Universidad de Zaragoza

Control de robots y sistemas multi-robot basado en visión

Ciclo de conferencias
Master y Programa de Doctorado en "Ingeniería de Sistemas y de Control"

UNED – ETS Ingeniería Informática

April -2014

Colaboradores:
Gonzalo López Nicolás
Héctor Manuel Becerra
Rosario Aragüés
Eduardo Montijano
Miguel Aranda

Carlos Sagues
Universidad de Zaragoza
http://www.unizar.es/~csagues

Motivation





















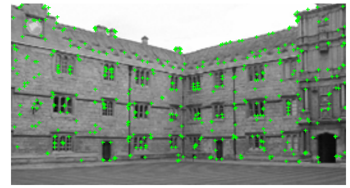
Visitas a la Catedral: acceso por C/ Palafox
Acceso reservado para actos de culto

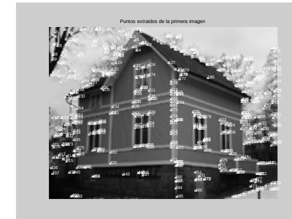
Index

- Features. FM, H, TT (Fundamental Matriz, Homography and Trifocal Tensor)
- Visual mobile robot control
  - FM based
  - H based
  - TT based
  - Long term navigation
- Control of Multi-robot systems
  - Data association
  - Coordinated motion with epipoles
  - Central decision with flying camera on scene - Homography

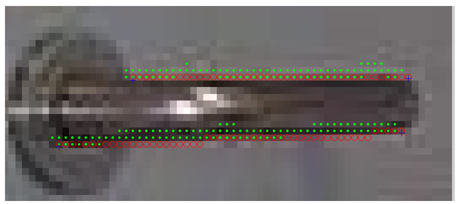
Features

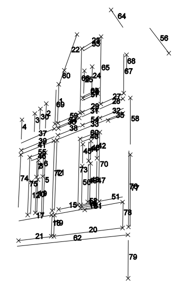
- Harris corner extractor





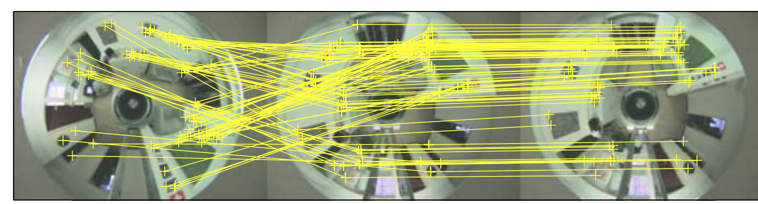
- Lines



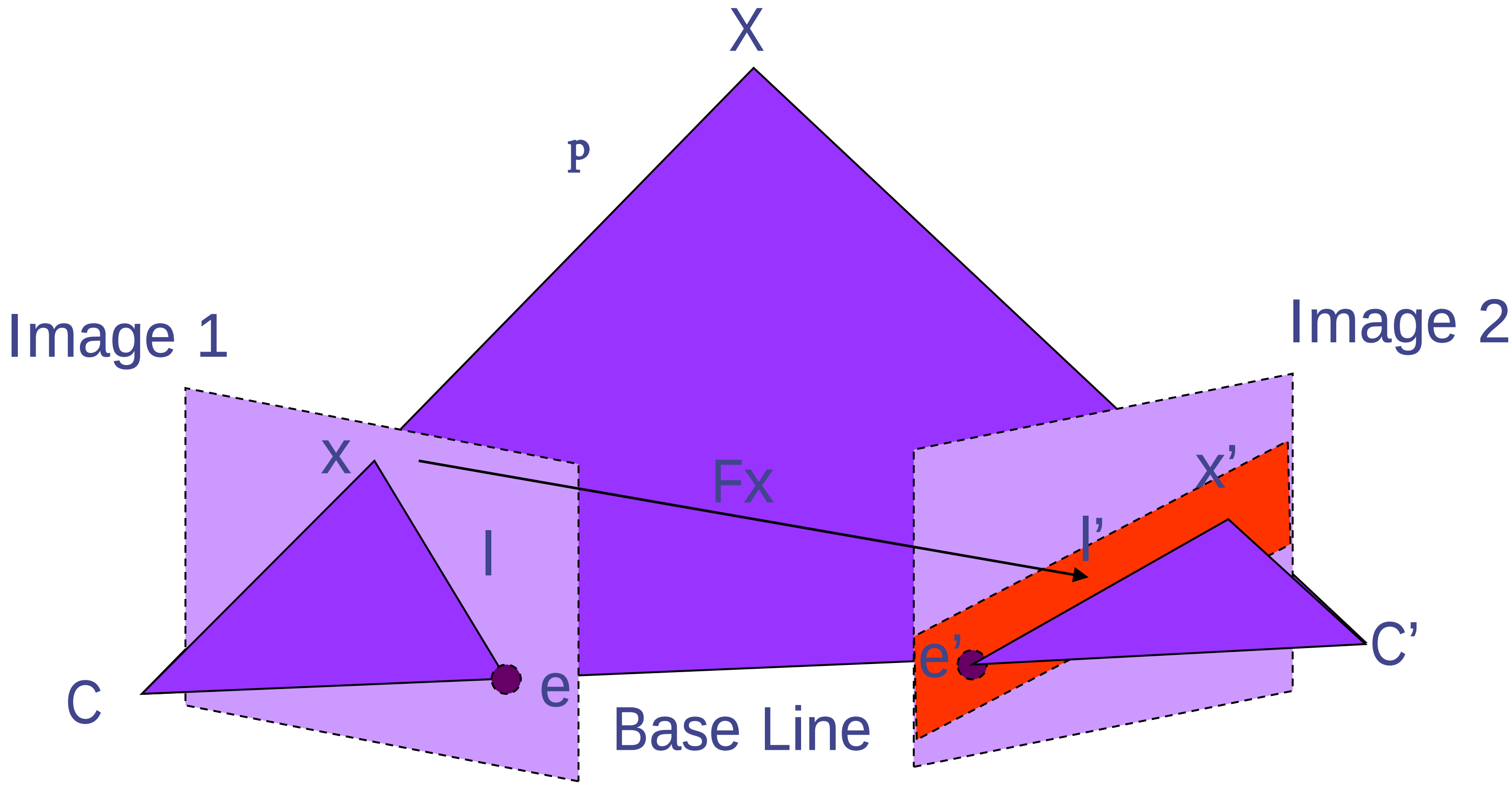


- SIFT

- SURF



FM: Fundamental Matriz



FM: Matriz Fundamental

Fundamental Matrix

- Matrix 3x3 satisfying: $\mathbf{x}'^T F \mathbf{x} = 0$
  - Independent of scene structure

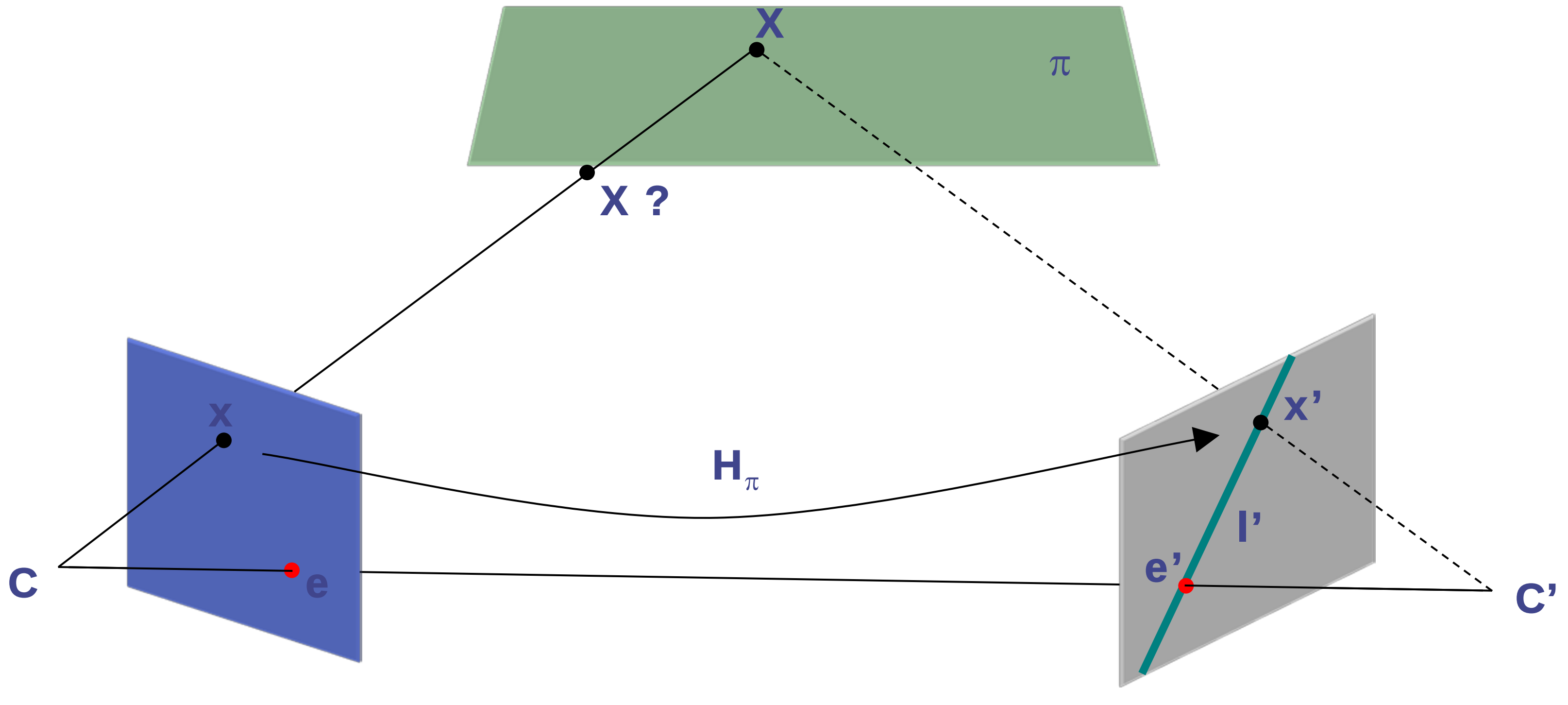
$$\mathbf{F} = \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix}$$

- As a dot product:
- $(x'\cdot x ,\ x'\cdot y ,\ x' ,\ y'\cdot x ,\ y'\cdot y ,\ y' ,\ x ,\ y ,\ 1) \cdot \mathbf{f} = 0$
- With 8 points we have: $A\cdot f=0$
  - 8 points=> Solution to scale factor
  - SVD(A) => Singular vector of smallest singular value

$$\mathbf{F} = \mathbf{K}_2^{-T} \left([\mathbf{t}]_\times \mathbf{R} \right) \mathbf{K}_1^{-1}$$

