Adaptive depth estimation in image based visual servo control of dynamic systems

Robert Mahony
Dep. Engineering,
Australian National University, Australia
Robert.Mahony@anu.edu.au

Coauthors

Peter Corke
ICT Centre,
CSIRO, Australia
Peter.Corke@csiro.au

Tarek Hamel
I3S UNSA-CNRS,
Universite de Nice,
Sophia Antipolis, France,
t.hamel@unice.fr

Motivation: A Typical Aerial Robotic System

Sensor systems:

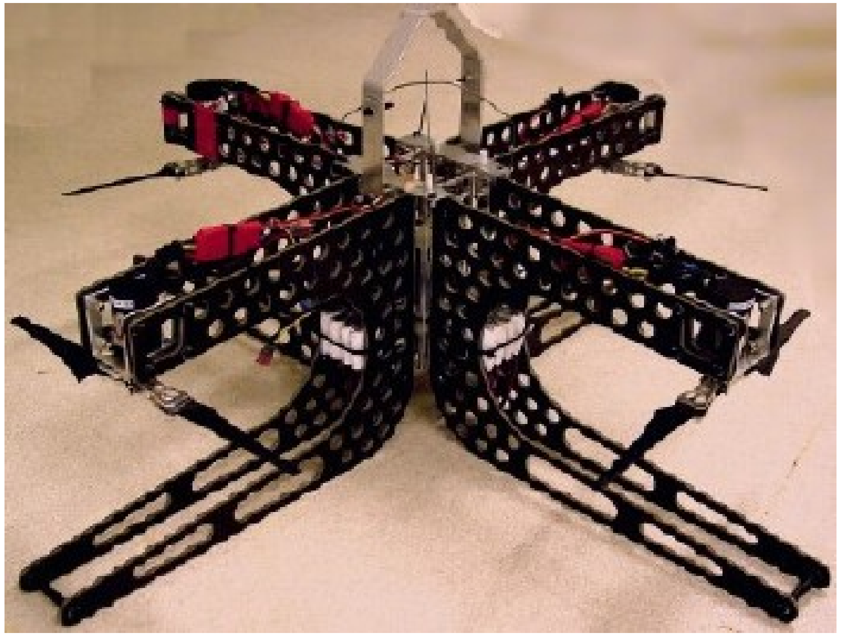
Inertial Measurement Unit (IMU)

- angular rates @ 70Hz
- accelerometers @ 70Hz
- magnetometers @ 70Hz

Fused to provide attitude @ 70Hz

Camera

- image features. @ 10-20Hz
- optic flow. @ 10-20Hz



- ➤ Development of commercial systems requires development of robust control for a range basic tasks.
- ➤ In this paper we consider stabilisation of the vehicle in hover over an observed visual target.

Alternative Control Architectures

Classical Estimation and Control:

1. State space model of the system.
2. Control task is expressed in terms of system state.
3. Estimator or filter ‘maps’ sensor measurements to a state estimate.
4. Controller algorithm ‘maps’ state estimates to actuator inputs.

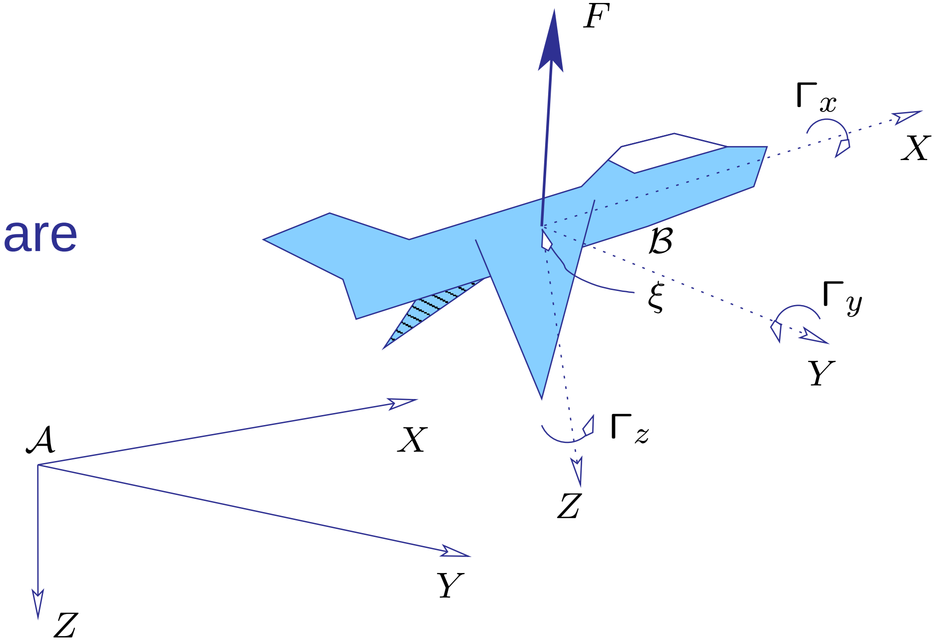
Sensor Based Control: Image Based Visual Control (IBVS)

