Elective in Robotics

coordinator: Prof. Giuseppe Oriolo

Visual Hovering of a quadrotor aerial vehicle

(slides prepared by L. Rosa)

Dipartimento di Informatica e Sistemistica Antonio Ruberti

Sapienza Università di Roma

Task

Visual hovering: regulate the position of a quadrotor vehicle (in hovering conditions) above a selected target, using visual feedback

Motivations:

- GPS is unusable in indoor/cluttered environments
- Image analysis can provide a lot of information

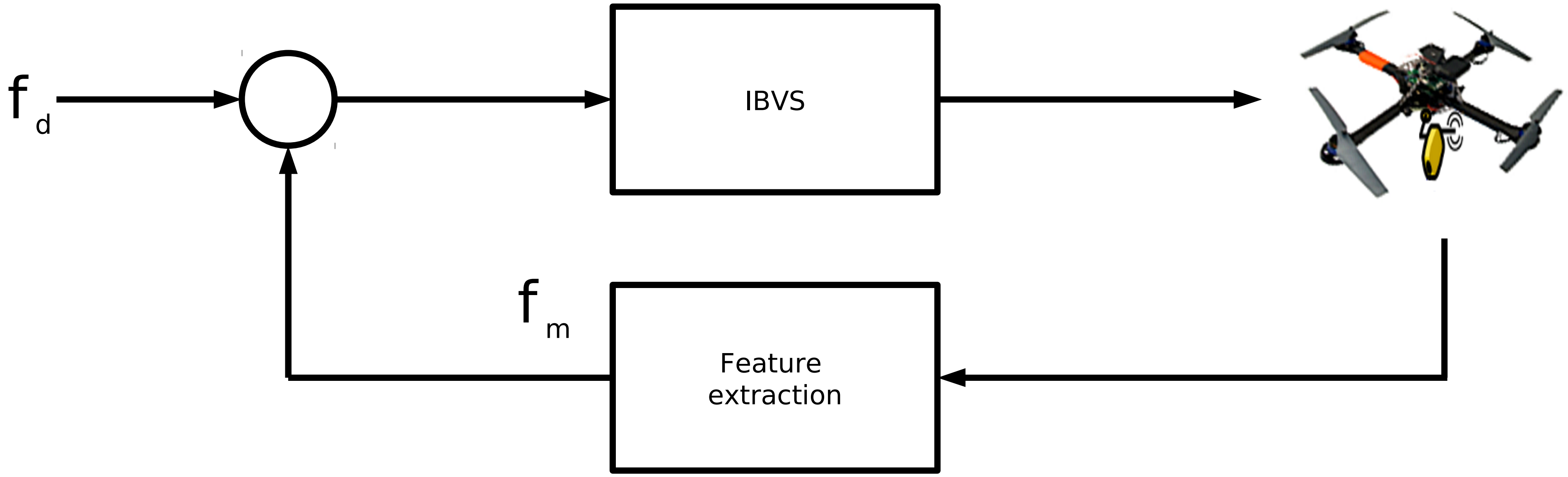
Note:

- Barometer is unreliable indoor
- Image analysis requires computationl power

Control scheme

Image Base Visual Servoing:

- Monocular camera (eye in hand configuration)
- IMU data (attitude/rotational velocities)



Visual error on image plane (tracking a desired target/object)

⇓

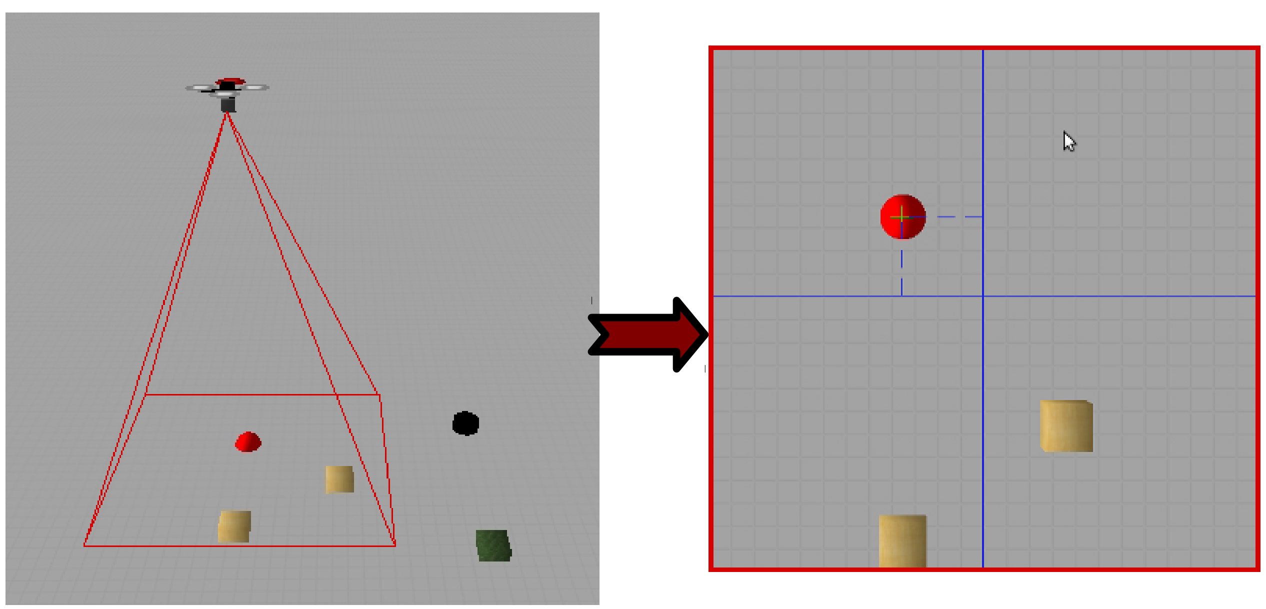
Do not need pose (and linear velocity) estimation