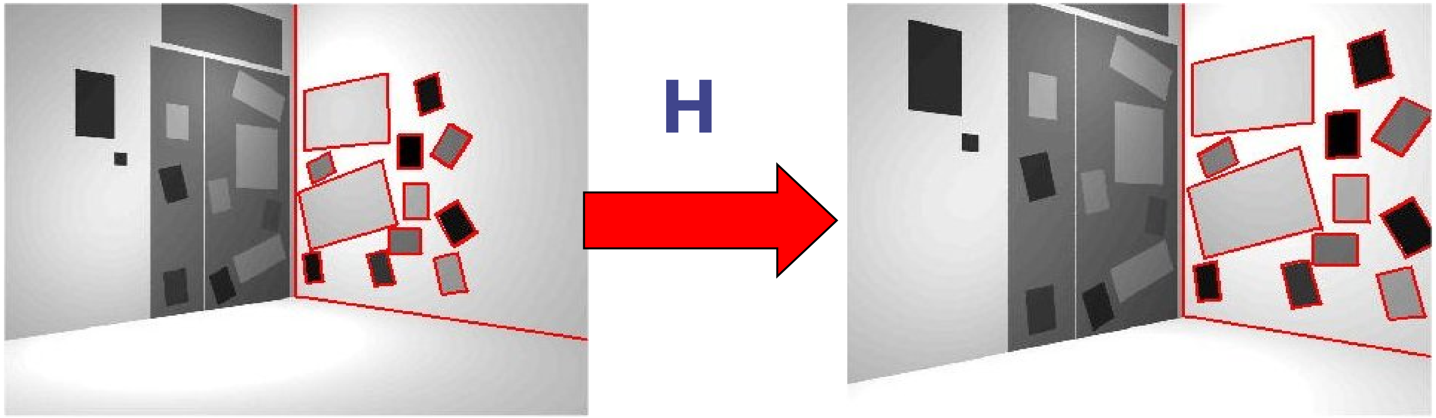
H: Homography

- Projective trasformation between two planes



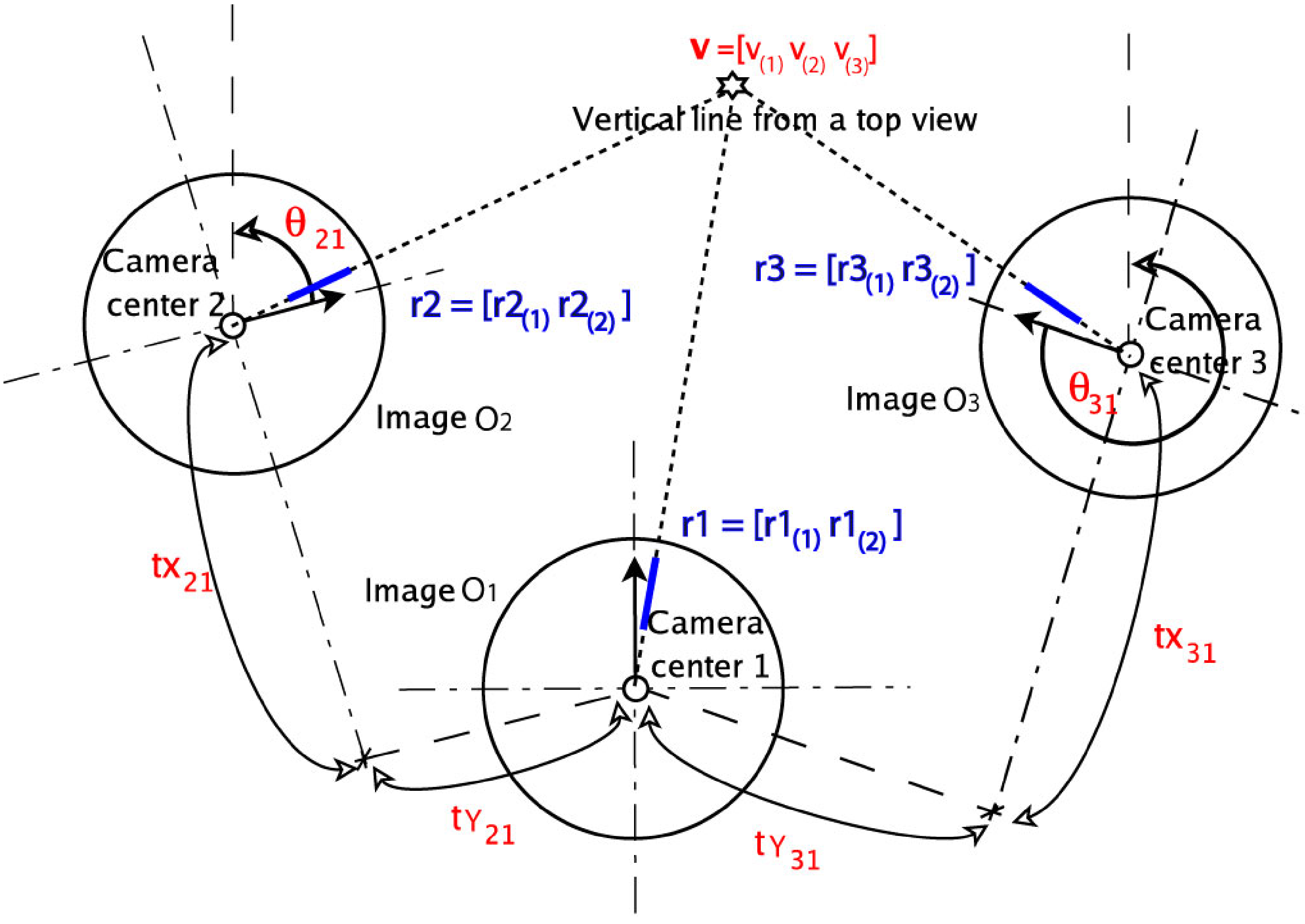
H: Homography

$$\begin{bmatrix} x'_1 \\ x'_2 \\ x'_3 \end{bmatrix} = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \Leftrightarrow \mathbf{x}' = \mathbf{H}\mathbf{x}$$



$$\mathbf{H} = \mathbf{K}\,(\mathbf{R} - \mathbf{t}\,\frac{\mathbf{n}^T}{d})\,\mathbf{K}^{-1}$$

TT: Trifocal tensor (1D)



TT: Trifocal tensor

$$\lambda_1 \mathbf{r}_1 = \mathbf{P}_1 \mathbf{v}$$
$$\lambda_2 \mathbf{r}_2 = \mathbf{P}_2 \mathbf{v}$$
$$\lambda_3 \mathbf{r}_3 = \mathbf{P}_3 \mathbf{v}$$

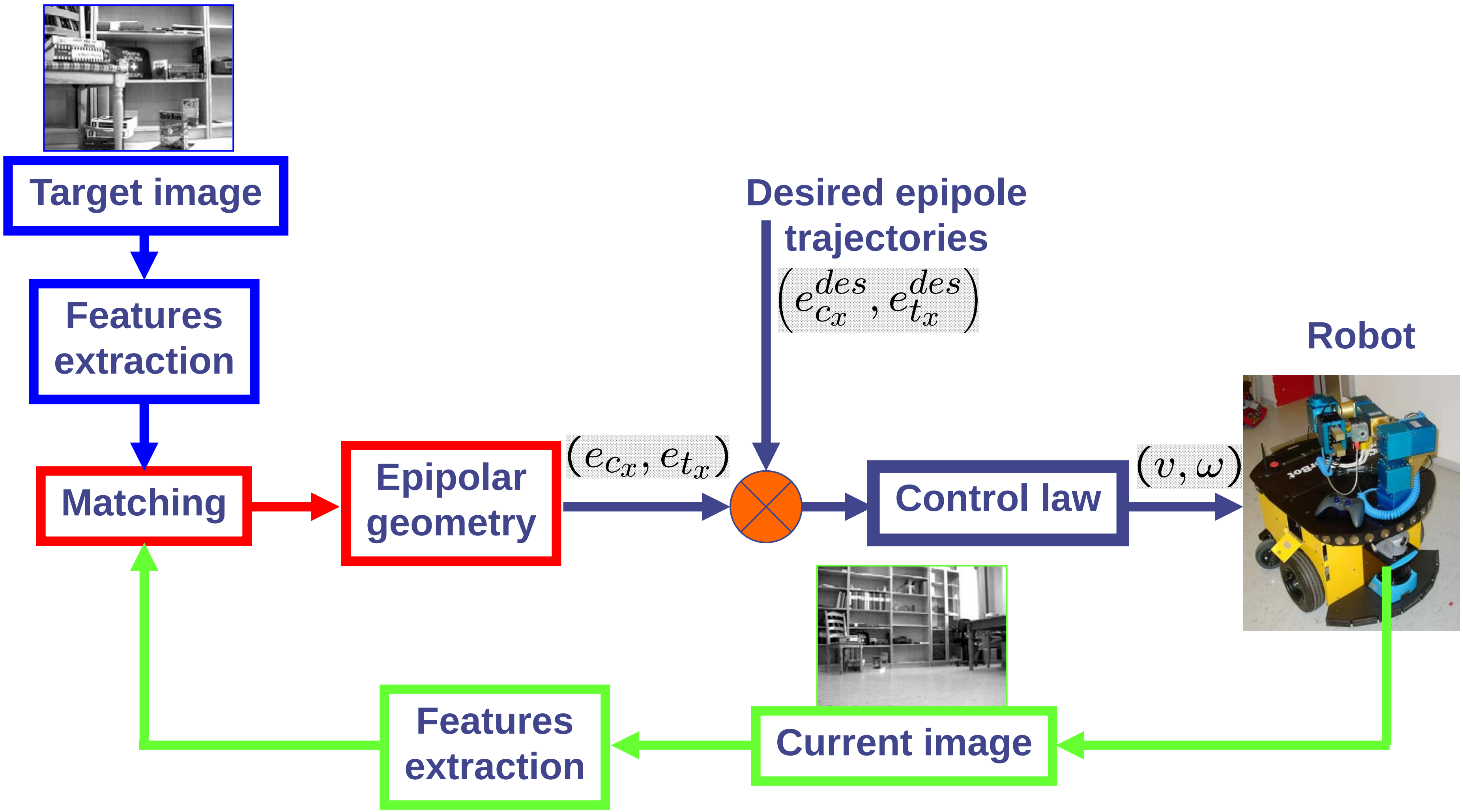
$$\begin{bmatrix} \mathbf{P}_1 & \mathbf{r}_1 & 0 & 0 \\ \mathbf{P}_2 & 0 & \mathbf{r}_2 & 0 \\ \mathbf{P}_3 & 0 & 0 & \mathbf{r}_3 \end{bmatrix} [\mathbf{v}, -\lambda_1, -\lambda_2, -\lambda_3]^T = 0 \qquad \begin{vmatrix} \mathbf{P}_1 & \mathbf{r}_1 & 0 & 0 \\ \mathbf{P}_2 & 0 & \mathbf{r}_2 & 0 \\ \mathbf{P}_3 & 0 & 0 & \mathbf{r}_3 \end{vmatrix} = 0$$

$$\sum_{i=1}^{2} \sum_{j=1}^{2} \sum_{k=1}^{2} T_{ijk} \, \mathbf{r}_{1(i)} \, \mathbf{r}_{2(j)} \mathbf{r}_{3(k)} = 0$$

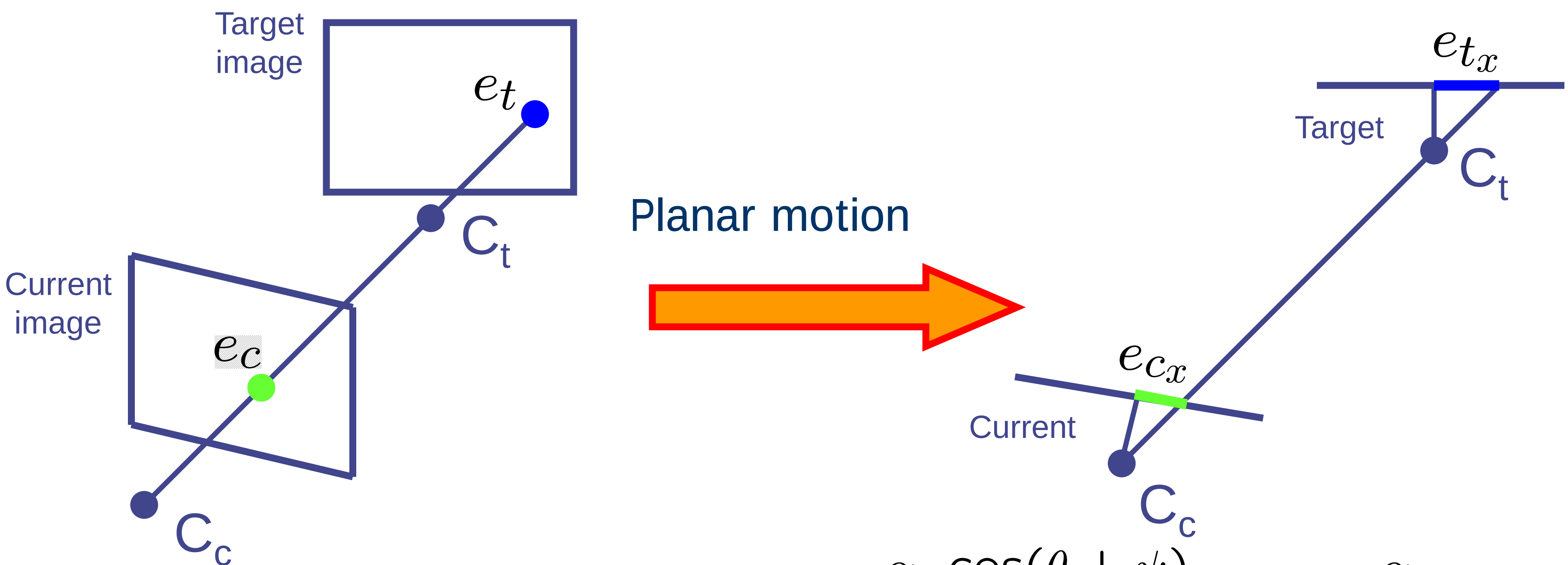
$$\begin{aligned} T_{111} &= t'_z \sin\theta'' - t''_z \sin\theta'; & T_{211} &= -t'_z \cos\theta'' + t''_z \cos\theta' \\ T_{112} &= t'_z \cos\theta'' + t''_x \sin\theta'; & T_{212} &= t'_z \sin\theta'' - t''_x \cos\theta' \\ T_{121} &= -t'_x \sin\theta'' - t''_z \cos\theta'; & T_{221} &= t'_x \cos\theta'' - t''_z \sin\theta' \\ T_{122} &= -t'_x \cos\theta'' + t''_x \cos\theta'; & T_{222} &= -t'_x \sin\theta'' + t''_x \sin\theta'. \end{aligned}$$

The tensor 1D has 2x2x2 elements, wl < 3w−3+2l −1, 5 features needed
The 2D tensor has 3x3x3 elements

Nonholonomic Epipolar Visual Servoing – FM based



Nonholonomic Epipolar Visual Servoing – FM based



$$e_{t_x} = \alpha_x \frac{x}{z}$$

$$e_{c_x} = \alpha_x \frac{x \cos\theta - z \sin\theta}{z \cos\theta + x \sin\theta}$$

