Optical Flow-Based Controller for Reactive and Relative Navigation dedicated to a Four Rotor Rotorcraft

Eduardo Rondon, Isabelle Fantoni,
Anand Sanchez, Guillaume Sanahuja

Heudiasyc UMR UTC/CNRS
Université de Technologie de Compiègne

Outline

- Introduction
- Visual System
- Quad rotor Dynamics
- Control Strategy
- Results, vidéos
- Conclusions and future works

Introduction

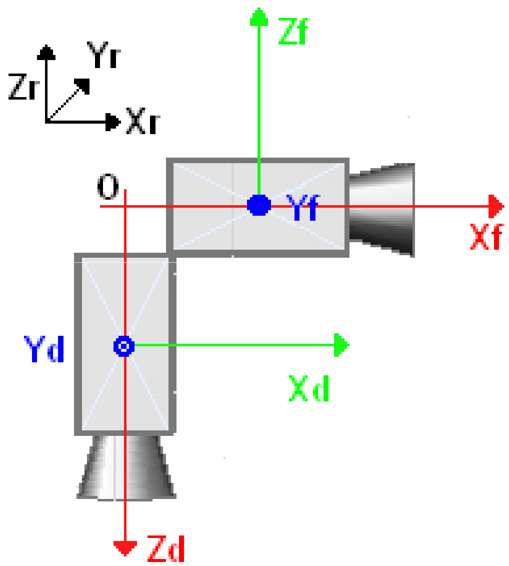
- Projet NAVIFLOW (2007-2010), Fondation de Recherche pour l'Aéronautique et l'Espace,
  Assistance à la NAVIgation par FLux Optique
- Programme de recherche « Autonomie des systèmes aéronautiques et spatiaux »
- Partenaires: Robosoft, CEA List, ONERA Toulouse, Heudiasyc

- Navigation réflexe en milieu urbain, basée sur la fusion de capteurs de vision utilisant des algorithmes de traitement de flux optique avec des capteurs inertiels, d'altitudes, en visant des expérimentations réelles
- Développement complémentaire aux techniques plus conventionnelles de type SLAM qui souffrent de problèmes de robustesse et de temps de calcul

Visual System

Cameras Setup

- Optical flow from the frontal camera is used to detect and avoid obstacles
- Optical flow from the downward camera is used to regulate translational speed
- <u>Definition</u> :
  Optical flow: Visual motion field of image points obtained by projection in 2D of objects velocities in movement from the 3D space



Positions of the cameras

Visual System

Optical flow computation

- Lucas-Kanade pyramidal algorithm
- Fusion approach between the IMU and the optical flow based on Kalman Filter theory

$$\bar{OF}_x^d = \bar{V}_{OF_x}^d + K_x^x \bar{V}_{OF_z}^d + K_{xy}^x \omega_x - K_{x^2}^x \omega_y + K_y^x \omega_z$$
$$\bar{OF}_y^d = \bar{V}_{OF_y}^d + K_y^y \bar{V}_{OF_z} v + K_{y^2}^y \omega_x - K_{xy}^y \omega_y - K_x^y \omega_z$$