1. A model of the dynamic response of the observed sensor response.
2. The control task is expressed in terms of the sensor measurements.
3. A control design ‘maps’ sensor measurements to actuator inputs.

Dynamics of a flying brick

$$\dot{\xi} = RV$$
$$m\dot{V} = -m\Omega_\times V + F$$
$$\dot{R} = R\Omega_\times,$$
$$\mathbf{I}\dot{\Omega} = -\Omega_\times \mathbf{I}\Omega + \Gamma.$$

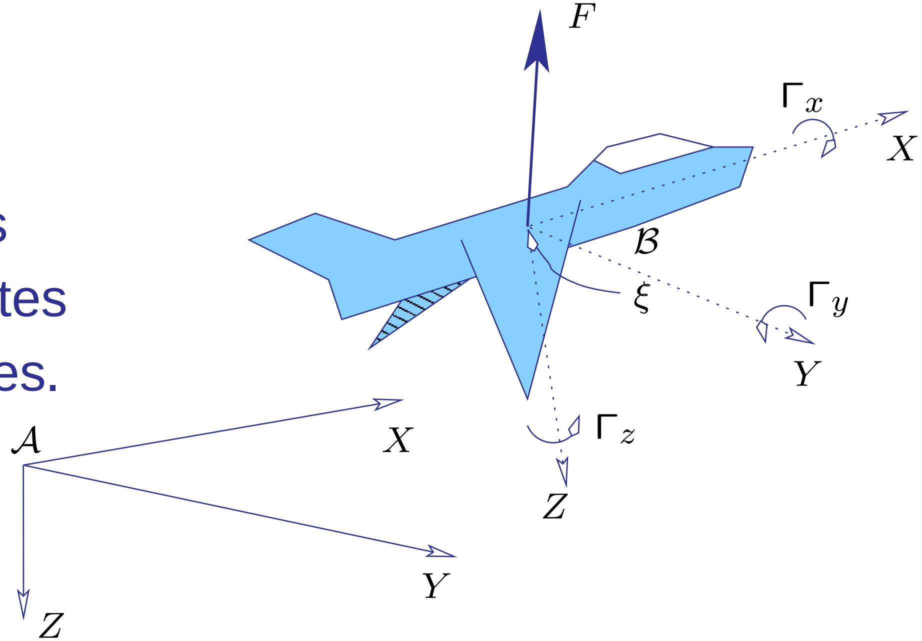
The exogenous force and torque are denoted F and Γ.



Simplified dynamics of a flying brick

$$\dot{\xi} = RV$$
$$m\dot{V} = -m\Omega_{\times} V + F$$

Consider the translation dynamics independently with the attitude states entering as exogenous disturbances.