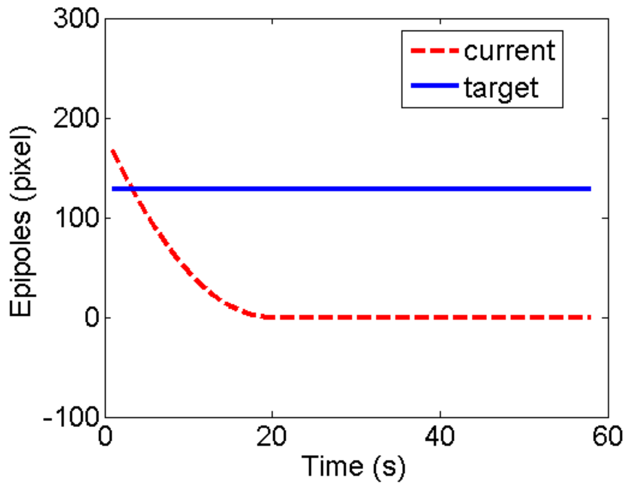
$$\dot{e}_{cx} = -\frac{\alpha_x \cos(\theta + \psi)}{d \sin^2(\theta + \psi)} v - \frac{\alpha_x}{\sin^2(\theta + \psi)} \omega$$

$$\dot{e}_{tx} = -\frac{\alpha_x \cos(\theta + \psi)}{d \sin^2(\psi)} v$$

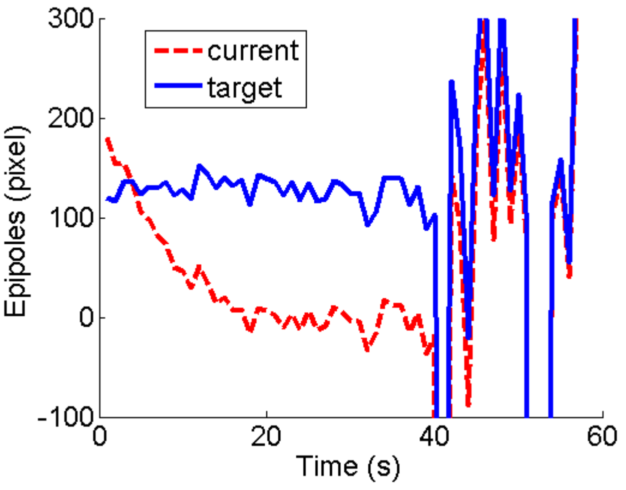
$$\begin{pmatrix} v \\ \omega \end{pmatrix} = L^{-1} \begin{pmatrix} \nu_c \\ \nu_t \end{pmatrix} \quad \text{with} \quad L = \begin{bmatrix} -\frac{\alpha_x \cos(\theta+\psi)}{d \sin^2(\theta+\psi)} & -\frac{\alpha_x}{\sin^2(\theta+\psi)} \\ -\frac{\alpha_x \cos(\theta+\psi)}{d \sin^2(\psi)} & 0 \end{bmatrix}$$

Nonholonomic Epipolar Visual Servoing – FM based

Desired epipole trajectories



Epipoles evolution



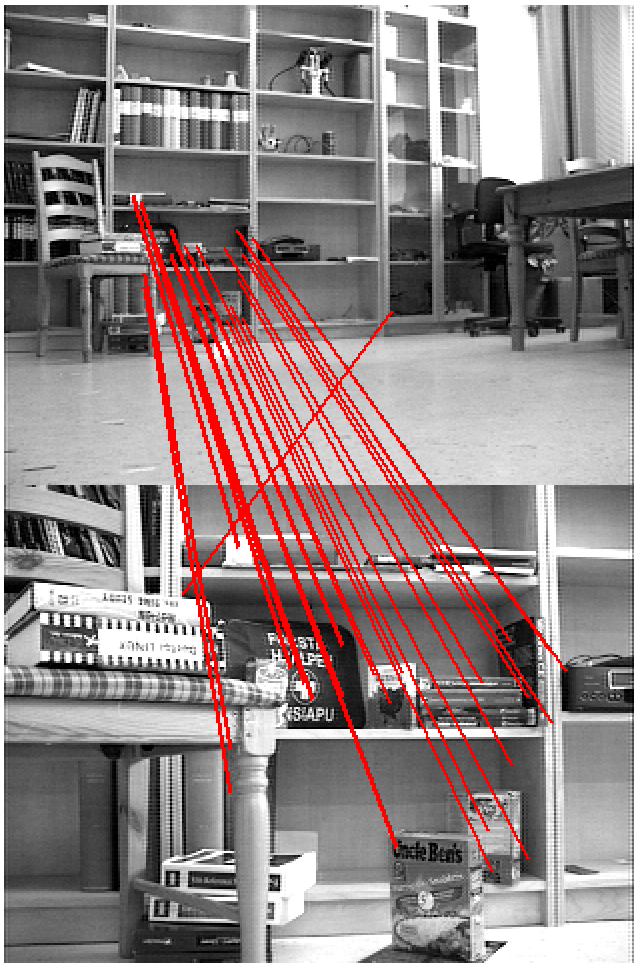
Invertible if det#0

$$\det(L) = -\alpha^2{}_x \cos(\theta+\psi)/d\,\sin^2(\psi)\sin^2(\theta+\psi)$$

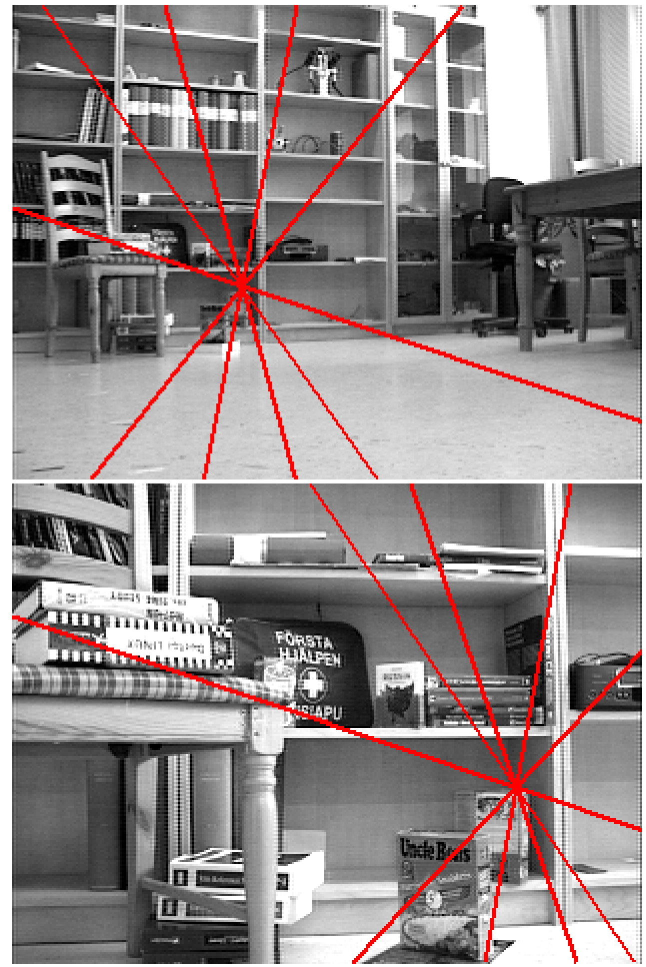
Singularidad $e_{cx} = 0$

$$(\theta+\psi) = 90^{\circ}$$

Nonholonomic Epipolar Visual Servoing – FM based



Matches



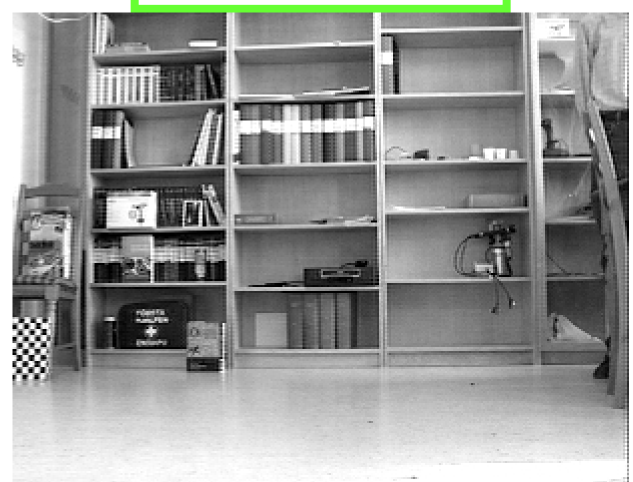
Epipoles and epipolar lines

Nonholonomic Epipolar Visual Servoing – FM based

Target position

Target image

Current image

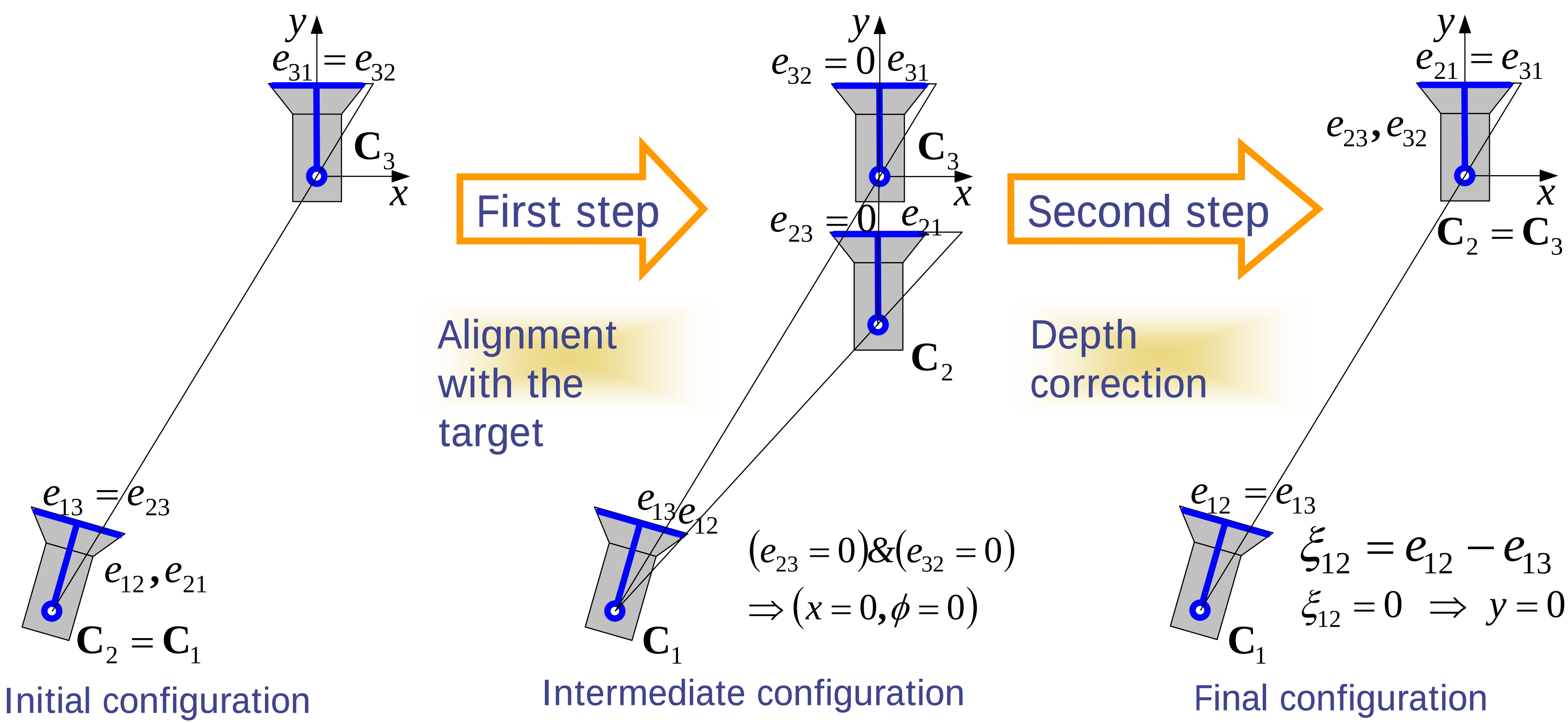


Nonholonomic Epipolar Visual Servoing – FM based

Nonholonomic Epipolar Visual Servoing – FM based

Sliding mode control to avoid singularity

- The control task is carried out in two steps:



Nonholonomic Epipolar Visual Servoing – FM based

- **Control goal of the step** – Solve the stabilization problem in the following error system, where $\xi_{23} = e_{23} - e_{23}^d(t)$, $\xi_{32} = e_{32} - e_{32}^d(t)$.

$$\begin{bmatrix} \dot{\xi}_{23} \\ \dot{\xi}_{32} \end{bmatrix} = \begin{bmatrix} -\frac{\alpha_x \sin(\phi_2 - \psi_2)}{d_{23}\cos^2(\phi_2 - \psi_2)} & \frac{\alpha_x}{\cos^2(\phi_2 - \psi_2)} \\ -\frac{\alpha_x \sin(\phi_2 - \psi_2)}{d_{23}\cos^2(\psi_2)} & 0 \end{bmatrix} \begin{bmatrix} \upsilon \\ \omega \end{bmatrix} - \begin{bmatrix} \dot{e}_{23}^d \\ \dot{e}_{32}^d \end{bmatrix} = L(\phi_2, \psi_2)\mathrm{u} - \dot{\mathrm{e}}^d$$

Desired trajectories

$$e_{23}^d(t) = \sigma \frac{e_{23}(0)}{2}\left(1 + \cos\left(\frac{\pi}{\tau}t\right)\right)$$

$$e_{32}^d(t) = \frac{e_{32}(0)}{2}\left(1 + \cos\left(\frac{\pi}{\tau}t\right)\right)$$

where $\mathbf{L}(\phi,\psi)$ is the so-called decoupling matrix.