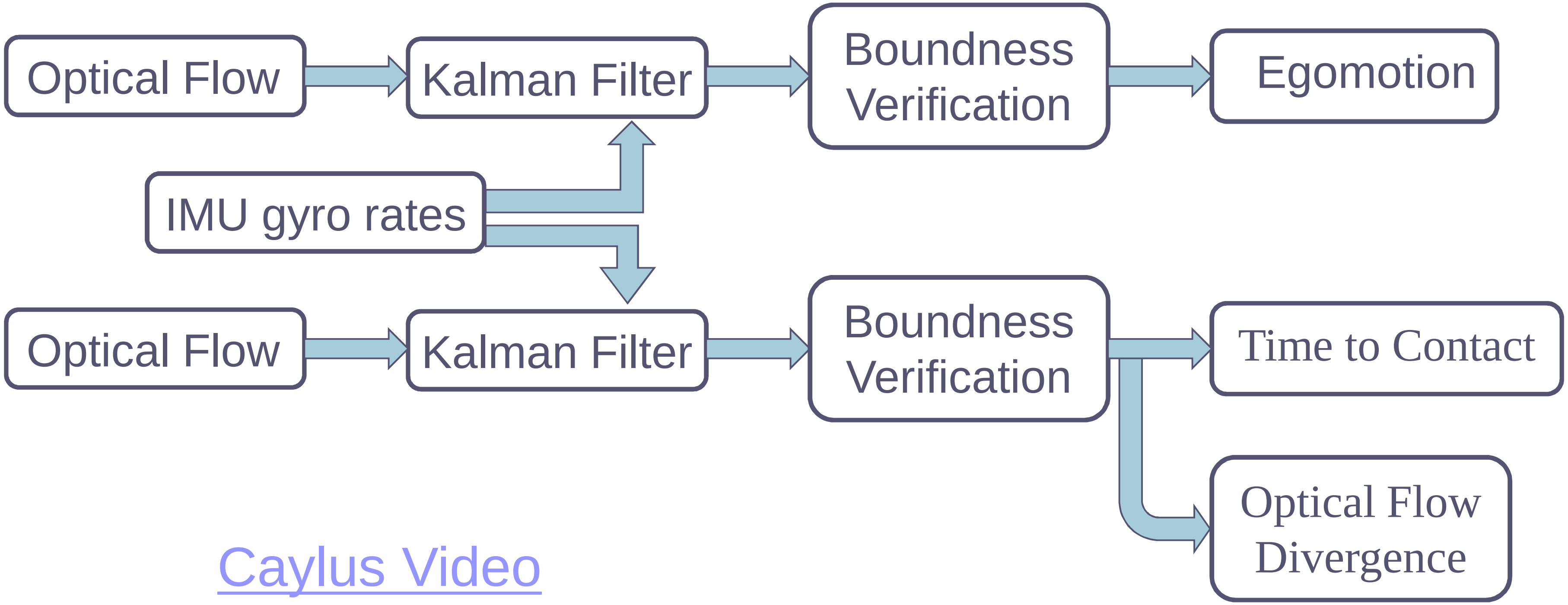
- Rejection of the wrong measurements in order to eliminate the wrong points

$$\|OF_{x_i}^d - (K_{xy}^x \omega_x - K_{x^2}^x \omega_y + K_y^x \omega_z)\| \leq \sigma_i^o \tilde{V}_{OF_x}$$
$$\|OF_{y_i}^d - (K_{y^2}^y \omega_x - K_{xy}^y \omega_y - K_x^y \omega_z)\| \leq \sigma_i^o \tilde{V}_{OF_y}$$

- Estimation of the three translational components of the optical flow

Visual System

Optical flow computation



Caylus Video

Quad rotor dynamics

- Obtained from Euler-Lagrange equations
- Quad rotor dynamic is commonly known in the literature

Four rotor helicopter

Quad rotor dynamics

- Dynamics on the axes X and Y are not taken into account
- Dynamic on the Z axis is used to stabilize altitude
- Rotational dynamics used to stabilize the attitude

$$
\begin{aligned}
m\ddot{z} &= u\cos\theta\cos\phi - mg \\
m\ddot{\theta} &= \tilde{\tau}_\theta \\
m\ddot{\phi} &= \tilde{\tau}_\phi \\
m\ddot{\psi} &= \tilde{\tau}_\psi
\end{aligned}
$$

Dynamical equations

- At a constant pitch angle, the vehicle moves in the forward direction
- At a constant roll angle, the vehicle is displaced at a constant velocity in the lateral direction

Control Strategy

1. Inner loop

- Attitude and altitude stabilization
- Attitude stabilization made using saturation functions
- The reference angles on pitch and roll are given by the optical flow controller
- Altitude stabilization with a linear controller

$$\begin{aligned}
\tilde{\tau}_\theta &= -\sigma_{\theta_1}\left(K_1^\theta\left(\theta-\theta_{ref}\right)\right)-\sigma_{\theta_2}\left(K_2^\theta\dot{\theta}\right)\\
\tilde{\tau}_\phi &= -\sigma_{\phi_1}\left(K_1^\phi\left(\phi-\phi_{ref}\right)\right)-\sigma_{\phi_2}\left(K_2^\phi\dot{\phi}\right)\\
\tilde{\tau}_\psi &= -\sigma_{\psi_1}\left(K_1^\psi\psi\right)-\sigma_{\psi_2}\left(K_2^\psi\dot{\psi}\right)
\end{aligned}$$

$$u=m\left(-k_1^z\dot{z}-k_2^z\left(z-z_{ref}\right)+g\right)$$

Control laws

Control Strategy

2. Optical flow controller (Downward camera)

- Vehicle's speed sensed with downward camera used to design a speed controller which regulates the roll and pitch reference angles
- Downward camera is used to regulate speed
- The goal is to maintain the optical flow constant
- A linear controller is implemented to regulate speed via the reference angles
- Two flying modes are defined: frontal displacement and lateral displacement

$$\theta_{ref} = k_p^{\theta}\left(\bar{V}_{OF_x}^{d} - \bar{V}_{OF_x}^{d} ref\right) + k_i^{\theta}\int_0^{Te}\left(\bar{V}_{OF_x}^{d} - \bar{V}_{OF_x}^{d} ref\right)dt$$