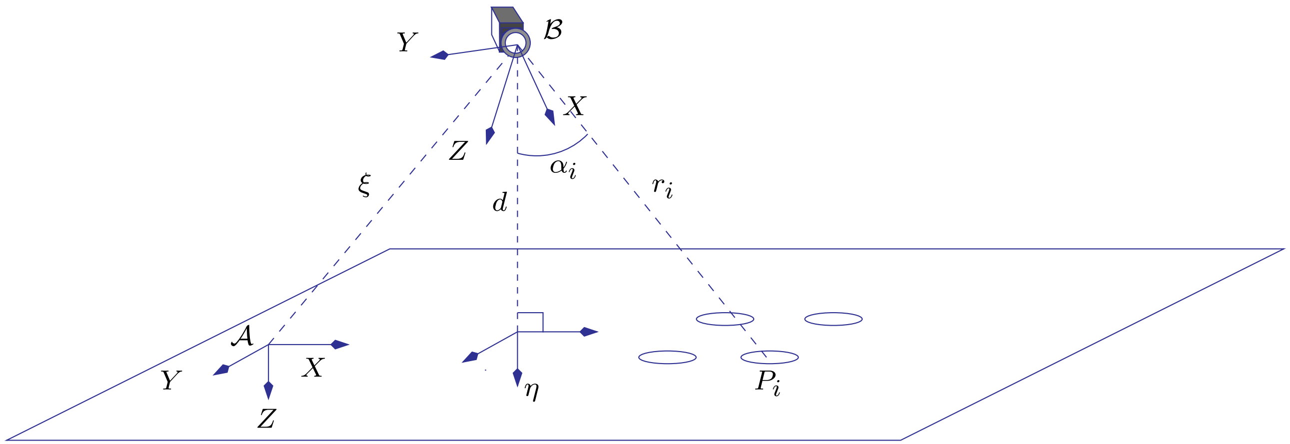


Visual features

▶Target consists of an ensemble of point targets on a flat target plane.

▶Use a spherical projection of visual data

$$p_i = \frac{P_i}{|P_i|} = \frac{1}{r_i} P_i.$$



▶From geometry one has

$$r_i = \frac{d}{\cos(\alpha_i)}$$

d is height of camera from target plane, α_i is angle of observed target from target plane normal.

Feature Kinematics

Image point kinematics

$$\dot{p}_i = -\Omega_\times p_i + \left(I_3 - p_i p_i^T\right) \cos(\alpha_i) \frac{V}{d}.$$

Image feature

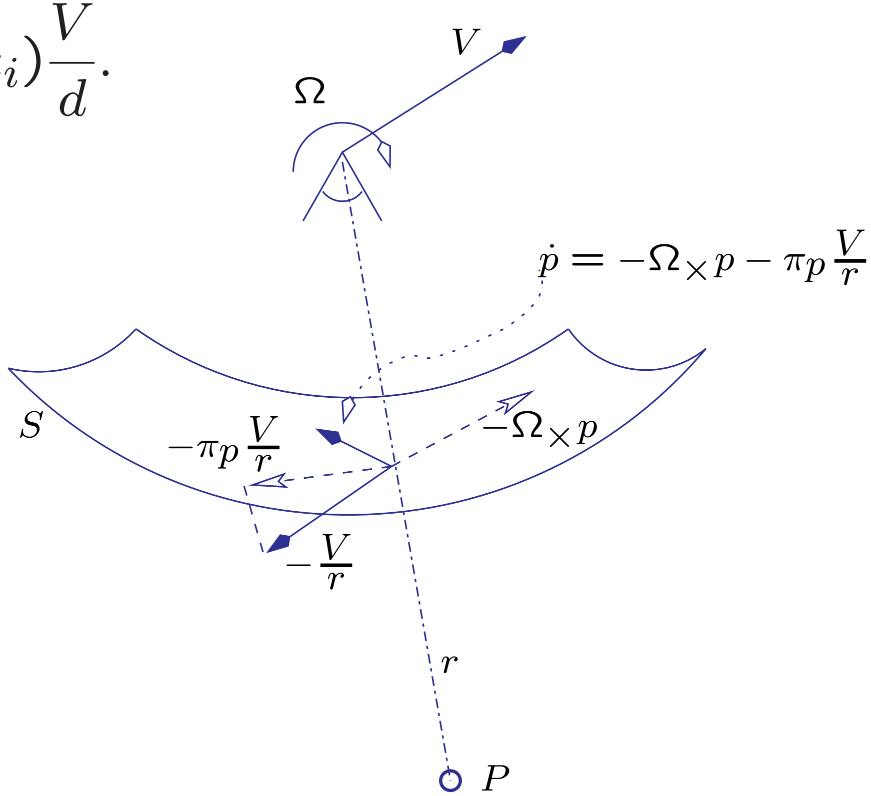
$$q := \sum_{i=1}^{n} p_i \in \mathbb{R}^3$$

Feature kinematics

$$\dot{q} = -\Omega_\times q - Q \frac{V}{d(t)},$$

where

$$Q = \sum_{i=1}^{i=n} \cos(\alpha_i) \pi_{p_i}.$$



Note that $Q > 0$ is positive definite and $\Omega_\times$ is skew-symmetric.

