IRISA

Part I:
Visual servoing:
concepts, recent results and applications

François Chaumette

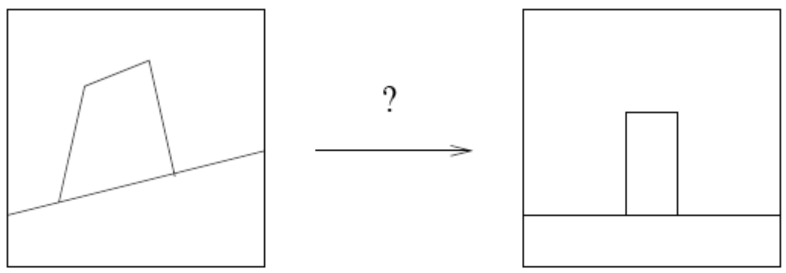
INRIA Rennes-Bretagne Atlantique - IRISA

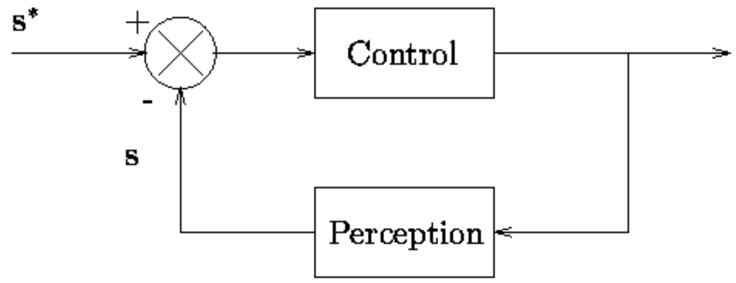
http://www.irisa.fr/lagadic

lagadic INRIA

What is visual servoing?

Visual servoing : vision-based control of a dynamic system





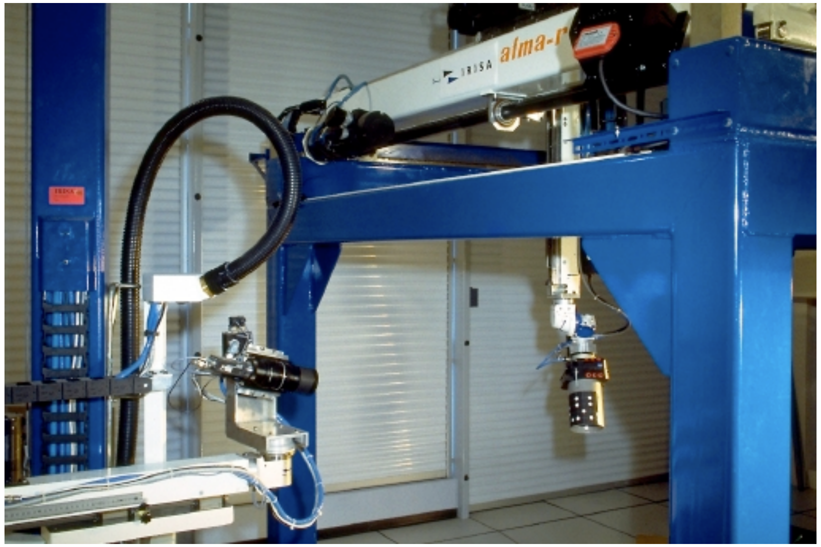
Necessary steps:

- design adequate visual features from the available measurements to control the dof required by the task
- design control schemes to regulate (s-s*) to 0 (or to minimize || s-s*||)
- taking into account the system and environment constraints
- image processing (initial matching and then real time tracking)

for an adequate system behavior (stability, robustness, …)

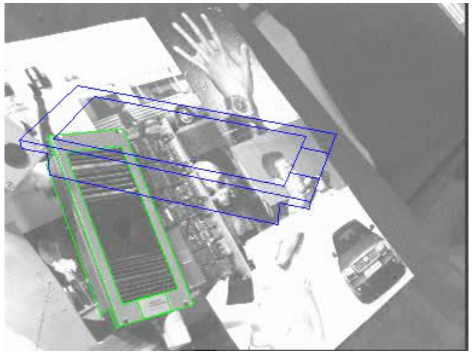
Possible configurations

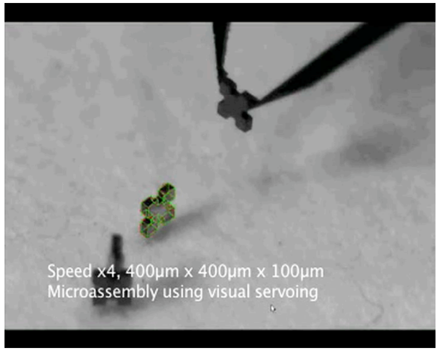
- Eye-in-hand system
- Eye-to-hand system
- Multi vision sensor possible but not necessary