- Sliding mode control with sliding surfaces

$$\mathbf{s} = \begin{bmatrix} s_c \\ s_t \end{bmatrix} = \begin{bmatrix} \xi_{23} \\ \xi_{32} \end{bmatrix} = \begin{bmatrix} e_{23} - e_{23}^d \\ e_{32} - e_{32}^d \end{bmatrix} = \mathbf{0}.$$

- Decoupling-based controller.

$$\mathbf{u}_{db} = \begin{bmatrix} \upsilon_{db} \\ \omega_{db} \end{bmatrix} = \frac{1}{\alpha_{xe}} \begin{bmatrix} 0 & -\frac{d_{23e}\cos^2(\psi_2)}{\sin(\phi_2 - \psi_2)} \\ \cos^2(\phi_2 - \psi_2) & -\cos^2(\psi_2) \end{bmatrix} \begin{bmatrix} u_c \\ u_t \end{bmatrix}$$

Singularity if $\phi_2 - \psi_2 = n\pi$

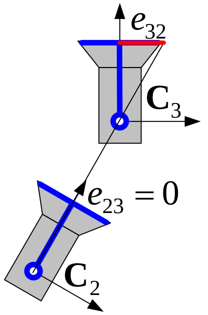
where $u_c = \dot{e}_{23}^d - \lambda_c s_c - \kappa_c \mathrm{sign}(s_c)$,

$$u_t = \dot{e}_{32}^d - \lambda_t s_t - \kappa_t \mathrm{sign}(s_t)$$

Nonholonomic Epipolar Visual Servoing – FM based

- A singular pose is shown in the figure

$$\phi - \psi = \mathbf{arctan}(e_{23} / \alpha_x) = 0$$

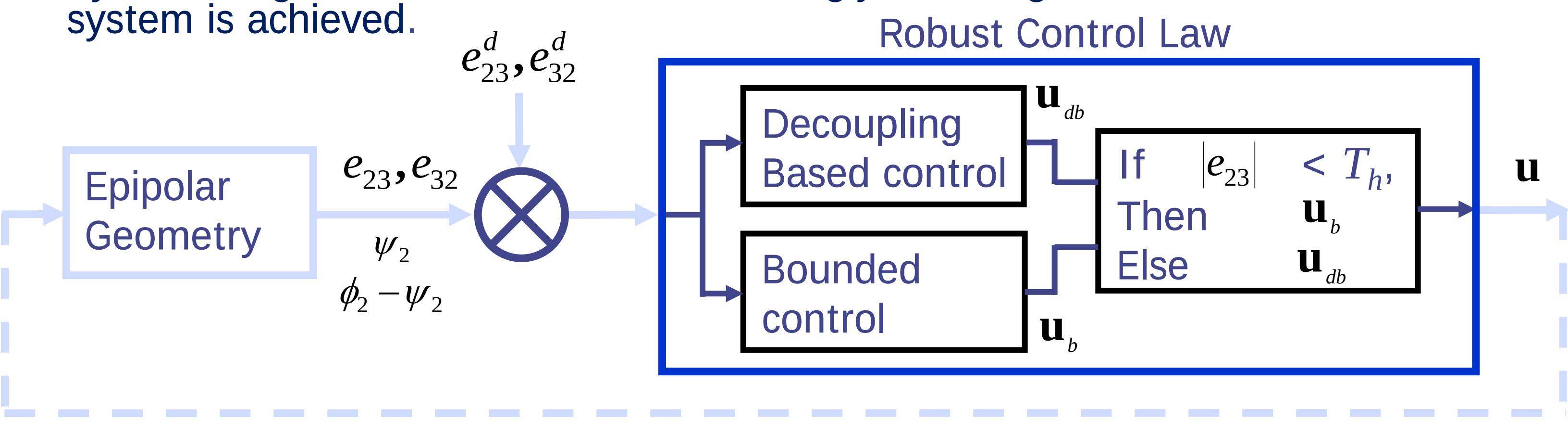


- **Bounded controller.** These inputs don't use the decoupling matrix

$$\mathbf{u}_b = \begin{bmatrix} \upsilon_b \\ \omega_b \end{bmatrix} = \begin{bmatrix} k_\upsilon \ \mathrm{sign}(s_t \ \mathbf{sin}(\phi_2 - \psi_2)) \\ -k_\omega \ \mathrm{sign}(s_c) \end{bmatrix}.$$

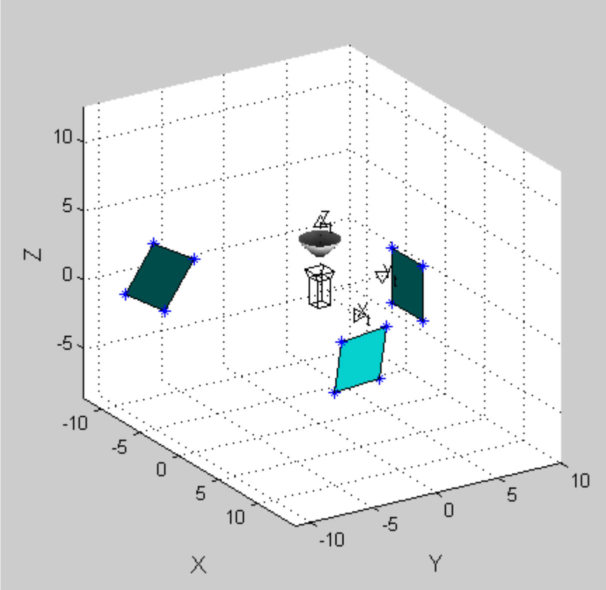
- This is a local control law for the error system.

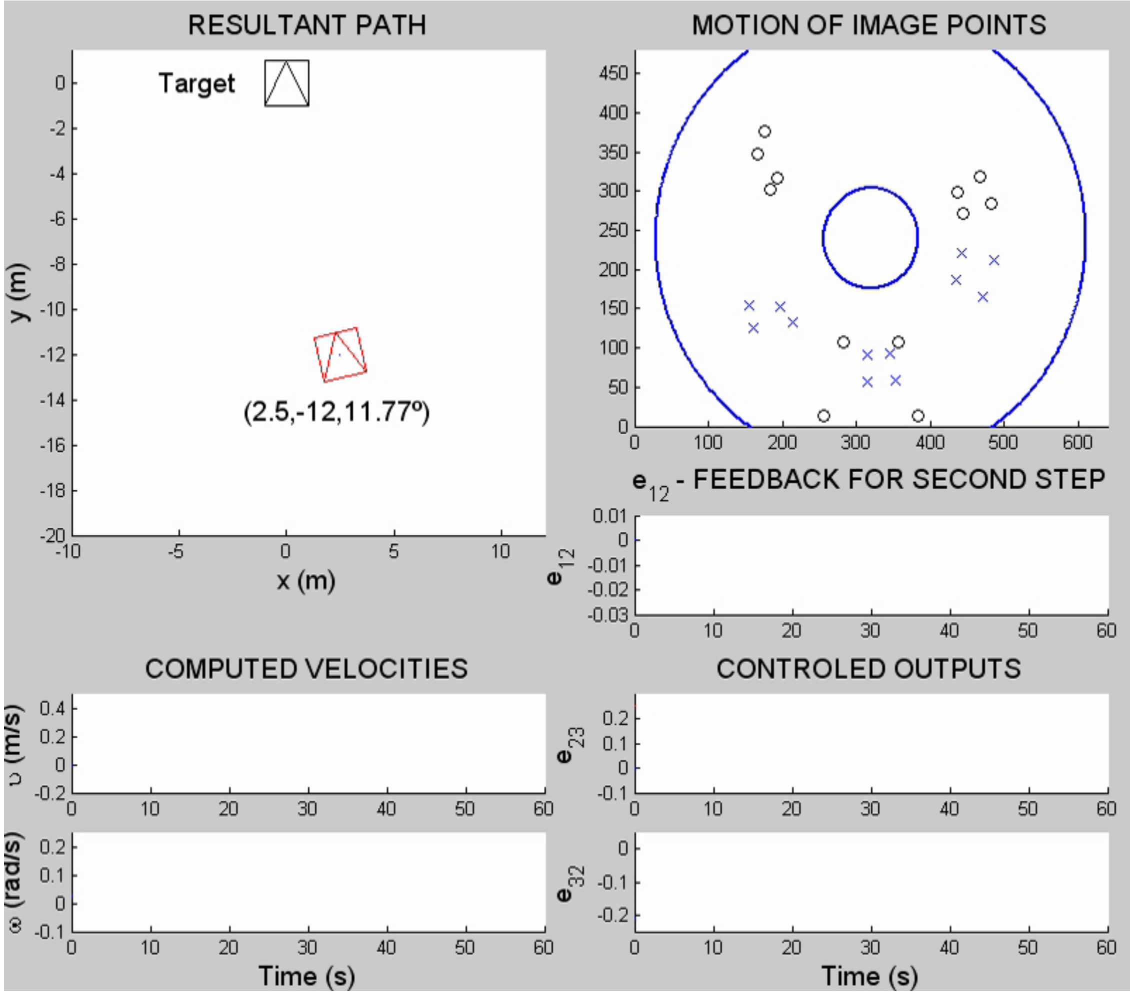
- By switching between controllers accordingly, robust global stabilization of the error system is achieved.



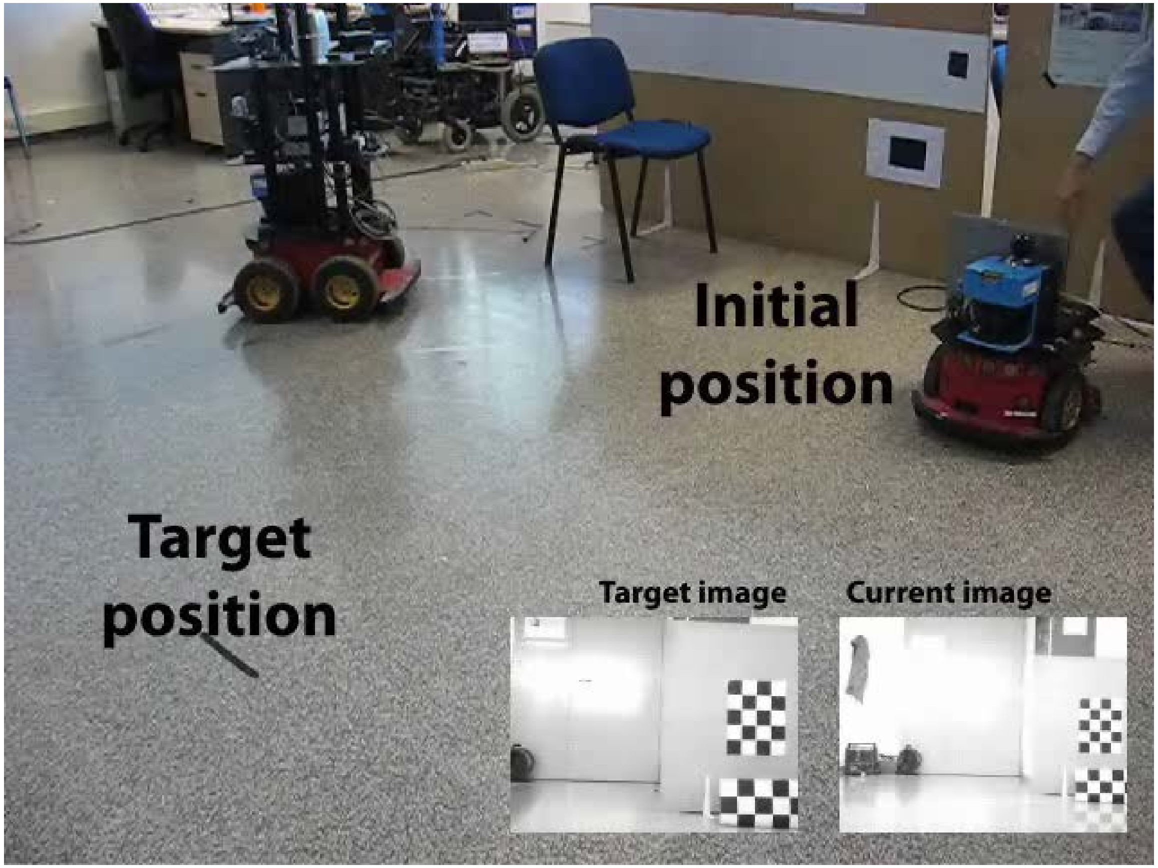
Nonholonomic Epipolar Visual Servoing – FM based