Reduced computation time

Feature exctraction and tracking

Two algoritms tested:

- Camshift: based on analysis of color probability distribution
  - Robust to occlusion
  - Suffers light intensity variations (also shadows)
- VISP (IRISA project): based on analysis of image moments
  - Tracks elliptical targets
  - Robust to occlusion
  - Robust to light intensity variations



We get the error (on the image plane) and its derivative

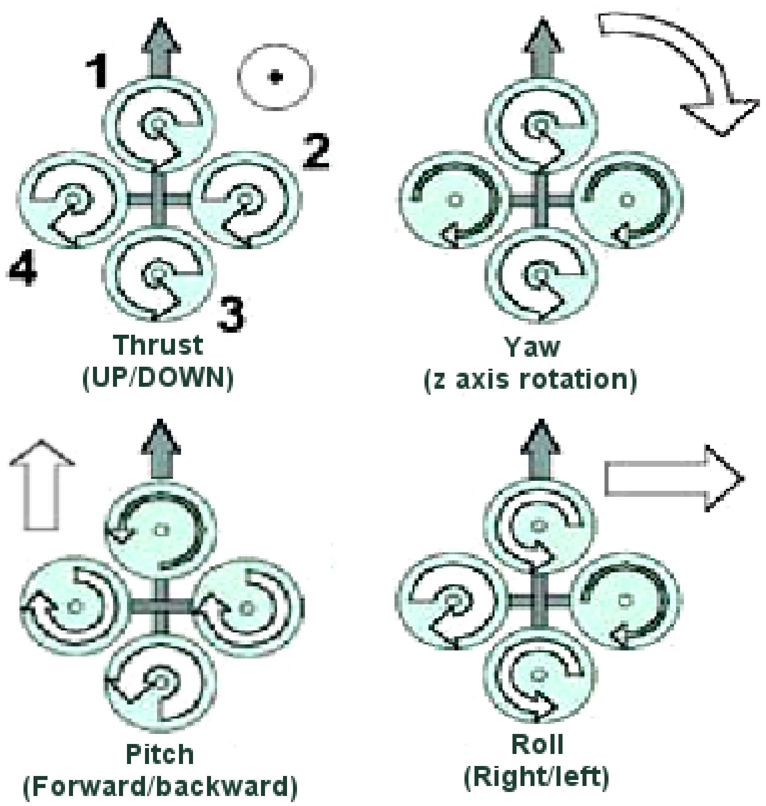
Note:

- The loss of the target leads to an undefined/wrong value
- Usually the reference is the center of the image (far from boundaries)

Quadrotor motion

The vehicle achieves motion by varying its attitude

Symmetric motion



Hypothesis 1: a “fast” innner control loop is performing attitude control

Desired attitude is realized as soon as the vehicle receives control data.

Dynamic model

To develop control law, we can use the following simplified model

$$\begin{cases} \ddot{x} = \dot{V}_x = (s_\theta c_\phi)\dfrac{T}{m} \\ \ddot{y} = \dot{V}_y = -s_\phi \dfrac{T}{m} \\ \ddot{z} = \dot{V}_z = g - (c_\theta c_\phi)\dfrac{T}{m} \end{cases}$$

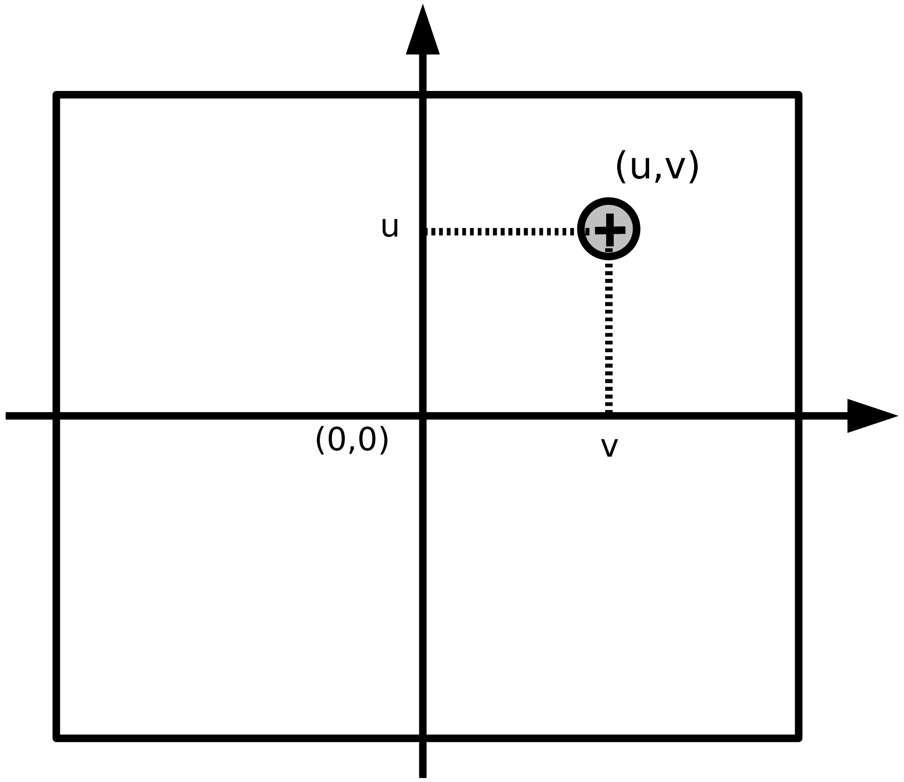
Hypotheses:

- Aerodynamic disturbances are neglected
- Inertial matrix is diagonal
- Camera center coincides with quadrotor center of gravity
- Yaw angle is zero
- Attitude controller is faster than visual control (1Khz vs 25Hz)

$$\begin{aligned} u_x &= s_\theta \\ u_y &= s_\phi \end{aligned}$$

Control inputs are the attitude references on roll and pitch

Controller IBVS



Relation between camera motion (6 degree of freedom) and feature motion (on image plane):

$$\begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = J_V V + J_\omega \Omega$$

Error on image plane (w.r.t. Image center)

$$e = \begin{bmatrix} e_u \\ e_v \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} - \begin{bmatrix} u \\ v \end{bmatrix}$$

Desired camera velocities

$$V_d = J_V^{\#} \left(-K \begin{bmatrix} u \\ v \end{bmatrix} - J_\omega \Omega \right)$$

$$\dot{e} = -(J_V V_d + J_\omega \Omega) = -Ke$$

Error exponentially converges to zero

Controller IBVS

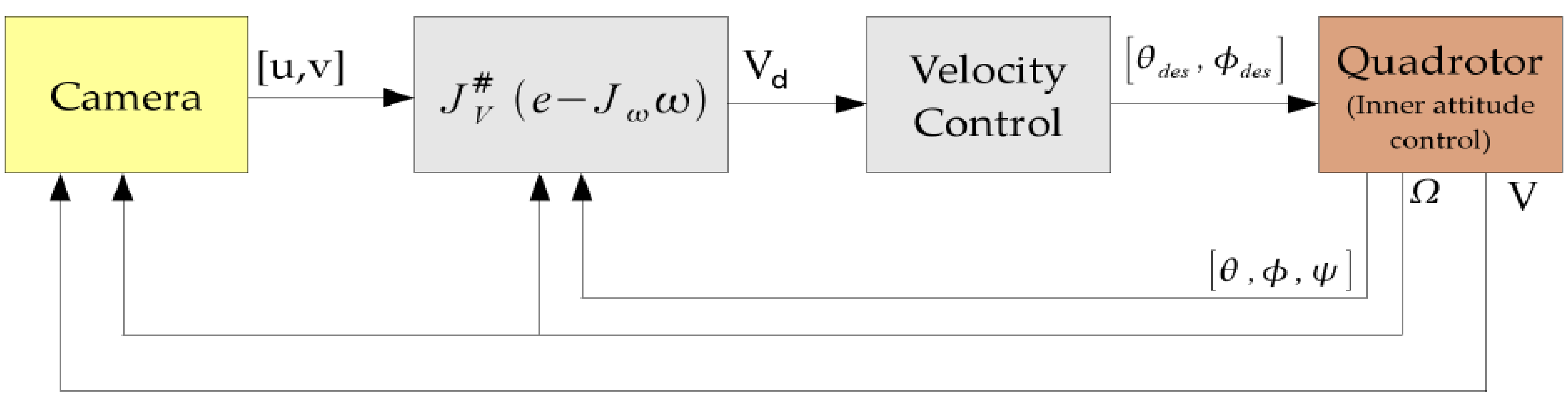
$$u_x = \frac{T}{m} K_a (V_{dx} - V_x)$$
$$u_y = \frac{T}{m} K_a (V_{dy} - V_y)$$

Control inputs

$$\begin{cases} \theta_d &= \arcsin\left(\dfrac{\frac{m}{T} K_a (V_{dx} - V_x)}{\cos\phi}\right) \\ -\phi_d &= \arcsin\left(\dfrac{m}{T} K_a (V_{dy} - V_y)\right) \end{cases}$$

Angular references

Control scheme



Main features

- Can be used in unknown environments
- No pose estimation is needed:
  - Linear velocities on the x-y plane are calculated from

$$\hat{V} = J_V^{\#} \left(\begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} - J_\omega \Omega \right)$$

Low computational cost

Could be implemented on embedded CPU (but requires image analysis)