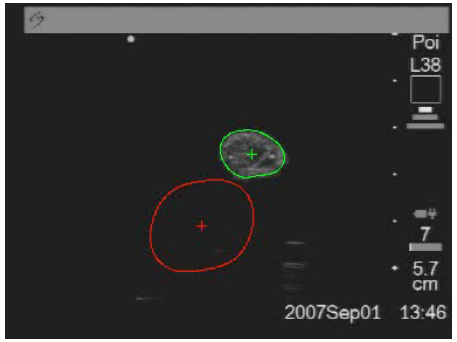


Possible systems: robot arm, pan-tilt unit, mobile vehicle, flying robot, underwater robot, micro systems, humanoid robot, etc.

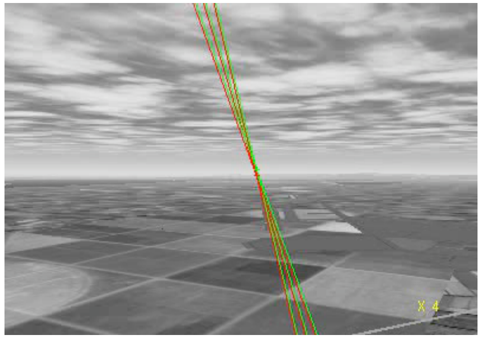
Examples

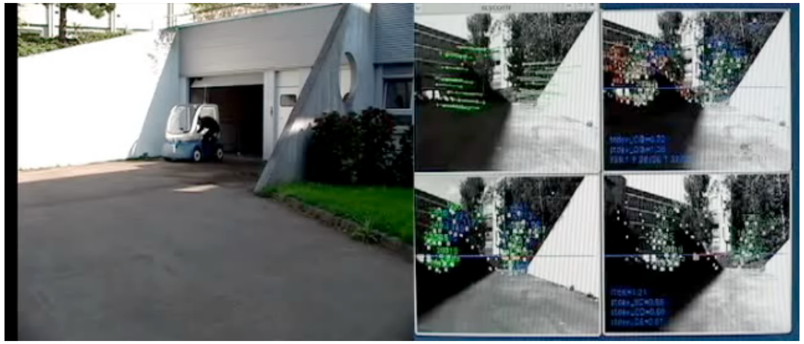


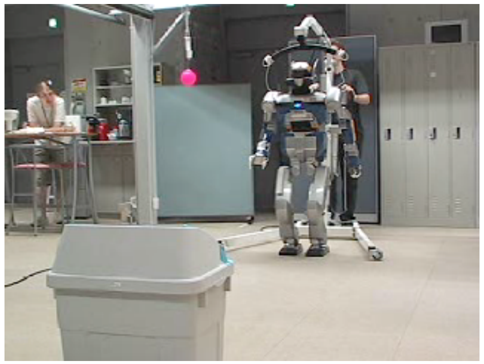

Speed x4, 400µm x 400µm x 100µm
Microassembly using visual servoing

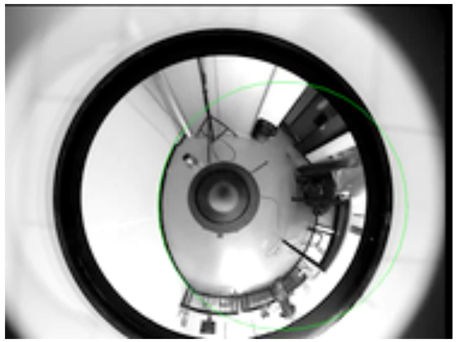

Poi
L38
7
5.7
cm
2007Sep01 13:46


X 4


X 4

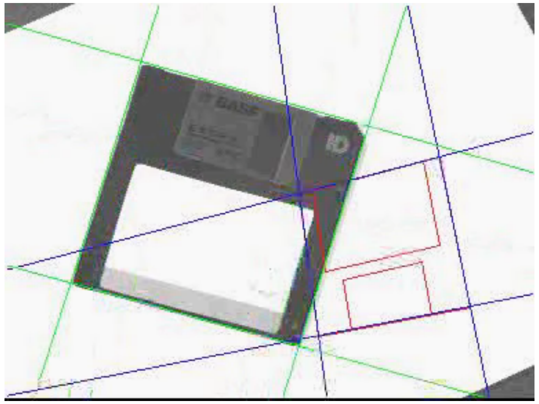






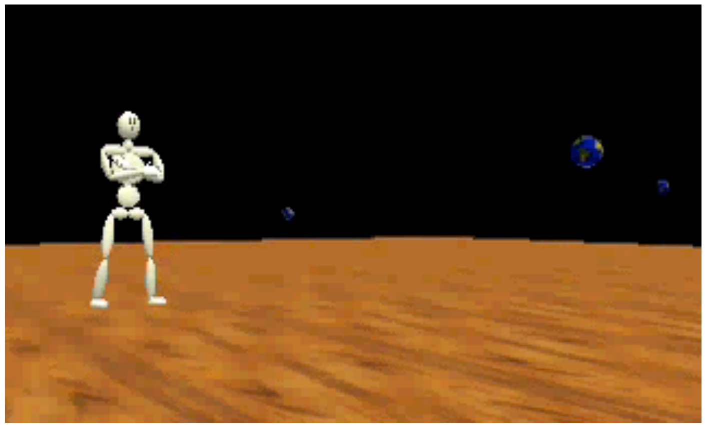
Other examples outside robotics

Virtual visual servoing for 3D localization in augmented reality



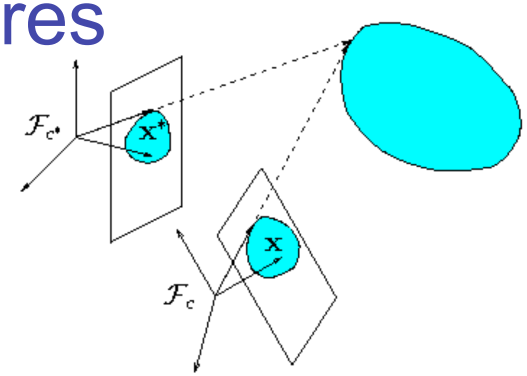
Visual servoing in computer animation





Recent results

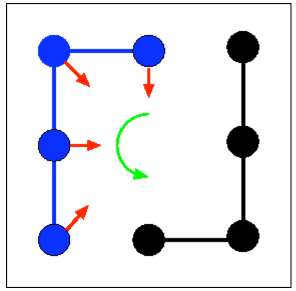
Modeling visual features



Modeling: $\mathbf{s} = \mathbf{s}(\mathbf{x}, \mathbf{a})$

Interaction: $\dot{\mathbf{s}} = \mathbf{L}(\mathbf{s}, \mathbf{a}, \mathbf{z})\ \mathbf{v}$

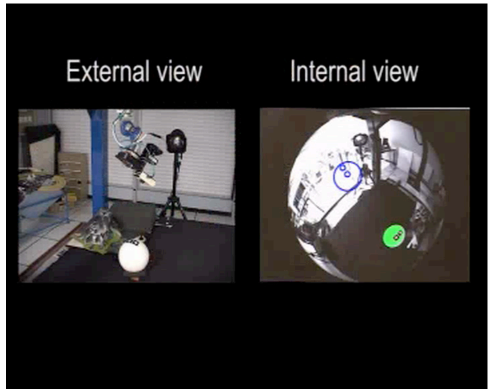