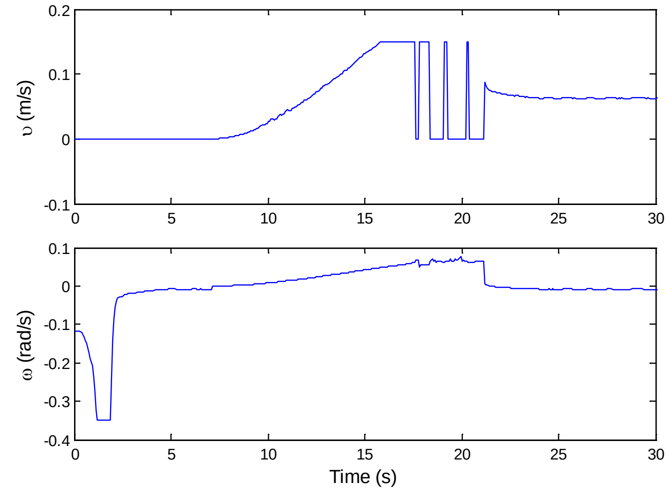
- The epipoles are computed from synthetic images of size 640x480 pixels.
- Target location is (0,0,0º).
- Virtual scene:

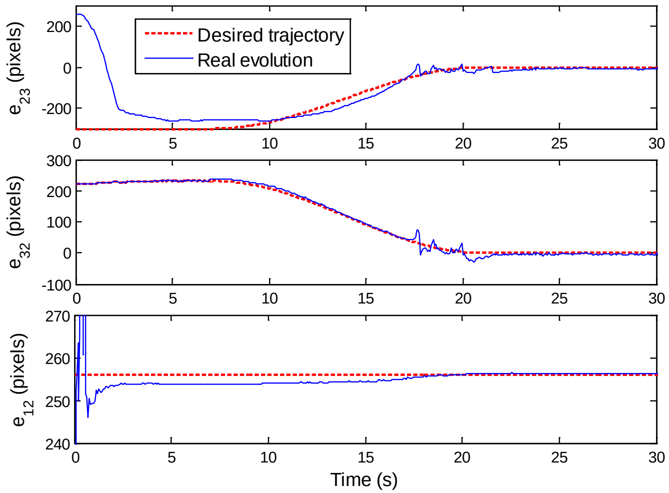




Nonholonomic Epipolar Visual Servoing – FM based





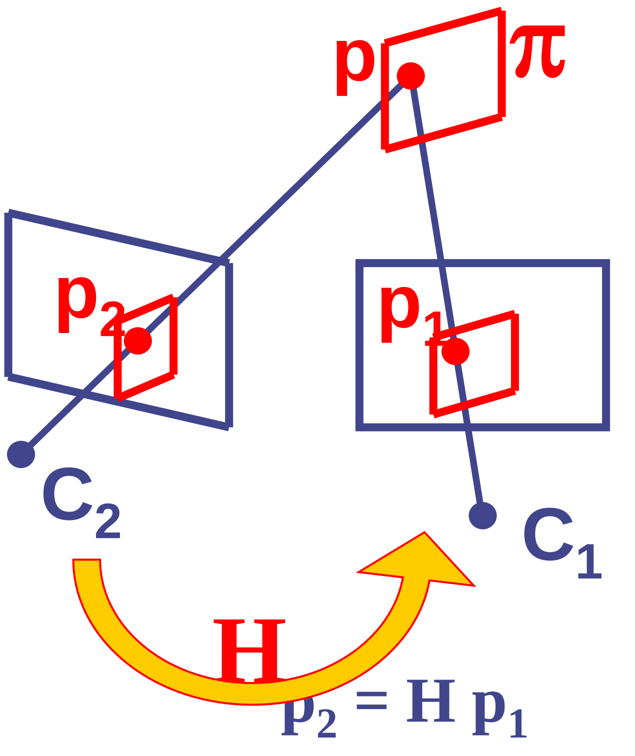


Nonholonomic Homography based – H based

- Two images can be geometrically linked by a homography
- The homography is generated by a plane of the scene
- The homography can be computed from point matches

$$\mathbf{H} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$$

- Goal: $\mathrm{H} = \mathrm{I}$



Nonholonomic Homography based – H based

- The homography is related to camera motion:

$$\mathbf{H} = \mathbf{K}\,(\mathbf{R} - \mathbf{t}\,\frac{\mathbf{n}^T}{d})\,\mathbf{K}^{-1}$$

- Planar motion:

$$\mathbf{H} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ 0 & 1 & 0 \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \quad \text{with:} \quad \begin{cases} h_{11} = \cos\phi + (x\cos\phi + z\sin\phi)\frac{n_x}{d} \\ h_{12} = \frac{\alpha_x}{\alpha_y}(x\cos\phi + z\sin\phi)\frac{n_y}{d} \\ h_{13} = \alpha_x\left(\sin\phi + (x\cos\phi + z\sin\phi)\frac{n_z}{d}\right) \\ h_{31} = \frac{1}{\alpha_x}\left(-\sin\phi + (-x\sin\phi + z\cos\phi)\frac{n_x}{d}\right) \\ h_{32} = \frac{1}{\alpha_y}(-x\sin\phi + z\cos\phi)\frac{n_y}{d} \\ h_{33} = \cos\phi + (-x\sin\phi + z\cos\phi)\frac{n_z}{d} \end{cases}$$

- Non-linear relation of H with state system:

$$(x,\ z,\ \phi)^T \Longleftrightarrow h_{ij}$$

Nonholonomic Homography based VS

- 2 dof system → Two elements of the homography are enough to define the control
- Derivatives of the output functions:

$$\begin{cases} \dot{h}_{13} = \alpha_x \, h_{33} \, \omega \\ \dot{h}_{33} = \frac{n_z}{d} \, v - \frac{h_{13}}{\alpha_x} \, \omega \end{cases}$$

- State space form

$$\begin{cases} \dot{\mathbf{x}} & = & f(\mathbf{x}, \mathbf{u}) \\ \mathbf{y} & = & h(\mathbf{x}) \end{cases}$$

with:

- **State vector:** $\mathbf{x} = (x, z, \phi)^T$
- **Input vector:** $\mathbf{u} = (v, \omega)^T$
- **Output vector:** $\mathbf{y} = (h_{13}, h_{33})^T$

- Linear relation between the input and output

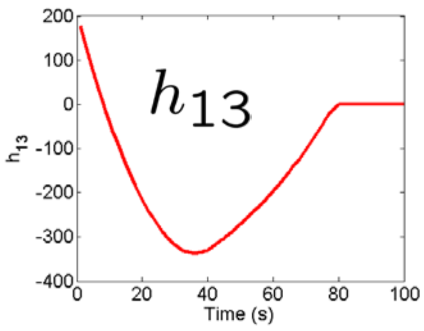
$$\begin{pmatrix} v \\ \omega \end{pmatrix} = \mathbf{L}^{-1} \begin{pmatrix} \nu_{13} \\ \nu_{33} \end{pmatrix}$$

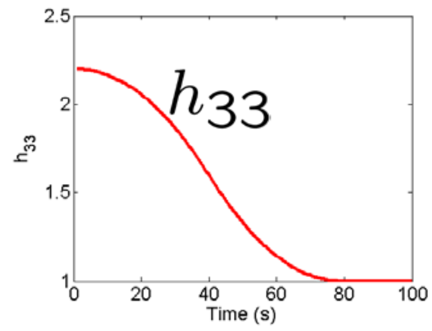
with: $\mathbf{L} = \begin{bmatrix} 0 & \alpha_x h_{33} \\ \frac{n_z}{d} & -\frac{h_{13}}{\alpha_x} \end{bmatrix}$

Nonholonomic Homography based – H based

- Tracking of the desired trajectories of the homography elements

- Input of the control:
  - Exponentially stable error dynamics





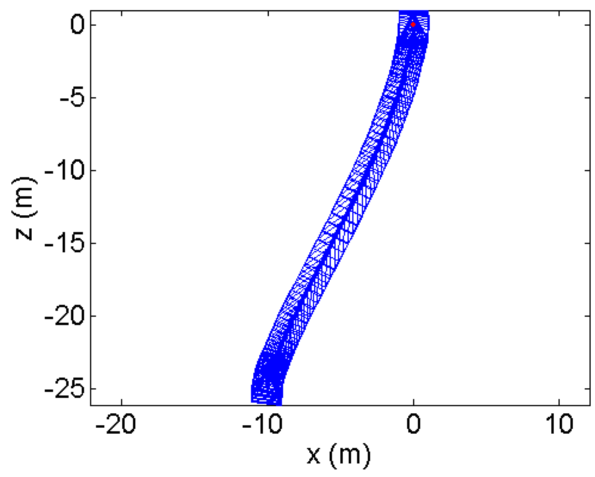
$$\begin{pmatrix} \nu_{13} \\ \nu_{33} \end{pmatrix} = \begin{pmatrix} \dot{h}^d_{13} - k_{13}(h_{13} - h^d_{13}) \\ \dot{h}^d_{33} - k_{33}(h_{33} - h^d_{33}) \end{pmatrix}$$

- Desired trajectories:

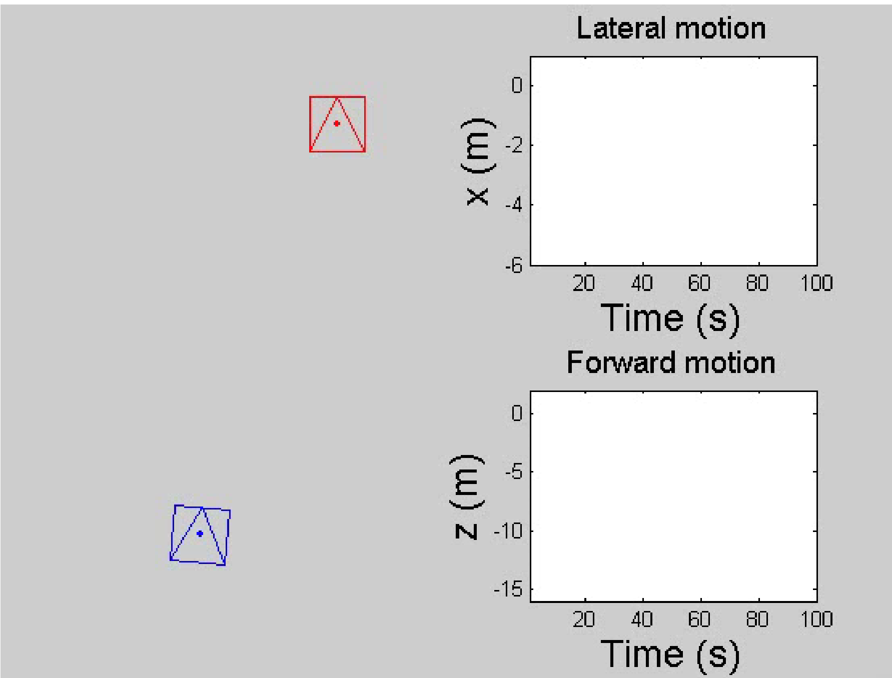
$$0 \le t \le T_1 \begin{cases} h^d_{13}(t) = (h_{13}(0) - g_t)\left(\frac{t^2}{T_1^2} - 2\frac{t}{T_1} + 1\right) + g_t \\ h^d_{33}(t) = \left(\frac{1-h_{33}(0)}{2}\right)\left(\frac{t^2}{T_1^2} + 1\right) + (3\,h_{33}(0) - 1)/2 \end{cases}$$