Example of controller, regulation of the pitch angle

Control Strategy

3. Optical flow controller (Frontal camera)

Detection subsystem

- In frontal displacement. time-to-contact is used to reduce speed of travel

$$\theta_{prev} = k^{\theta}_{prev}\eta$$

- The speed is reduced by subtracting the time to contact information to the reference angle

Avoidance subsystem: lateral or altitude avoidance

- In lateral displacement, optical flow divergence is used to avoid obstacles

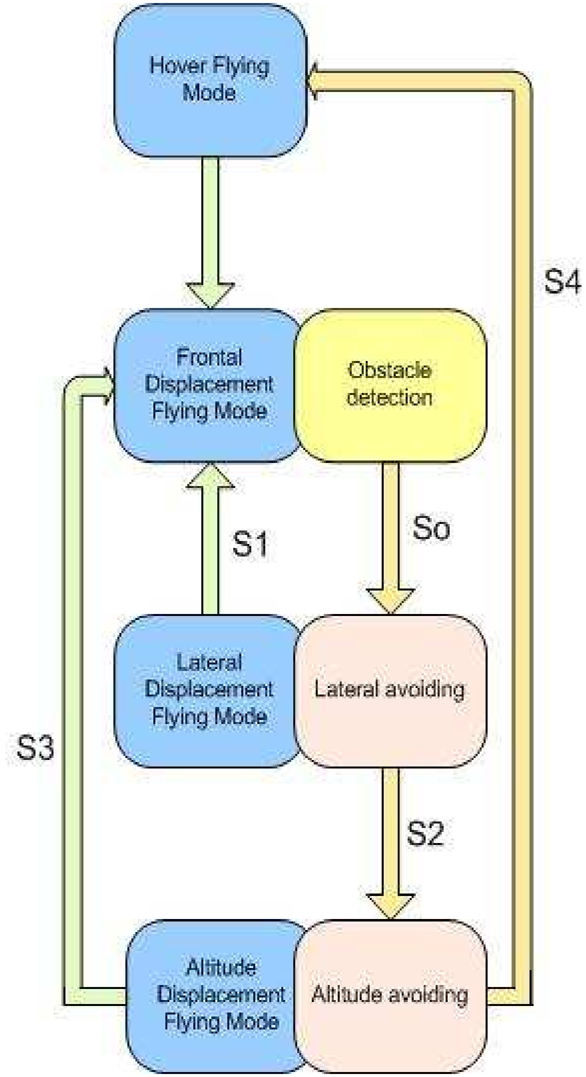
$$\bar{V}_{av}ref = -k^{\phi}_{av}\|\bar{V}^{f}_{OF_y} - WMA_{OF^{f}_{y}}\|$$

Example of controller

Control Strategy

4. State machine approach

- Frontal displacement is the default flying mode
- Frontal and lateral displacement are assured by the downward camera
- Obstacle detection, lateral and altitude avoidance are assured by the frontal camera
- Presence of alert signals