Control law: $\mathbf{v} = \mathbf{f}\left(\widehat{\mathbf{L}^{+}}\ (\mathbf{s} - \mathbf{s}^{*})\right)$



Basically a non linear control problem -> potential problems
(local minima, singularities, inadequate trajectories)

Objective: design the features so that it becomes
a linear control problem (as most as possible)

Contribution: modeling revisited using the spherical projection model



- nice invariance properties
- selection of optimal features for a marked sphere:
  IBVS GAS, robustness
- can be used for classical perspective cameras
  and omnidirectional vision sensors

Visual servoing based on image intensity

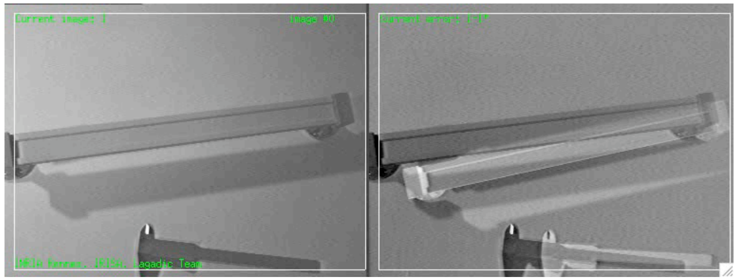
Goal : Using directly the intensity level of all pixels as input of visual servoing

Advantages :

- no image processing: neither features tracking nor matching
- excellent positioning accuracy

Problems :

- modeling the interaction between the intensity level and the 3D motion
  - Lambertian model and Blinn-Phong model to be robust to lighting variations and specularities
- corresponding Lyapunov function highly non linear