Image Error Kinematics

Goal vector q^* (set point fixed in inertial frame)

$$\dot{q}^* = -\Omega \times q^*.$$

Image error

$$\delta := q - q^*.$$

The image error kinematics are

$$\dot{\delta} = -\Omega_\times \delta - QW$$

Where W denotes the *feature velocity*

$$\boxed{W = \frac{V}{d(t)}}$$

feature velocity

Optic Flow as velocity measure

The feature velocity W is related to a measure of optic flow computed over entire target plane.

Define the total target plane optic flow to be

$$\phi := \int_{\underline{S}^2_\eta} \dot{p} dp \qquad = \int_{\underline{S}^2_\eta} \left(-\Omega_\times p_i + \left(I_3 - p_i p_i^T \right) \cos(\alpha_i) \frac{V}{d} \right) dp$$

For $\underline{S}^2_\eta$ the lower hemisphere around the pole normal to the target plane η.

The image feature velocity can be computed as

$$\boxed{W = \frac{4}{\pi} \left(I_3 - \frac{1}{2} \eta \eta^T \right) (\phi + 2\pi \Omega_\times \eta)}$$

Alternatively, the feature velocity can be directly calculated as differentiation of image feature

$$W = -Q^{-1} (\dot{q} + \Omega \times q)$$

Feature based dynamics

Differentiate W to compute the feature dynamics

$$\begin{aligned}\dot{W} &= \frac{\dot{V}}{d} - \frac{V\dot{d}}{d^2} \\ &= -\Omega \times W + W\langle W, \eta\rangle + \frac{1}{d(t)}\frac{F}{m}\end{aligned}$$

Where

$$\dot{d} = -\langle \eta, V\rangle = -d\langle \eta, W\rangle$$

Image based representation of full dynamics

$$\dot{\delta} = -\Omega_{\times}\delta - QW \qquad \text{Kinematics}$$

$$\dot{W} = -\Omega \times W + W\langle W, \eta\rangle + \frac{F}{md} \qquad \text{Dynamics}$$

$$\dot{d} = -d\langle \eta, W\rangle \qquad \text{Depth Kinematics}$$

Adaptive control for high frequency gain with integrable dynamics

Let $\hat{d}$ be an internal controller state with dynamics

$$\dot{\hat{d}} = -\hat{d}\langle \eta, W \rangle$$

these dynamics act like *virtual* depth dynamics

Then

$$\frac{d}{dt}\left(\frac{\hat{d}}{d}\right) = \frac{\dot{\hat{d}}}{d} - \frac{\hat{d}\dot{d}}{d^2} = -\frac{\hat{d}\langle \eta, W \rangle}{d} + \frac{\hat{d}d\langle \eta, W \rangle}{d^2} = 0$$

Integrating this relationship over time one obtains

$$\frac{\hat{d}(t)}{d(t)} = a = \frac{\hat{d}(0)}{d(0)}$$

where a is an unknown constant depending on the error between true depth $d(0)$ and the initial estimate $\hat{d}(0)$.

Image kinematics for the flying brick

Image based representation of full dynamics

$$\dot{\delta} = -\Omega_{\times}\delta - QW \qquad \text{Kinematics}$$

$$\dot{W} = -\Omega \times W + W\langle W, \eta\rangle + a\frac{F}{m\hat{d}(t)} \qquad \text{Dynamics}$$

$$\dot{\hat{d}} = -\hat{d}\langle \eta, W\rangle \qquad \text{Virtual Depth Kinematics}$$

- The gain a is an unknown high frequency gain.

  This will be dealt with using an adaptive parameter estimation.

- The virtual depth dynamics are an additional dynamic state in the control design. The role is to invert the time-varying response of true depth $d(t)$ so that the adaptation can be undertaken in the space of initial conditions.

Adaptive control design

Let $\rho = 1/a$ and let $\hat{\rho}$ be an estimate for $\rho = 1/a$.

Let $k, c, \lambda > 0$ be positive gains.