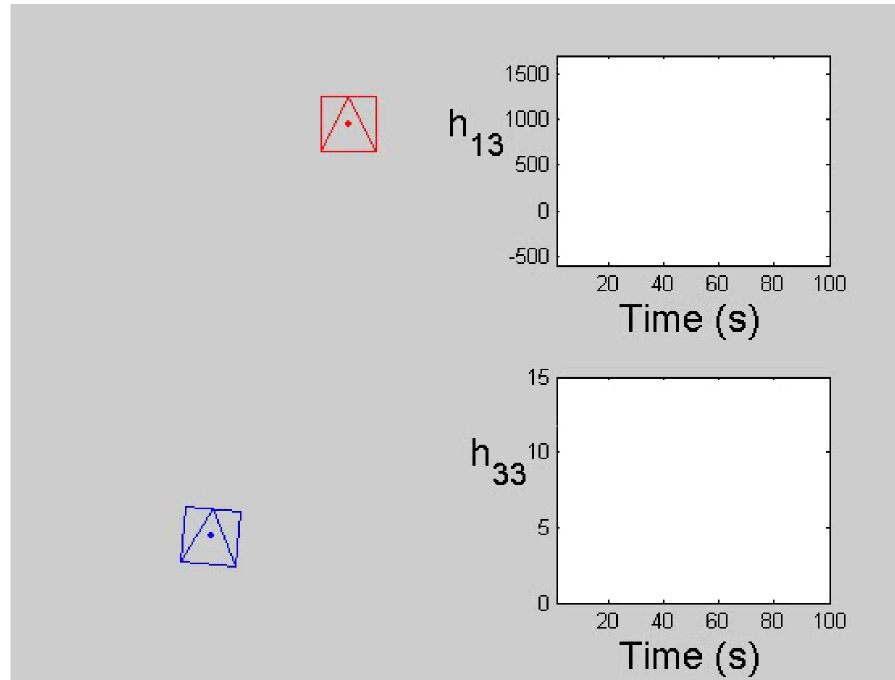
$$T_1 < t \le T_2 \begin{cases} h^d_{13}(t) = h_{13}(T_1)\frac{\phi_t(t)}{\phi_t(T_1)} \\ h^d_{33}(t) = \left(\frac{h_{33}(0)-1}{2}\right)\left(\frac{(t-T_1)^2}{(T_2-T_1)^2} - 2\frac{t-T_1}{T_2-T_1} + 1\right) + 1 \end{cases}$$

$$t > T_2 \begin{cases} h^d_{13}(t) = 0 \\ h^d_{33}(t) = 1 \end{cases}$$

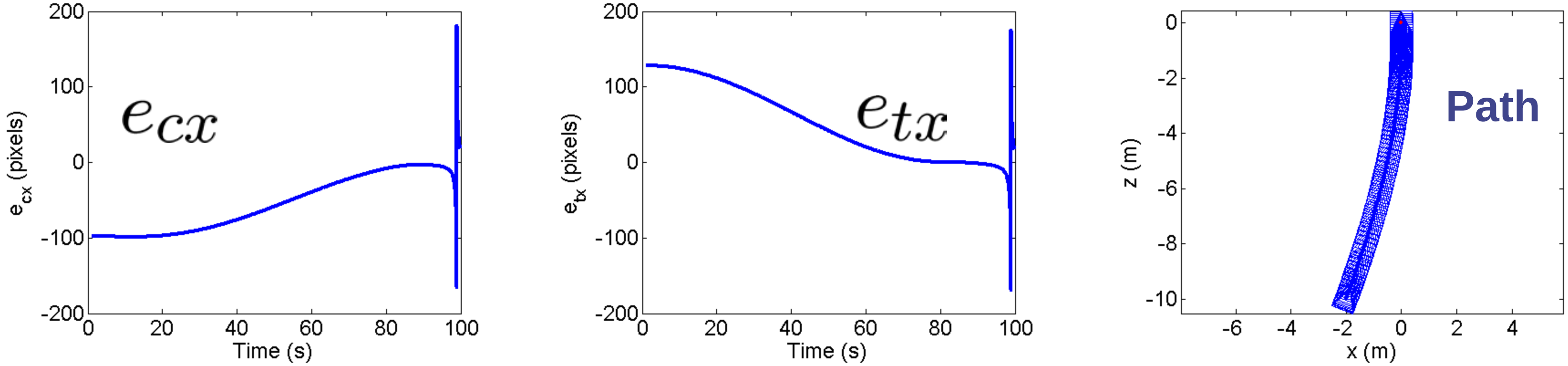


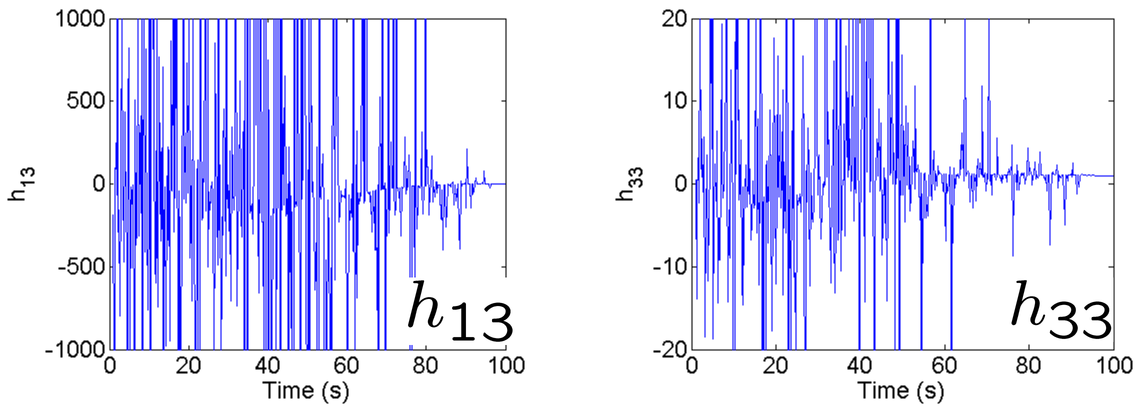
Nonholonomic Homography based – H based





Combination of Epipoles/Homographies for VS



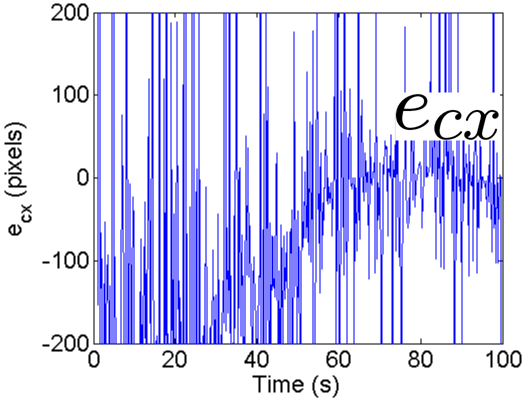


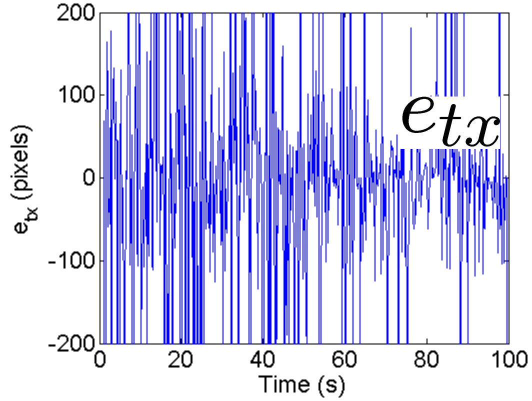
✓ **Epipoles**

✗ **Homography**

Epipolar-based control:
$$\begin{pmatrix} v_F \\ \omega_F \end{pmatrix} = \frac{1}{\alpha_x} \begin{bmatrix} 0 & -\frac{d\cos^2(\psi)}{\sin(\phi-\psi)} \\ \cos^2(\phi-\psi) & -\cos^2(\psi) \end{bmatrix} \begin{pmatrix} \nu_c \\ \nu_t \end{pmatrix}$$

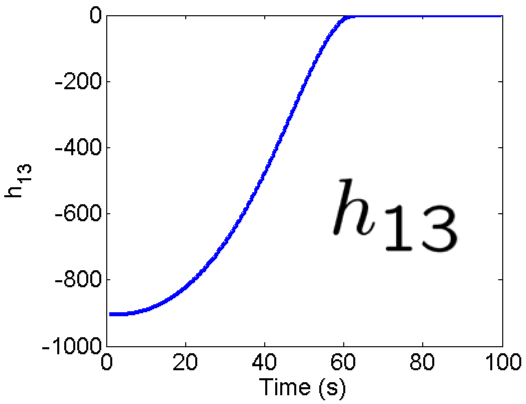
Combination of Epipoles/Homographies for VS

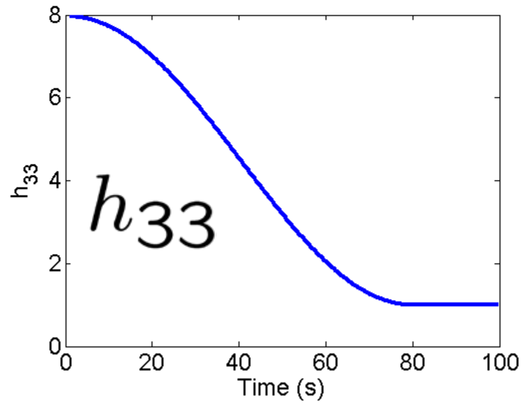


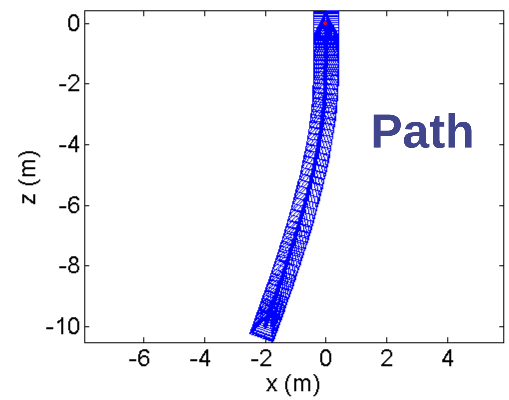


✘ **Epipoles**

✔ **Homography**



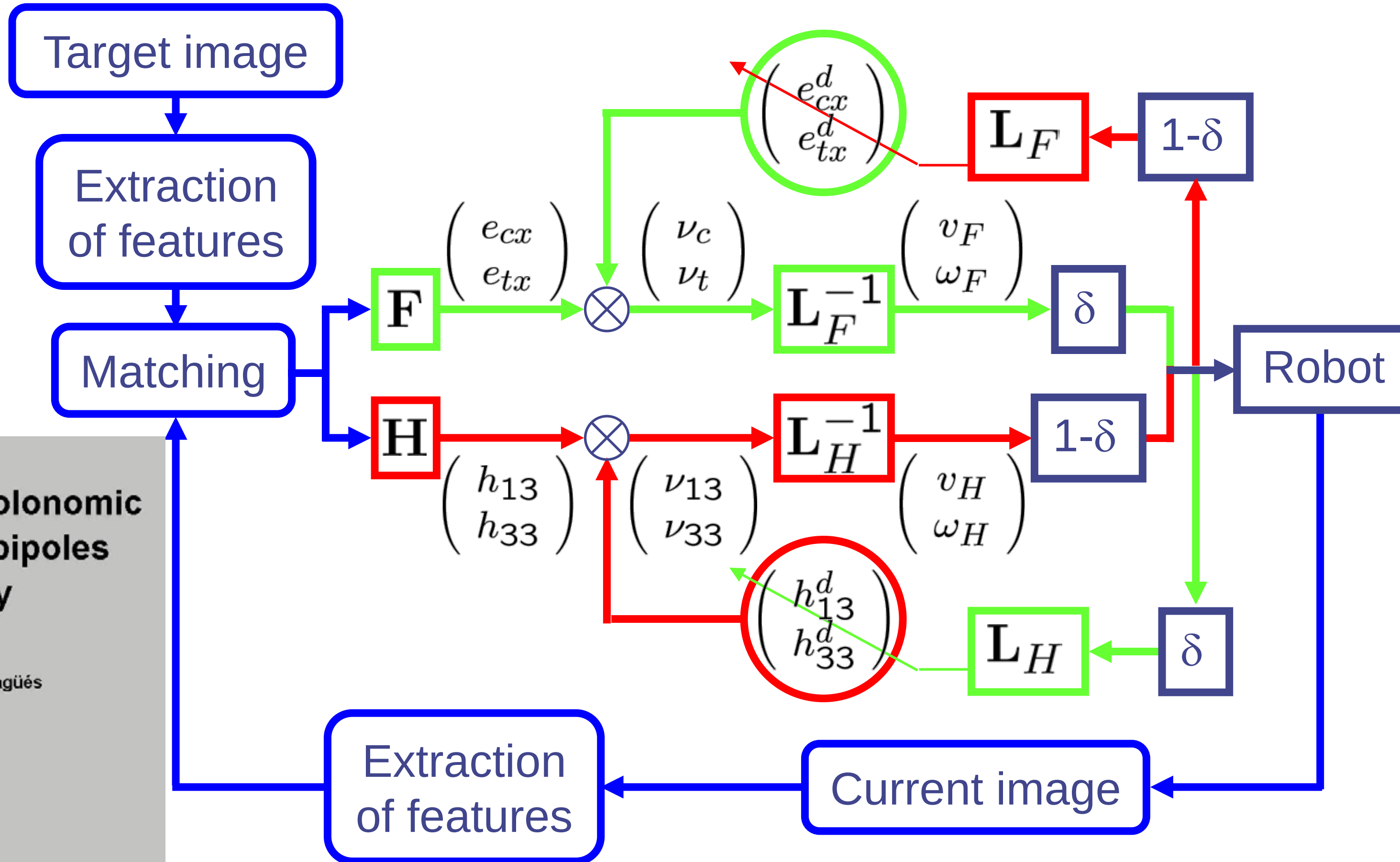




Homography-based control:

$$\begin{pmatrix} v_H \\ \omega_H \end{pmatrix} = \begin{bmatrix} \frac{h_{13}}{\alpha_x^2 h_{33}} \frac{d_\pi}{n_z} & \frac{d_\pi}{n_z} \\ \frac{1}{\alpha_x h_{33}} & 0 \end{bmatrix} \begin{pmatrix} \nu_{13} \\ \nu_{33} \end{pmatrix}$$