Scheme efficient for textured and non textured environments

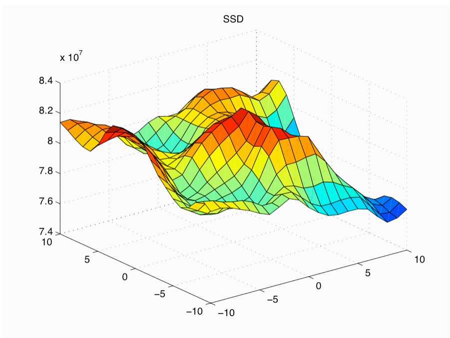
Visual servoing using mutual information

Still considering the image as a whole

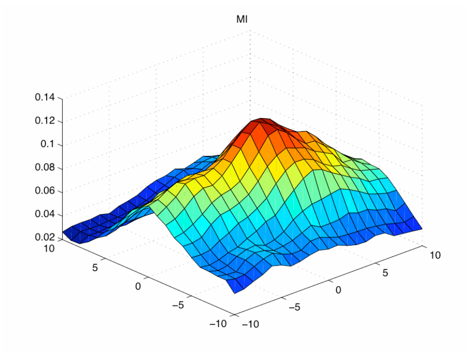
- ensuring robustness wrt perturbations (occlusion, light,…)
- using various image modalities

Approach

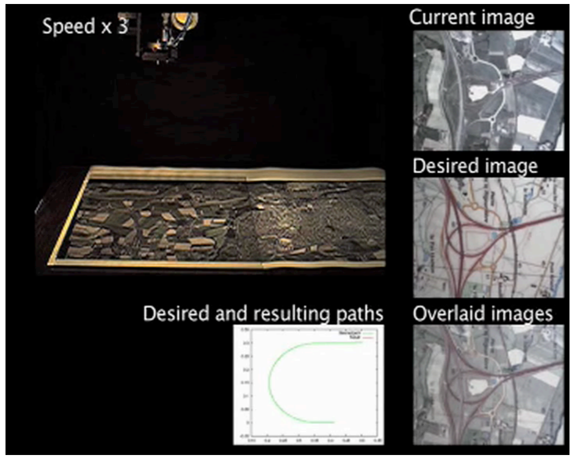
- using mutual information (based on the entropy)
- modeling the interaction between MI and motion parameters
- partial volume interpolation formulation for fast derivative computation



SSD



MI



Visual servoing for rotary wing aircrafts
(collaboration with CEA, I3S and ANU)

Homing and stabilization of a quadrotor (X4 flyer)

- Comparing a set of visual features and corresponding kinematics control laws
  - spherical coordinates of image centroid (passivity property)

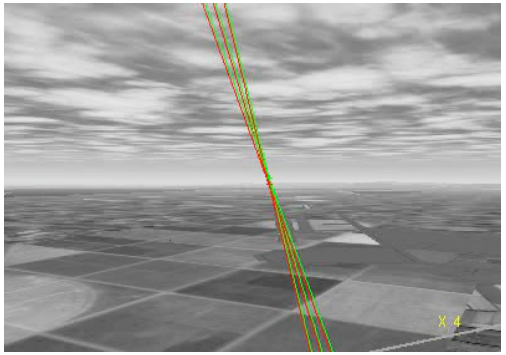
  - normalized area and centroid with perspective projection model

Personal conclusion: simple decoupled features are satisfactory

Visual servoing for fixed wing aircrafts

Automatic landing from the image of a runway (border and middle lines)

- dynamic model representative of a Falcon F7X (provided by Dassault Av.)
- planning image trajectories to be followed taking the aircraft dynamics into account
- decoupled lateral and longitudinal control law
- adequate visual features for control
  - vanishing point and lines orientation for lateral control
  - vanishing point, slope, aircraft velocity for longitudinal control (LQR)



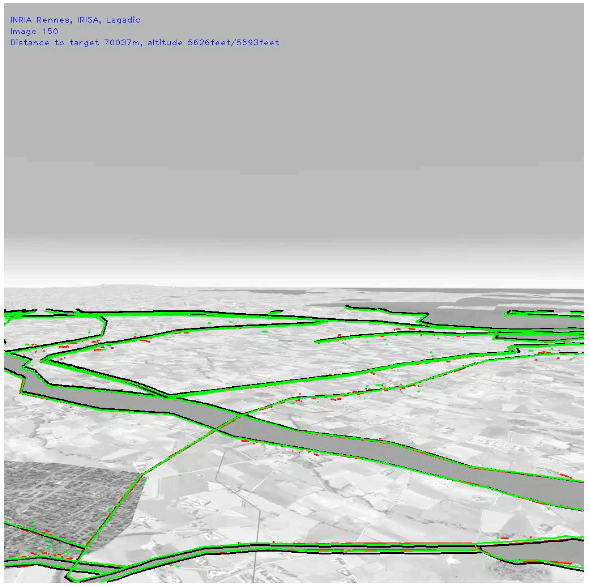
3D localization for aircrafts (FP6 Aerospace Pegase)