Let $\epsilon = W - k\delta$.

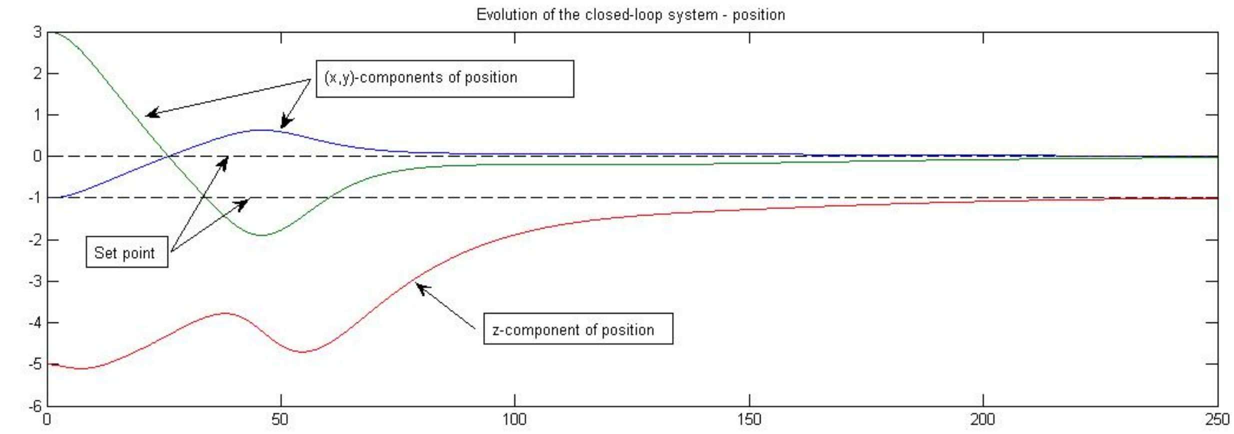
The control algorithm is computed according to the following dynamics

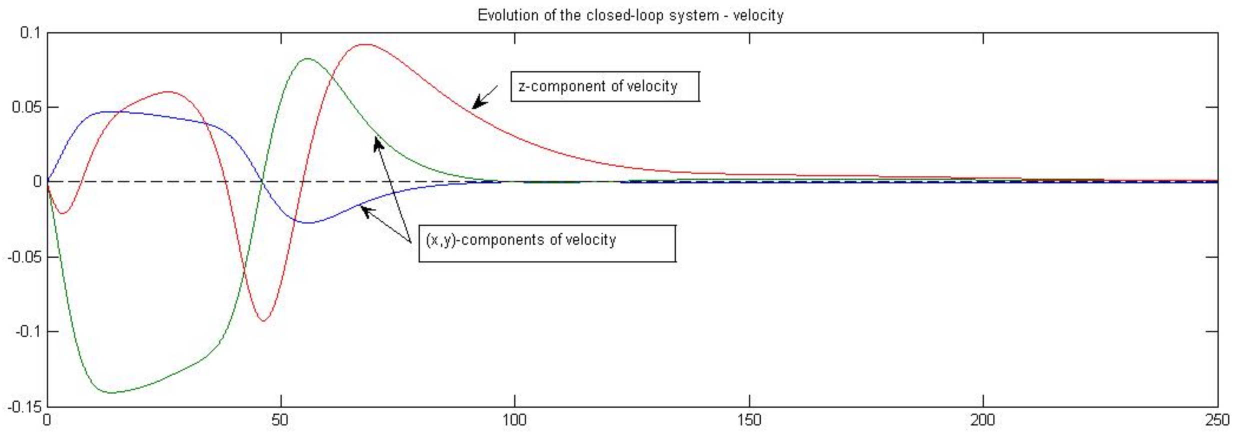
$$
\begin{aligned}
\dot{\hat{d}}(t) &= -\hat{d}\langle \eta, W \rangle, & \hat{d}(0) &= \hat{d}_0, \\
\dot{\hat{\rho}} &:= -\lambda \langle U, \epsilon \rangle, & \hat{\rho}(0) &= 1, \\
U &:= Q\delta - W\langle W, \eta \rangle - kQW - c\epsilon, \\
F(t) &:= \hat{\rho} m \hat{d}(t) U,
\end{aligned}
$$