Combination of Epipoles/Homographies for VS



Visual Control of Nonholonomic Vehicles Exploiting Epipoles and Homography

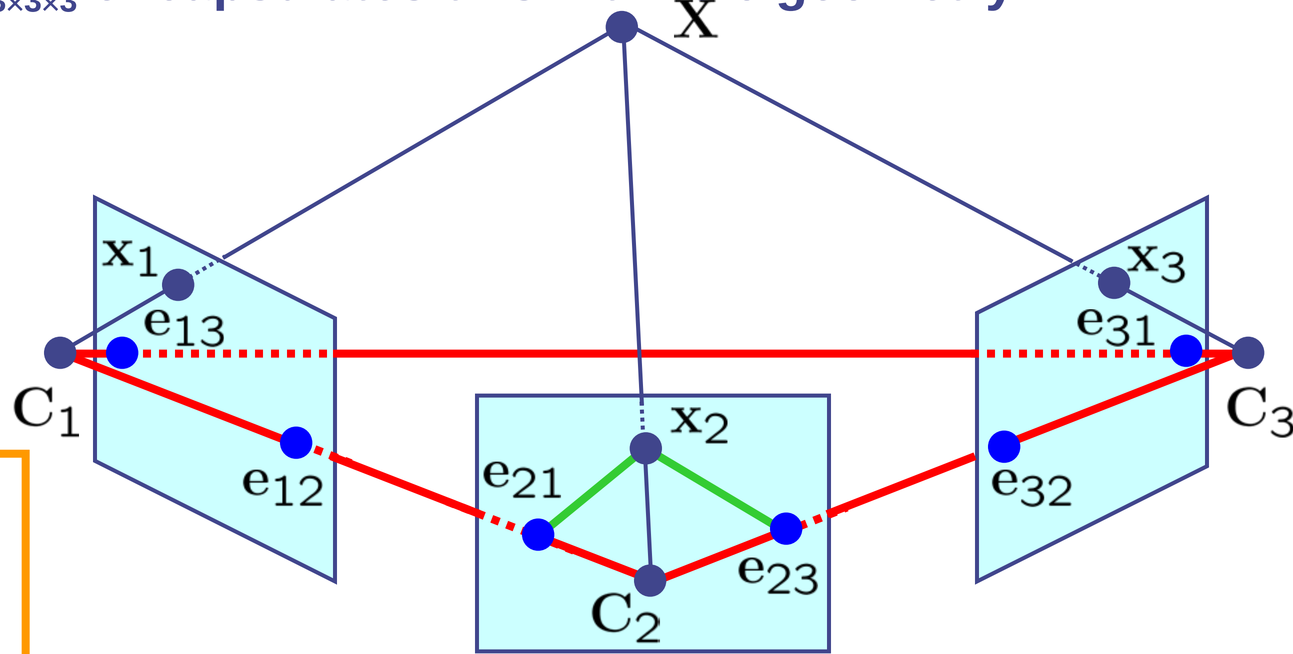
G. López-Nicolás, J.J. Guerrero and C. Sagüés

$$\begin{pmatrix} \upsilon \\ \omega \end{pmatrix} = \begin{bmatrix} 0 & \frac{-\delta d \cos^2(\psi)}{\alpha_x \sin(\phi-\psi)} & \frac{(1-\delta)h_{13}d_\pi}{\alpha_x^2 h_{33} n_z} & \frac{(1-\delta)d_\pi}{n_z} \\ \frac{\delta \cos^2(\phi-\psi)}{\alpha_x} & \frac{-\delta \cos^2(\psi)}{\alpha_x} & \frac{1-\delta}{\alpha_x h_{33}} & 0 \end{bmatrix} \begin{pmatrix} \nu_c \\ \nu_t \\ \nu_{13} \\ \nu_{33} \end{pmatrix}$$

Combination of Epipoles/Homographies for VS

Visual control – TT based

- The trifocal tensor is the intrinsic geometry between three views.
- It only depends on the camera internal parameters and relative pose.
- The trifocal tensor $\mathrm{T}_{3\times3\times3}$ encapsulates this intrinsic geometry.



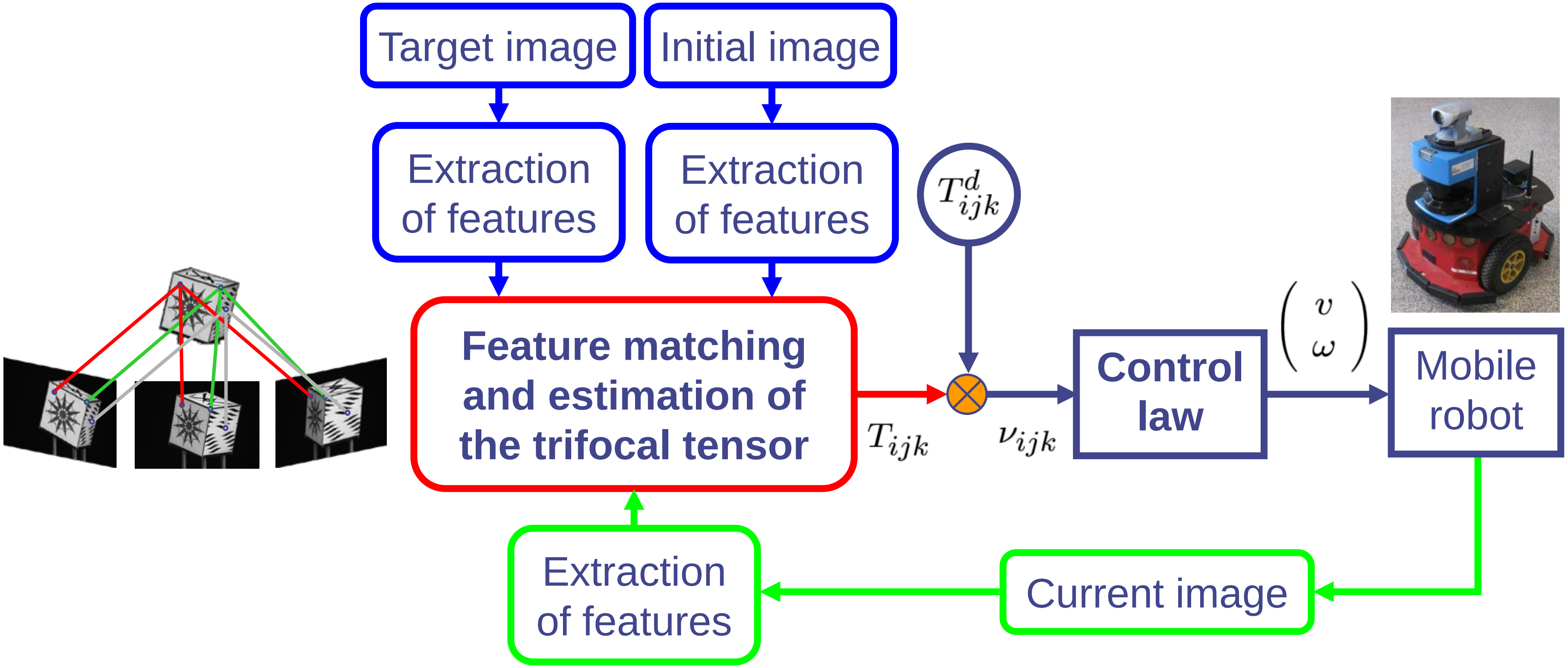
Matrix notation

$$\mathbf{T} = [\mathbf{T}_1, \mathbf{T}_2, \mathbf{T}_3]$$

$$[\mathbf{x}_2]_\times \left(\sum_i \mathbf{x}_1^i \mathbf{T}_i \right) [\mathbf{x}_3]_\times = 0_{3\times3}$$

Seven correspondences needed

Visual control – TT based

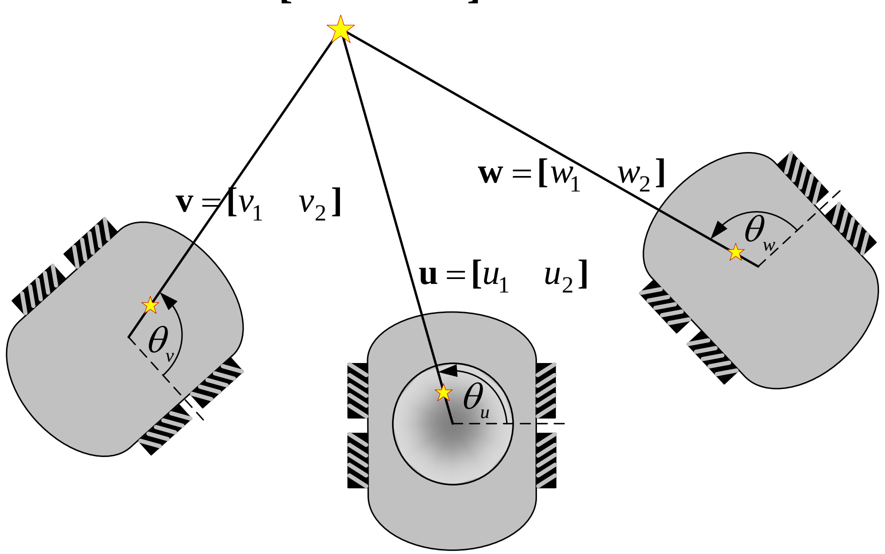


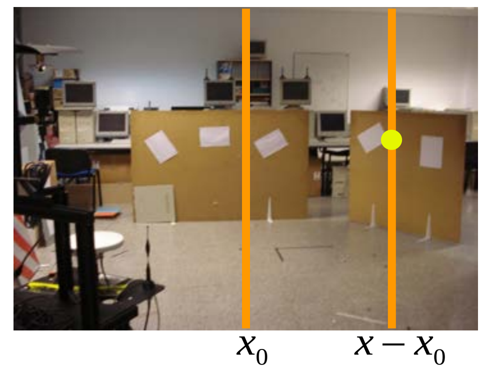
Visual control – TT based

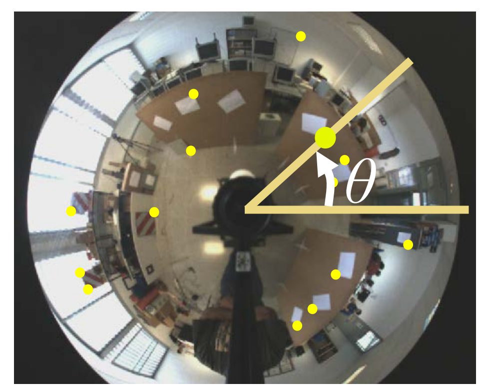
- Particularly the 1D trifocal tensor allows:
  - Exploit the bearing information.
  - Reduce the camera calibration parameters required for control (center of projection and vertical alignment).
- The trifocal tensor is a more general geometric constraint than epipolar geometry.
- Epipolar geometry is ill-conditioned with short baseline and with planar scenes.
- Five corresponding points

$$\sum_{i=1}^{2}\sum_{j=1}^{2}\sum_{k=1}^{2} T_{ijk} u_i v_j w_k = 0$$

$$\mathbf{X} = [X \quad Y \quad Z]$$

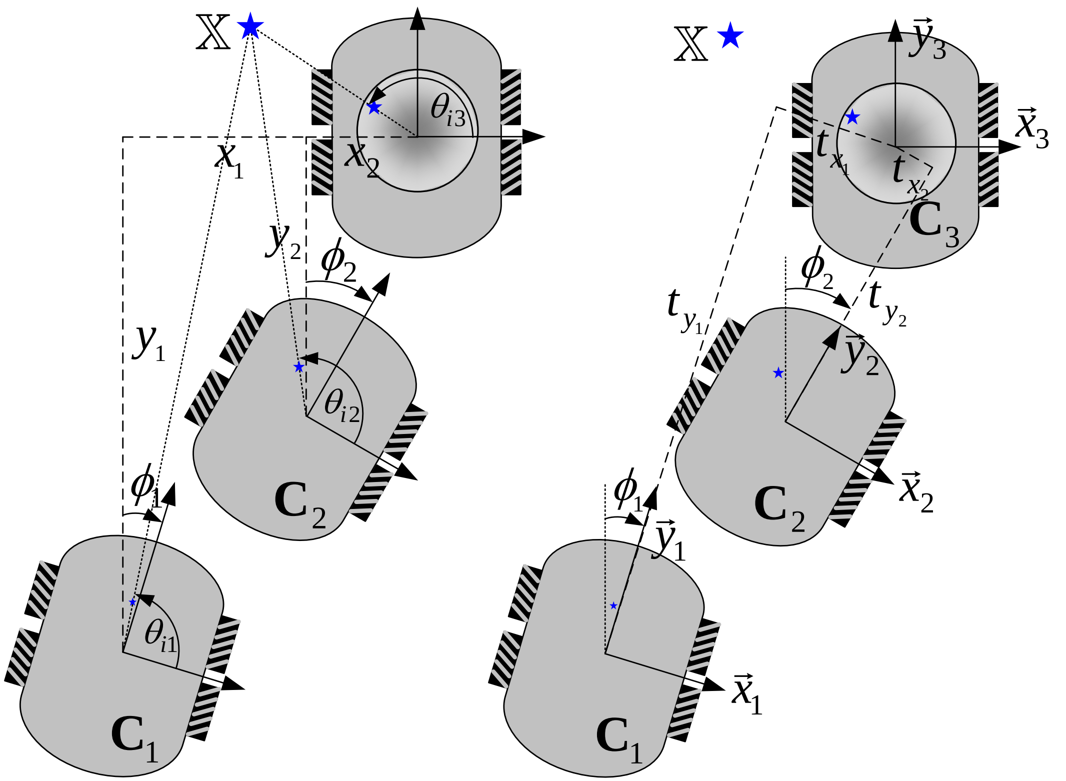






Visual control – TT based

- Initial location $\mathbf{C}_1 = (x_1, y_1, \phi_1)$.
- Target location $\mathbf{C}_3 = (0, 0, 0)$.
- Current location (moving camera) $\mathbf{C}_2 = (x_2, y_2, \phi_2)$.