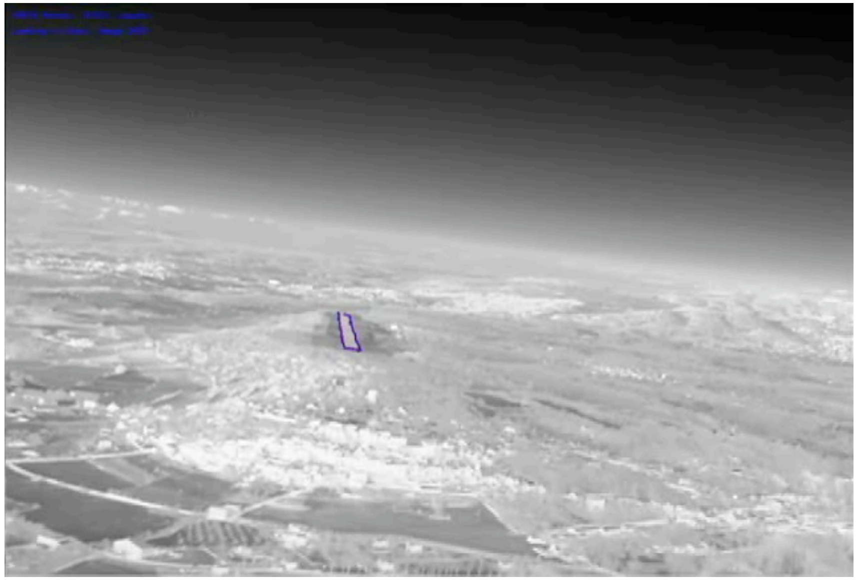
3D localization from a geographical database:

- set of 3D segments corresponding to roads, rivers, coasts, etc.

3D localization by virtual visual servoing:

- compute the camera pose so that the projection of the database fits with the edges in the current image





IR images provided by Thalès

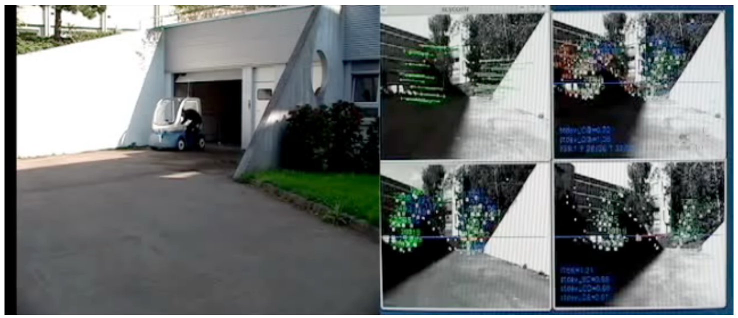
Autonomous navigation

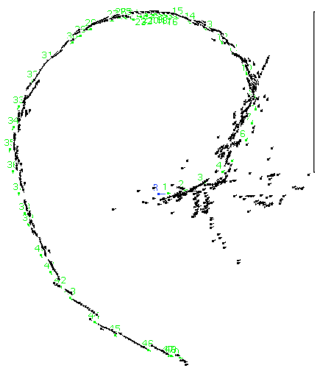
Classical approach:

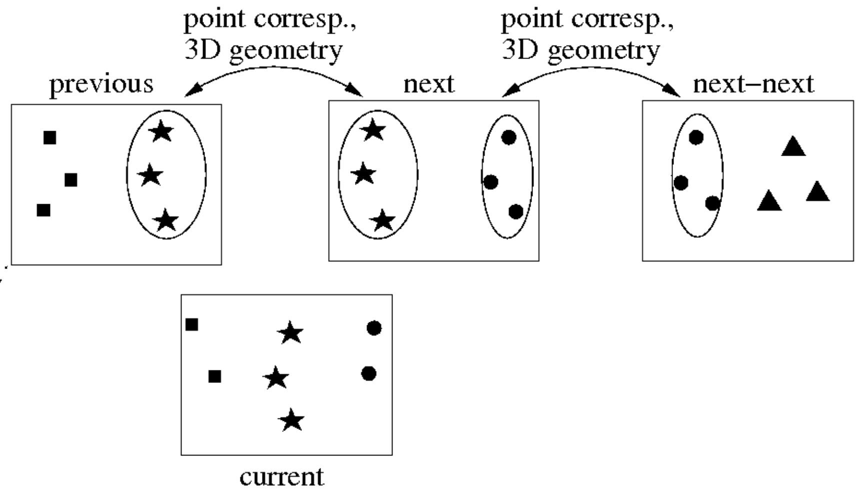
- teaching: global 3D reconstruction and accurate 3D localization (SLAM)
- following a specified 3D trajectory through accurate 3D localization