Control design is based on a control Lyapunov function argument

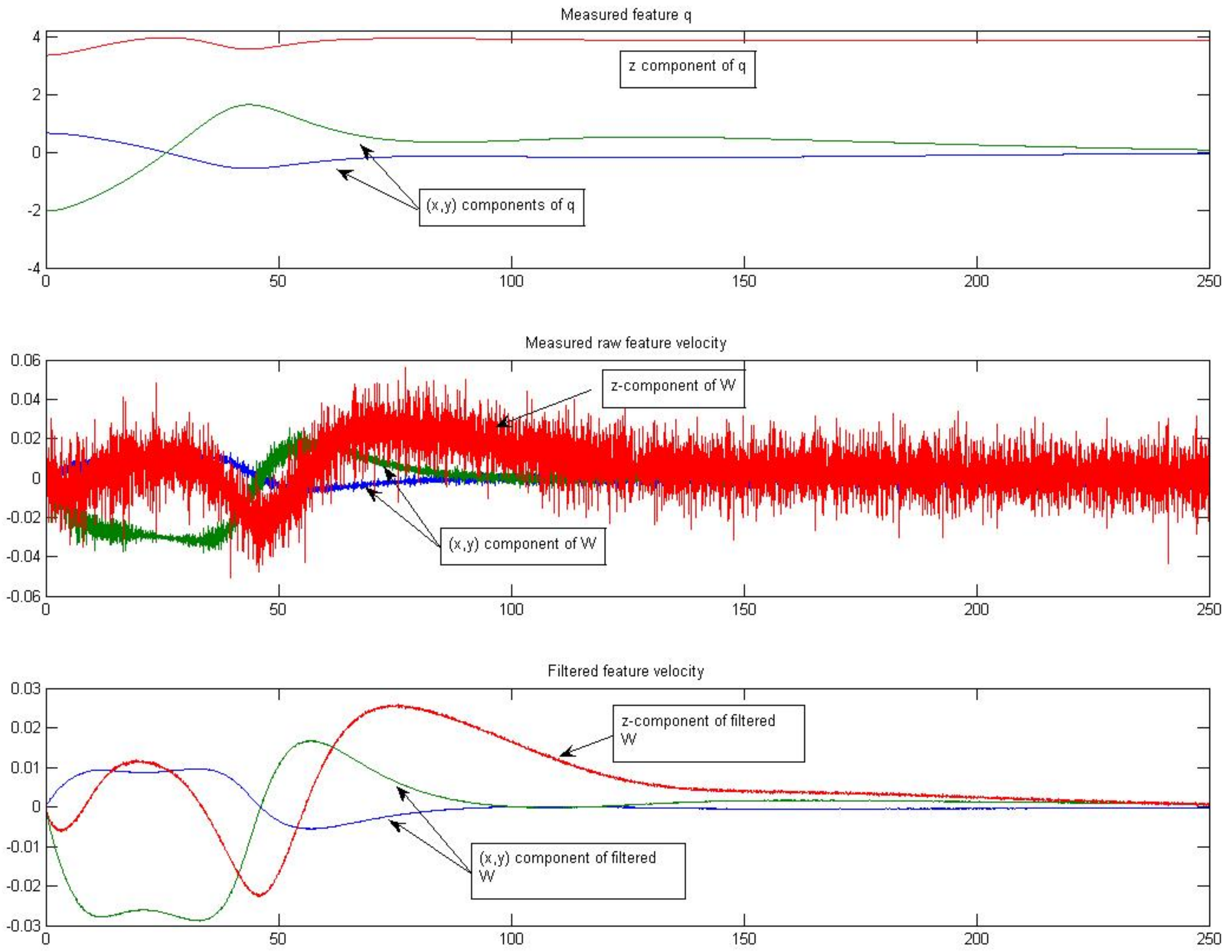
$$
\mathcal{L} = \frac{1}{2}|\delta|^2 + \frac{1}{2}|\epsilon|^2 + \frac{a}{2\lambda}|\tilde{\rho}|^2, \qquad \dot{\mathcal{L}} = -|\delta|^2 - |\epsilon|^2
$$