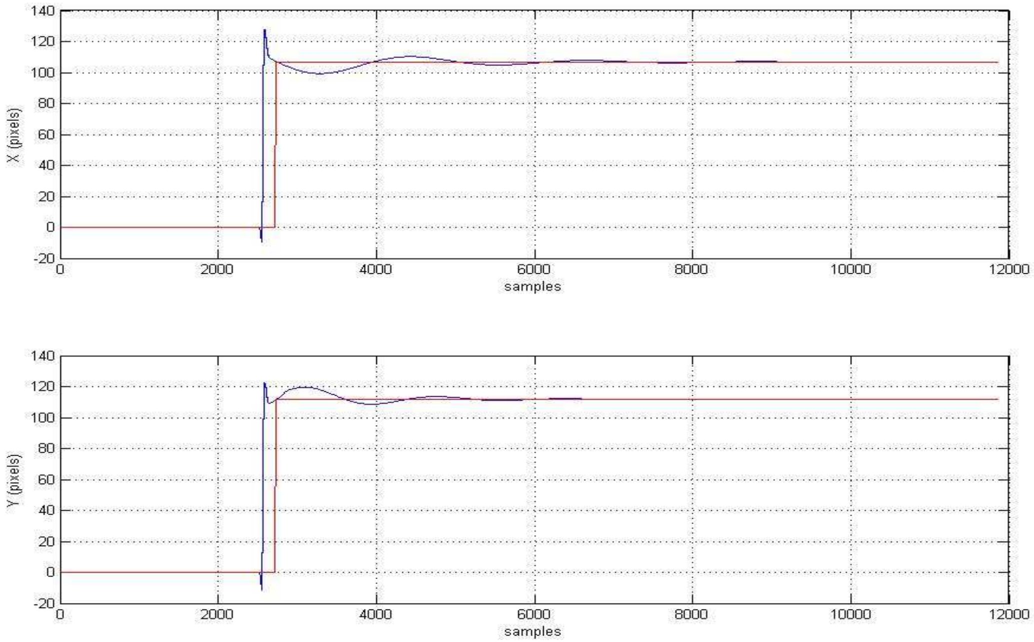


Example of controller

Results

Position Stabilization Simulations

- When Hover flying mode is activated, an area to track is selected, and the first center of the area is chosen as the reference position.