Approach developed: Accurate localization and mapping not mandatory

- teaching: topological description of the environment with key frames
- only local 3D reconstruction (points tracking and points transfer)
- navigation expressed as visual features to be seen (and not successive poses to be reached)
- simple IBVS for navigation







Part II:
Old results in
image motion-based visual servoing

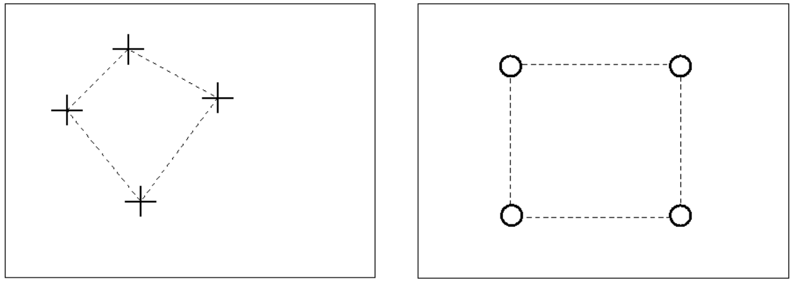
François Chaumette

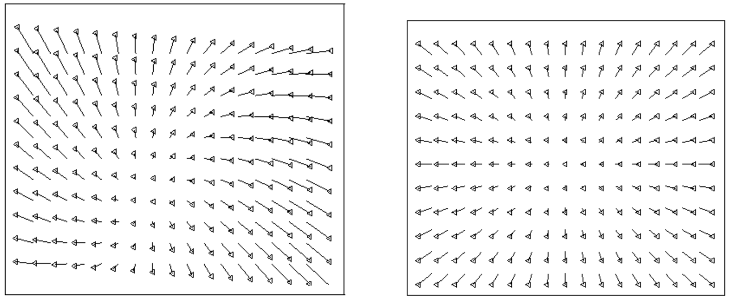
INRIA Rennes-Bretagne Atlantique - IRISA

http://www.irisa.fr/lagadic

Introduction

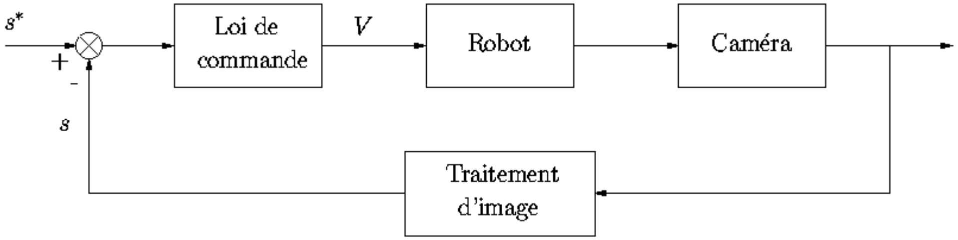
Goal: to control the motion of a dynamic system so that a desired motion is observed in the image.





Classical geometric VS

Image motion-based VS



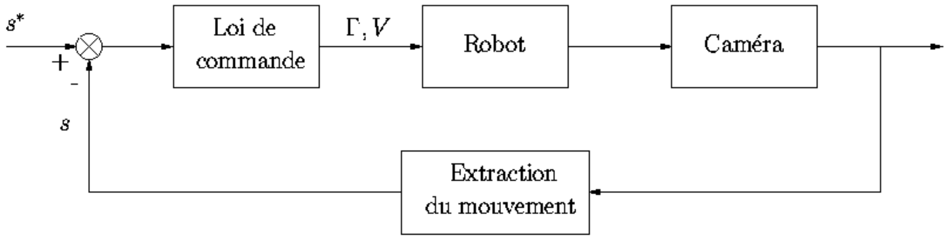


Image motion equations

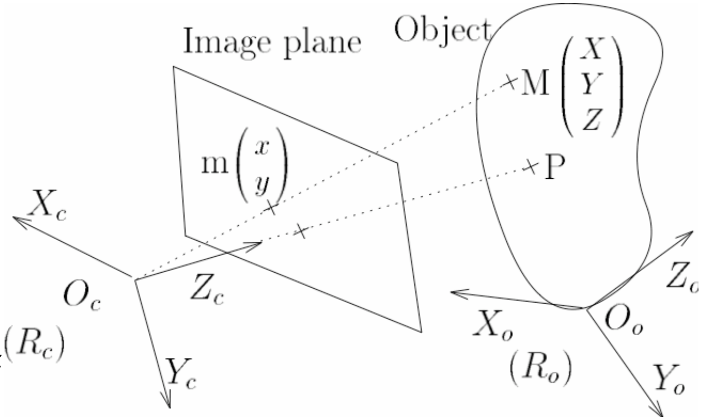


Image point: $x = X/Z \ , \ y = Y/Z$

Camera velocity: $\mathbf{v} = (V_x, V_y, V_z, \Omega_x, \Omega_y, \Omega_z)$

Image point velocity:

$$\begin{cases} \dot{x} = -V_x/Z + x\,V_z/Z + xy\Omega_x - (1+x^2)\Omega_y + y\,\Omega_z \\ \dot{y} = -V_y/Z + y\,V_z/Z + (1+y^2)\Omega_x - xy\Omega_y - x\,\Omega_z \end{cases}$$

If environment locally planar: $Z = Z_p + \gamma_1 X + \gamma_2 Y$

$$\forall (x,y) \begin{cases} \dot{x} = c_1 + a_1 x + a_2 y + q_1 x^2 + q_2 xy \\ \dot{y} = c_2 + a_3 x + a_4 y + q_2 y^2 + q_1 xy \end{cases}$$

with
$$\begin{cases} c_1 = v_x + \Omega_y & c_2 = v_y - \Omega_x \\ a_1 = -(\gamma_1 v_x + v_z) & a_3 = -\gamma_1 v_y + \Omega_z \\ a_2 = -(\gamma_2 v_x + \Omega_z) & a_4 = -(\gamma_2 v_y + v_z) \\ q_1 = \gamma_1 v_z + \Omega_y & q_2 = \gamma_2 v_z - \Omega_x \end{cases}$$
and $v_i = V_i / Z_p$

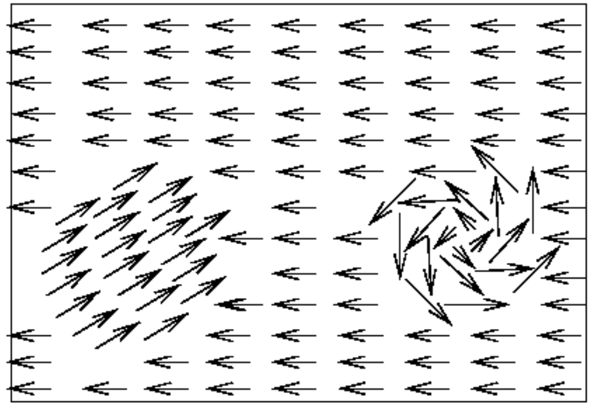
Image motion estimation

Classical and difficult problem widely studied in CV

Optical flow (= dense field) not so useful in robotics

More useful to estimate a few set of parameters

$$\forall(x,y) \begin{cases} \dot{x} = c_1 + a_1 x + a_2 y + q_1 x^2 + q_2 xy \\ \dot{y} = c_2 + a_3 x + a_4 y + q_2 y^2 + q_1 xy \end{cases}$$



It is possible to estimate (c_i, a_i, q_i) using for instance the RMR algorithm proposed in [Odobez, Bouthemy, CVIU, 1995], available on the web

- minimization of the SSD
- robust algorithm (M-estimator) to take into account outliers (non planar parts, moving objects) and detect them
- multi-resolution algorithm for fast computing

Image motion versus image displacement

For a planar object:

- quadratic image motion model

$$\forall(x,y)\begin{cases}\dot{x} = c_1 + a_1 x + a_2 y + q_1 x^2 + q_2 xy\\ \dot{y} = c_2 + a_3 x + a_4 y + q_2 y^2 + q_1 xy\end{cases}$$

from which $(V_i/Z_p, \Omega_i, \gamma_1, \gamma_2)$ can be estimated

- image displacement defined by an homography

$$\forall(x,y)\begin{cases}x_{k+1} = (h_{11}x_k + h_{12}y_k + h_{13})/(h_{31}x_k + h_{32}y_k + h_{33})\\ y_{k+1} = (h_{21}x_k + h_{22}y_k + h_{23})/(h_{31}x_k + h_{32}y_k + h_{33})\end{cases}$$

estimated for instance using the ESM algorithm [Benhimane, Malis, 2004]
from which $(\mathbf{R}, \mathbf{t}/d, \mathbf{n})$ can be estimated

