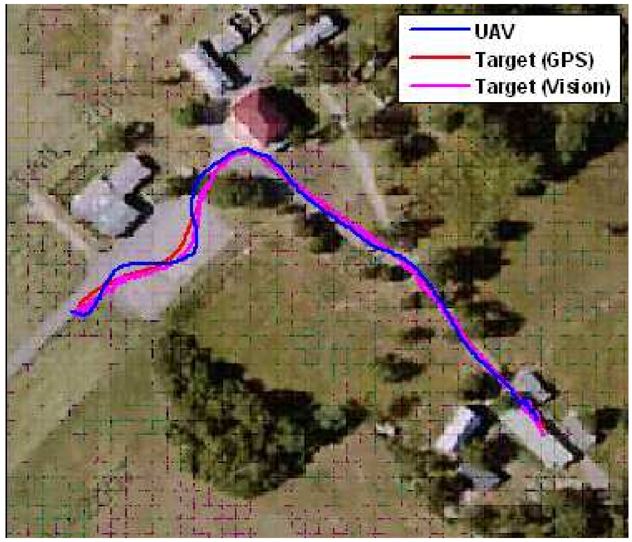
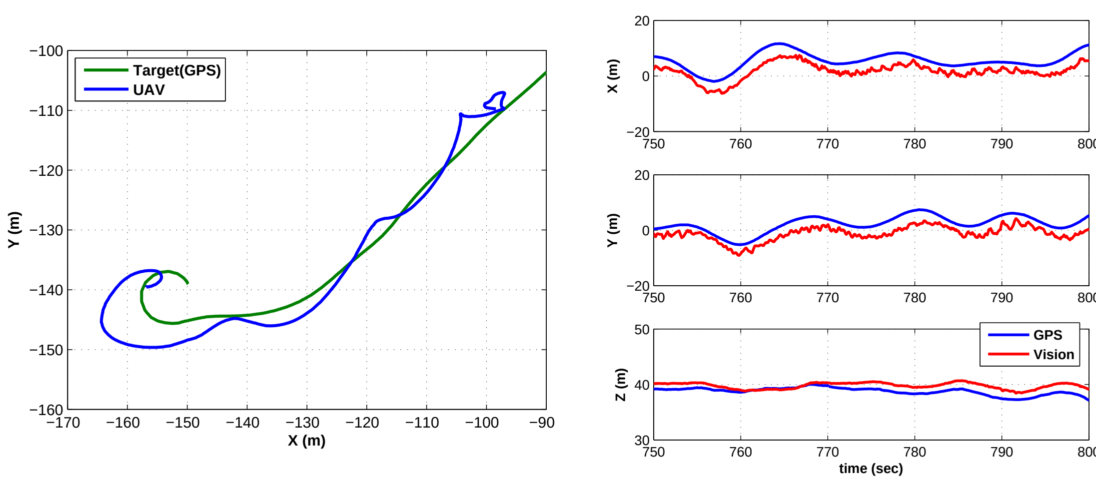


Figure 13. Closed-Loop Target Tracking Results in the Combat Village



a) Horizontal Trajectories

b) Relative Position Estimate

Figure 14. Closed-Loop Target Tracking with Optimal Guidance

References

[1]M. Sanfourche, G. Le Besnerais, P. Fabiani, A. Piquereau and M.S. Whalley, "Comparison of Terrain Characterization Methods for Autonomous UAVs", *American Helicopter Society (AHS) 65th Annual Forum*, 2009.

[2]Y. Watanabe, P. Fabiani and G. Le Besnerais, "Simultaneous Visual Target Tracking and Navigation in a GPS-Denied Environment," *International Conference on Advanced Robotics (ICAR)*, 2009.

[3]J. Speyer, D. Hull, C. Tseng and S. Larson, "Estimation Enhancement by Trajectory Modulation for Homing Missiles," *AIAA Journal of Guidance, Control and Dynamics*, 7(3), 1984.

[4]Y. Watanabe, "Stochastically Optimized Monocular Vision-Based Navigation and Guidance," *Ph.D. Thesis, Georgia Institute of Technology*, 2008.

[5]P. Soetens and H. Bruyninckx, "Realtime Hybrid Task-Based Control for Robots and Machine Tools," *International Conference on Robotis and Automation (ICRA)*, 2005.

[6]G. Echeverria, S. Lemaignan and N. Lassabe, "OpenRobots Simulator," `https://launchpad.net/openrobots-simulator`, 2009.

[7]G. Le Besnerais and F. Champagnat, "Dense Optical Flow Estimation by Iterative Local Window Registration," *IEEE International Conference on Image Processing (ICIP)*, 2005.

[8]E. Pollard, A. Plyer, B. Pannetier, F. Champagnat and G. Le Besnerais, "GM-PHD Filters for Multi-Object Tracking in Uncalibrated Aerial Videos," *International Conference on Information Fusion (FUSION)*, 2009.

[9]L. Muratet, S. Doncieux and J.A. Meyer, "A Biomimetic Reactive Navigation System Using the Optical Flow for a Rotary-Wing UAV in Urban Environment," *International Symposium on Robotics*, 2004.

[10]L. Metthies and T. Kanade, "Kalman Filter-based Algorithms for Estimating Depth from Image Sequences," *International Journal of Computer Vision*, 1989.

[11]P. Fabiani, A. Piquereau, V. Fuertes and P.M. Basset, "The ReSSAC Autonomous Rotorcraft: from autonomy demonstrations to out-of-sight flights," *AHS Specialist Meeting on Unmanned Rotorcraft*, 2005.