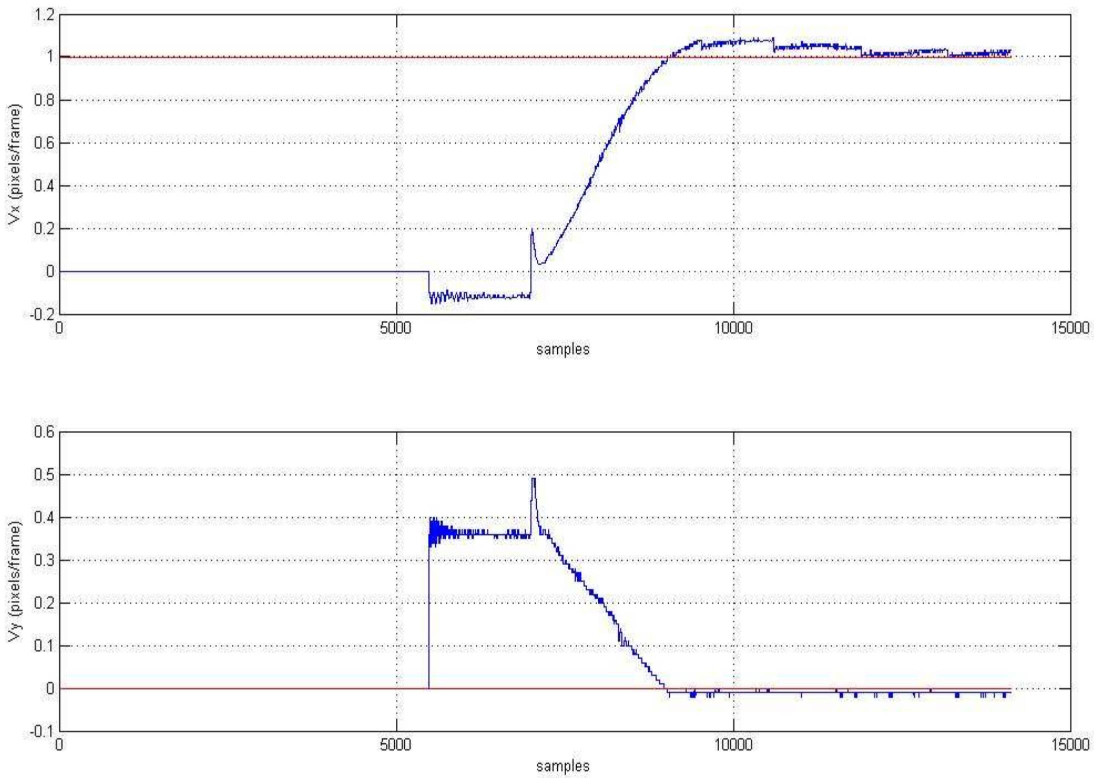


Position of the tracking in pixels

Results

Speed Regulation Simulations

- Optical flow regulation

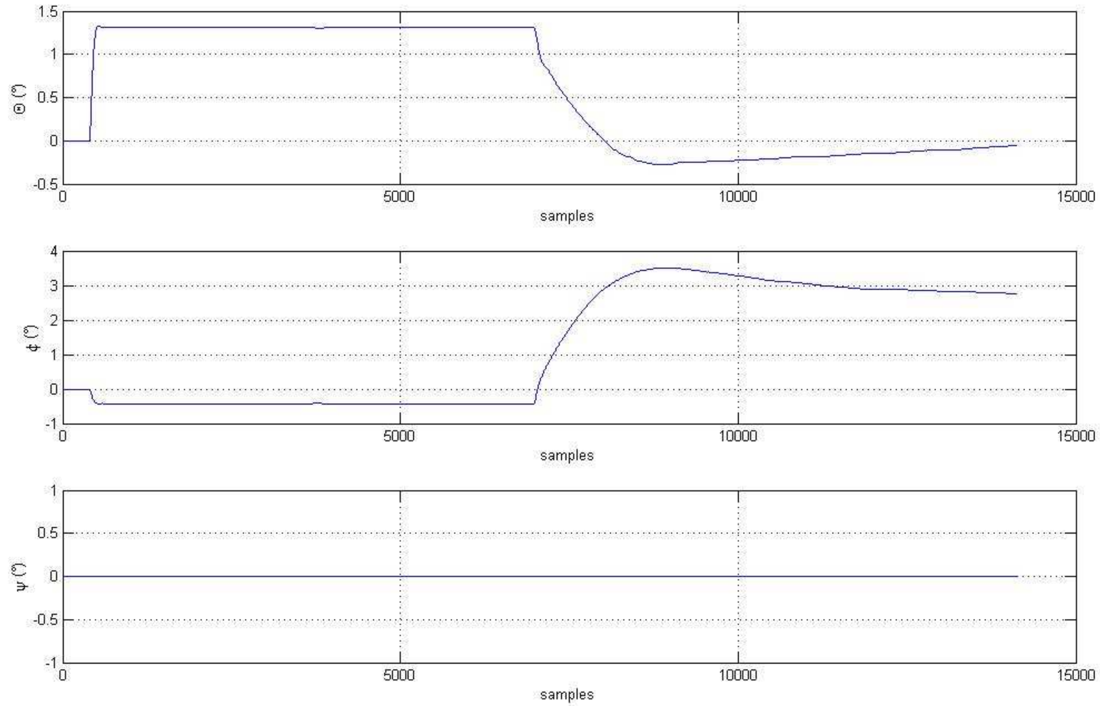


Optical flow regulation

Results

Speed Regulation Simulations

- Reference angles of the speed regulator



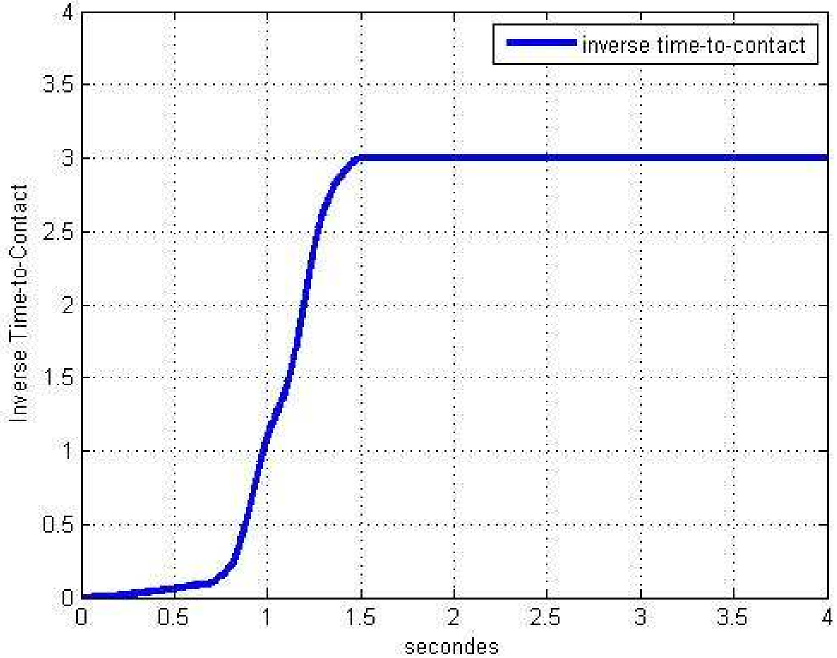
References Angles (Roll, Pitch,Yaw)

Results

Mobile robot

- The frontal camera subsystem measures the time to contact
- The time to contact is kept constant to maintain the vehicle stopped

Anticollision video



Time to contact

Simulations and videos

- Simulator
  - ➢ Stabilization
  - ➢ Regulation displacement
- Videos
  - ✓ Stabilization d'un quadrirotor
  - ✓ Stabilization using visual landmarks

Conclusion and future works

- Reactive navigation system using two orthogonal cameras
- New fusion method for optical flow computation
- Off-line validation of fusion algorithm with a real video
- First experimental results with a mobile car
- Simulations in 3D environment taking in consideration quad-rotor model and vision algorithm
- Current works consist in finishing implementation in the quad-rotor