Of course, same number of parameters: 8

Image motion VS versus 3D motion VS

Similar to 2D VS versus 3D VS

- (c_i, a_i, q_i) as inputs of the control scheme: image motion VS (d2D/dt VS)
- (v_i, Ω_i) as inputs of the control scheme: 3D motion VS (d3D/dt VS)

Example of image motion VS
[Crétual and Chaumette, IROS1997, IJRR 2001]

Orienting the camera so that it becomes parallel to a plane

- generally not possible using geometric visual servoing
- may be useful for landing a VTOL aircraft

$$Z = Z_p + \gamma_1 X + \gamma_2 Y$$

Task achieved when $\gamma_1 = \gamma_2 = 0$

From
$$\begin{cases} c_1 = v_x + \Omega_y & c_2 = v_y - \Omega_x \\ a_1 = -(\gamma_1 v_x + v_z) & a_3 = -\gamma_1 v_y + \Omega_z \\ a_2 = -(\gamma_2 v_x + \Omega_z) & a_4 = -(\gamma_2 v_y + v_z) \\ q_1 = \gamma_1 v_z + \Omega_y & q_2 = \gamma_2 v_z - \Omega_x \end{cases}$$

Task achieved when $\mathbf{s} - \mathbf{s}* = (q_1 - \Omega_y, q_2 + \Omega_x) - (0, 0) = (0, 0)$ if $v_z \neq 0$

Modeling and control scheme

$$\mathbf{s} = (q_1 - \Omega_y, q_2 + \Omega_x) \qquad \dot{\mathbf{s}} = \mathbf{L_s} \begin{bmatrix} \Omega_x \\ \Omega_y \end{bmatrix} + \mathbf{d_1} + \mathbf{d_2}$$

$$\mathbf{L_s} = \begin{bmatrix} 0 & -v_z \\ v_z & 0 \end{bmatrix} \qquad \mathbf{d_1} = \Omega_z \begin{bmatrix} s_2 \\ -s_1 \end{bmatrix} \qquad \mathbf{d_2} = (v_z - \gamma_1 v_x - \gamma_2 v_y)\mathbf{s}$$

Control scheme so that $\dot{\mathbf{s}} = -\lambda \mathbf{s}$

$$\begin{bmatrix} \Omega_x \\ \Omega_y \end{bmatrix} = -\widehat{\mathbf{L_s}}^{-1} (\lambda \mathbf{s} + \mathbf{d_1} + \widehat{\mathbf{d_2}})$$

$$\widehat{\mathbf{L_s}} = \begin{bmatrix} 0 & -\widehat{v_z} \\ \widehat{v_z} & 0 \end{bmatrix} \qquad \widehat{v_z} = (a_1 + a_4)/2 \qquad \widehat{\mathbf{d_2}} = \widehat{v_z}\ \mathbf{s}$$

- $(\Omega_x, \Omega_y, \Omega_z)$ have to be measured
- System LAS if and only if $\widehat{v_z}/v_z > 0$

Possible to add a fixation task ensuring $(c_1, c_2) = (0, 0)$ so that the same central point is always observed

Experimental results

Positioning the image plane parallel to a ham…

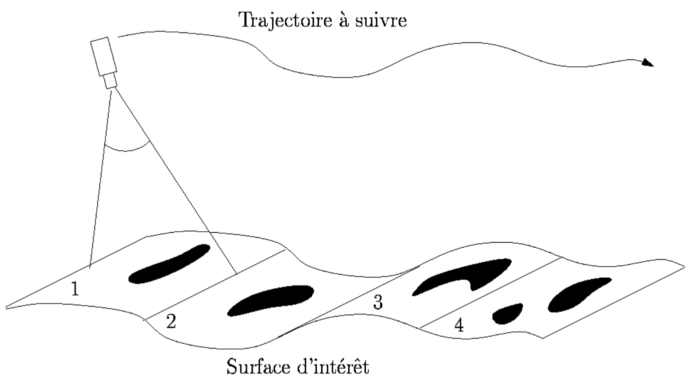
Other similar works

- [Sundareswaran, Bouthemy and Chaumette, IJRR, 1996]

  Alignment in the direction of translation of a mobile vehicle using a pan-tilt camera from
  - the coordinates of the focus of expansion
  - $(c_1 + \Omega_y, c_2 - \Omega_x)$

- [Crétual and Chaumette, IJRR, 2001]

  Following a trajectory above a locally planar environment



Conclusions

- These old works may be useful for vision-based control of aircrafts

- As in classical (geometric) visual servoing, it is possible to develop
  - image-based scheme (2D VS -> d2D/dt VS)
  - pose-based scheme (3D VS -> d3D/dt VS)

- Be careful in using the term “optical flow” (= dense image motion field), especially for the computer vision community.