Simulation - State evolution

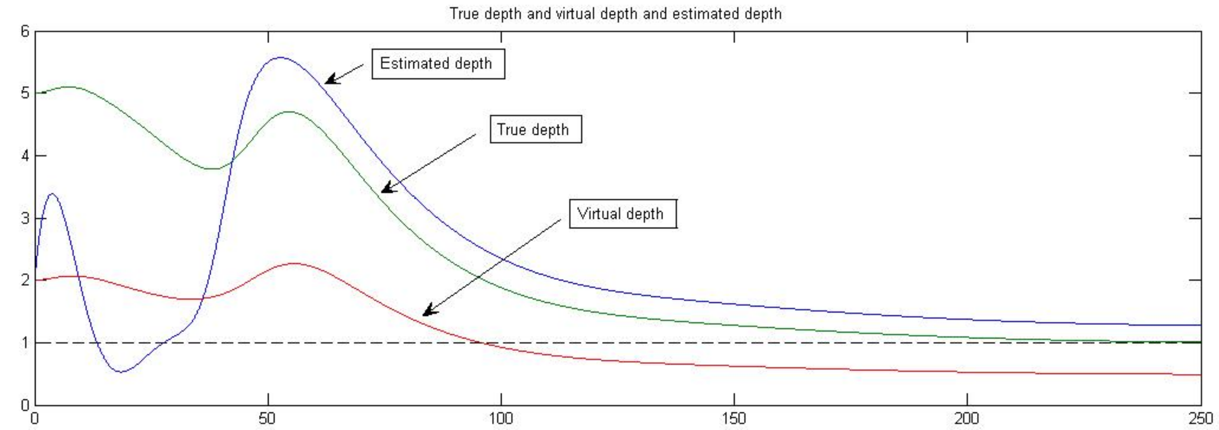


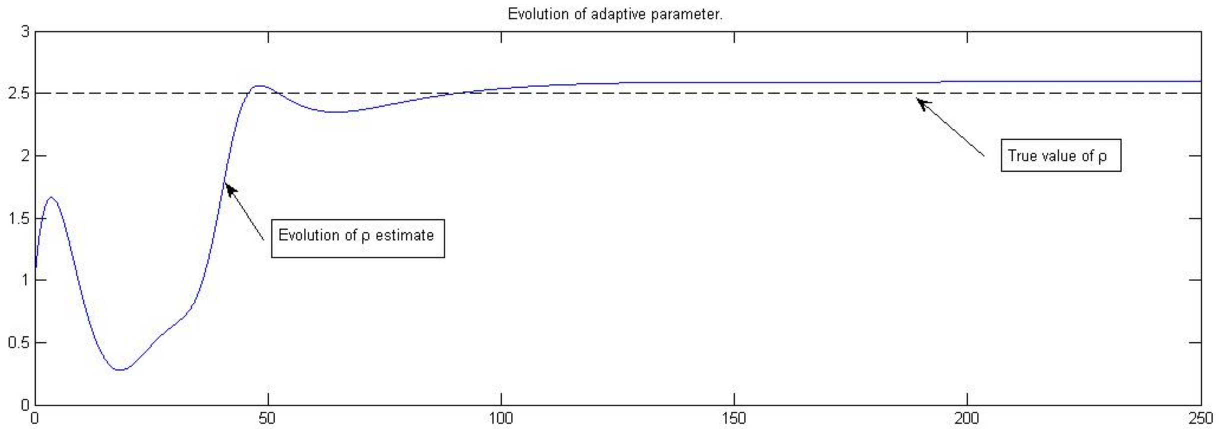


Simulation - Visual data



Evolution of internal controller state





Conclusions

- Very careful choice of visual features.
  - Spherical projection
  - Averaging of image data into a single image feature such that the image Jacobian is square.
- Use the most robust direct measure of optic flow that can be computed.
- Dynamic inversion of unknown depth dynamics in controller structure to reduce adaptive estimation of unknown depth to an estimation of unknown initial conditions.
- Constructive non-linear control design.

There is still a long way to go in developing practical dynamical IBVS algorithms.

Work presented was undertaken in collaboration with

PETER CORKE

Senior Principal Research Scientist,

Autonomous Systems Laboratory Director,

Robotics.,

CSIRO ICT Centre, Brisbane.

TAREK HAMEL

Professor, I3S-CNRS,

Nice-Sophia Antipolis, France.