8 elements of the tensor:

$$\mathbf{T}^m_{ijk} = \begin{bmatrix} T^m_{111} \\ T^m_{112} \\ T^m_{121} \\ T^m_{122} \\ T^m_{211} \\ T^m_{212} \\ T^m_{221} \\ T^m_{222} \end{bmatrix} = \begin{bmatrix} t_{y_1} \sin\phi_2 - t_{y_2} \sin\phi_1 \\ -t_{y_1} \cos\phi_2 + t_{y_2} \cos\phi_1 \\ t_{y_1} \cos\phi_2 + t_{x_2} \sin\phi_1 \\ t_{y_1} \sin\phi_2 - t_{x_2} \cos\phi_1 \\ -t_{x_1} \sin\phi_2 - t_{y_2} \cos\phi_1 \\ t_{x_1} \cos\phi_2 - t_{y_2} \sin\phi_1 \\ -t_{x_1} \cos\phi_2 + t_{x_2} \cos\phi_1 \\ -t_{x_1} \sin\phi_2 + t_{x_2} \sin\phi_1 \end{bmatrix}$$

where the relative locations between cameras are given as

$$\begin{bmatrix} t_{x_i} \\ t_{y_i} \end{bmatrix} = -\begin{bmatrix} \cos\phi_i & \sin\phi_i \\ -\sin\phi_i & \cos\phi_i \end{bmatrix} \cdot \begin{bmatrix} x_i \\ y_i \end{bmatrix}$$

for $i = 1,2$.

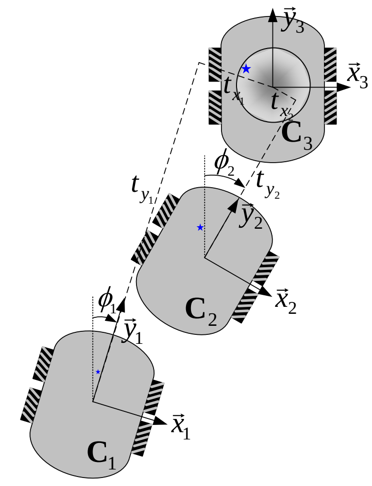
This is an over-constrained measurement

Visual control – TT based

Values of the trifocal tensor in particular locations

- When $C_2 = C_1$ $(t_{x_2} = t_{x_1}, t_{y_2} = t_{y_1})$ ➡ $T_{111} = 0,\ T_{112} = 0,\ T_{121} + T_{211} = 0,$ $T_{221} = 0,\ T_{222} = 0,\ T_{122} + T_{212} = 0.$

- When $C_2 = C_3$ $(t_{x_2} = 0, t_{y_2} = 0)$ ➡ $T_{111} = 0,\ T_{122} = 0,\ T_{112} + T_{121} = 0,$ $T_{211} = 0,\ T_{222} = 0,\ T_{212} + T_{221} = 0.$



Time-derivatives of the elements of the tensor

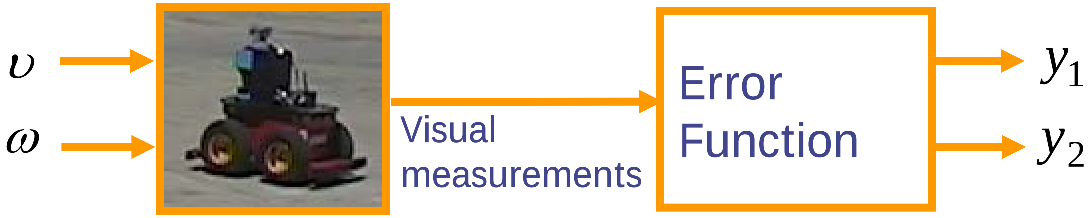
$$\dot{T}_{111} = \frac{\sin\phi_1}{T_N^m}\upsilon + T_{121}\omega, \qquad \dot{T}_{211} = \frac{\cos\phi_1}{T_N^m}\upsilon + T_{221}\omega,$$
$$\dot{T}_{112} = -\frac{\cos\phi_1}{T_N^m}\upsilon + T_{122}\omega, \qquad \dot{T}_{212} = \frac{\sin\phi_1}{T_N^m}\upsilon + T_{222}\omega,$$

$$\dot{T}_{121} = -T_{111}\omega, \qquad \dot{T}_{221} = -T_{211}\omega,$$
$$\dot{T}_{122} = -T_{112}\omega, \qquad \dot{T}_{222} = -T_{212}\omega.$$

➡ Useful only for orientation control

Visual control – TT based

- Three variables to desired values but we choose to make a Square control system.



- By using two outputs, the tensor provides three possibilities:

First part of the control — Second part

	Correcting	DOF	Drawback
1	Orientation and depth (ϕ, y)	Lateral error (x)	Non-holonomic constraint does not allow to correct the remainder lateral error.
2	Orientation and lateral error (ϕ, x)	Depth (y)	Unknown final values of the tensor elements to define the control objective.
3	Lateral error and depth (x, y)	Orientation (ϕ)	Differential-drive allows to correct the remainder orientation error.

Visual control – TT based

Position correction with two selected outputs:

$$\xi_1 = T_{112} + T_{121},$$
$$\xi_2 = T_{212} + T_{221}.$$

- When $\xi_1 \equiv 0, \; \xi_2 \equiv 0$

$$\begin{bmatrix} T_{112} + T_{121} \\ T_{212} + T_{221} \end{bmatrix} = \begin{bmatrix} \sin\phi_1 & \cos\phi_1 \\ \cos\phi_1 & -\sin\phi_1 \end{bmatrix} \begin{bmatrix} t_{x_2} \\ t_{y_2} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

- Zero dynamics:

$$Z^* = \left\{ \begin{bmatrix} x_2 & y_2 & \phi_2 \end{bmatrix}^T \mid \xi_1 \equiv 0, \; \xi_2 \equiv 0 \right\}$$
$$= \left\{ \begin{bmatrix} 0 & 0 & \phi_2 \end{bmatrix}^T, \; \phi_2 \in R \right\}.$$

- **Control goal of the step** – Stabilize the following error system, where $e_1 = \xi_1 - \xi_1^d$ and $e_2 = \xi_2 - \xi_2^d$

$$\begin{bmatrix} \dot{e}_1 \\ \dot{e}_2 \end{bmatrix} = \begin{bmatrix} -\dfrac{\cos\phi_1}{T_N^m} & T_{122} - T_{111} \\ -\dfrac{\sin\phi_1}{T_N^m} & T_{222} - T_{211} \end{bmatrix} \begin{bmatrix} \upsilon \\ \omega \end{bmatrix} - \begin{bmatrix} \dot{\xi}_1^d \\ \dot{\xi}_2^d \end{bmatrix} = \mathbf{M}(\mathbf{T}, \phi_1)\mathbf{u} - \dot{\xi}^d.$$

Desired trajectories

$$\xi_1^d = \frac{T_{112}^{ini} + T_{121}^{ini}}{2} \left(1 + \cos\left(\frac{\pi}{\tau} t \right) \right)$$
$$\xi_2^d = \frac{T_{212}^{ini} + T_{221}^{ini}}{2} \left(1 + \cos\left(\frac{\pi}{\tau} t \right) \right).$$

- The initial orientation ϕ_1 introduces uncertainty in this system and a robust control law is required.

Visual control – TT based

Position correction: It is carried out by two controllers, because the first one has a singularity problem when the robot is reaching the target location.

- **Sliding mode control** with sliding surfaces:

$$\mathbf{s} = \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} = \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} = \begin{bmatrix} \xi_1 - \xi_1^d \\ \xi_1 - \xi_2^d \end{bmatrix} = \mathbf{0}.$$

- **Decoupling-based controller**

$$\mathbf{u}_{db} = \begin{bmatrix} \upsilon_{db} \\ \omega_{db} \end{bmatrix} = \frac{1}{\mathbf{det}(\mathbf{M})} \begin{bmatrix} \dfrac{T_{222} - T_{211}}{\dfrac{\sin\phi_1}{T_N^m}} & \dfrac{T_{111} - T_{122}}{-\dfrac{\cos\phi_1}{T_N^m}} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

Singularity if $|\det(\mathrm{M})| = 0.$

At the final location

where $\mathbf{det}(\mathbf{M}) = \dfrac{1}{T_N^m}\left[(T_{122} - T_{111})\sin\phi_1 + (T_{211} - T_{222})\cos\phi_1\right], \quad T_N^m = T_{121}^m$

$$u_1 = \dot{\xi}_1^d - \lambda_1 s_1 - \kappa_1 \mathrm{sign}(s_1), \quad u_2 = \dot{\xi}_2^d - \lambda_2 s_2 - \kappa_2 \mathrm{sign}(s_2).$$

- **Bounded controller**

$$\mathbf{u}_b = \begin{bmatrix} \upsilon_b \\ \omega_b \end{bmatrix} = \begin{bmatrix} k_\upsilon \ \mathrm{sign}(s_1) \\ -k_\omega \ \mathrm{sign}(s_2 \ (T_{222} - T_{211})) \end{bmatrix}.$$

- Robust global stabilization of the error system is achieved by commuting from the decoupling controller to the bounded one if $|\mathbf{det}(\mathbf{M})| < T_h$.

Visual control – TT based

- Correction orientation: We can use any single tensor element whose dynamics depends on ω and its final value being zero.

- **Control goal of the step** – Stabilization of the following dynamics

$$\dot{T}_{122} = -T_{112}\omega.$$

- A suitable input ω that yields T_{122} **exponentially stable** is

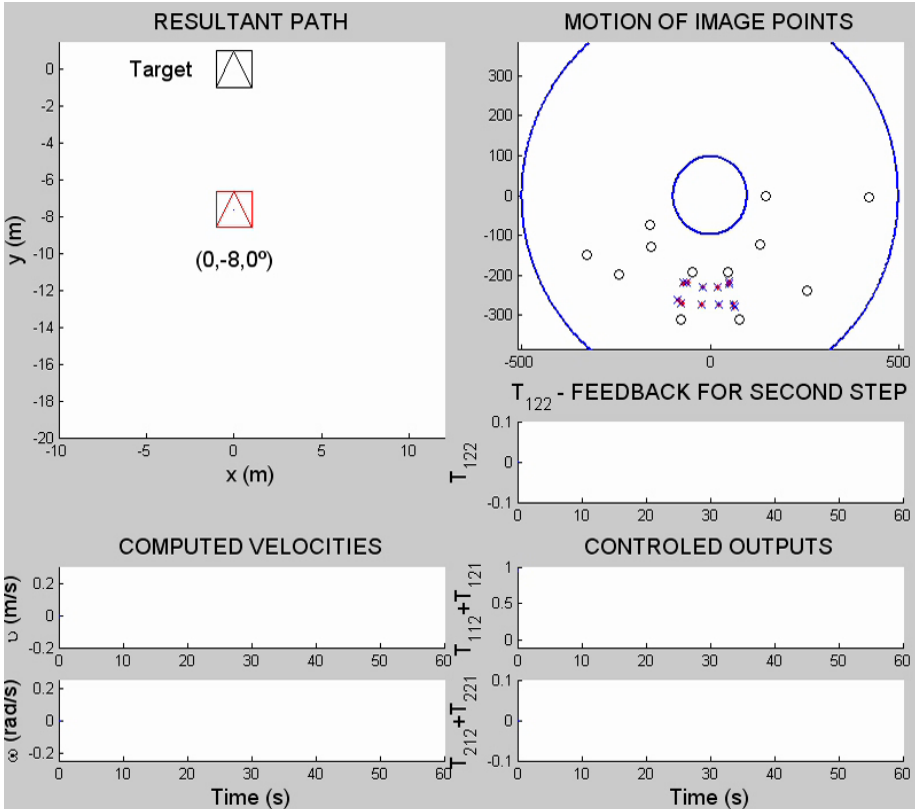
$$\omega = k_{\omega}\frac{T_{122}}{T_{112}}, \qquad t > \tau$$

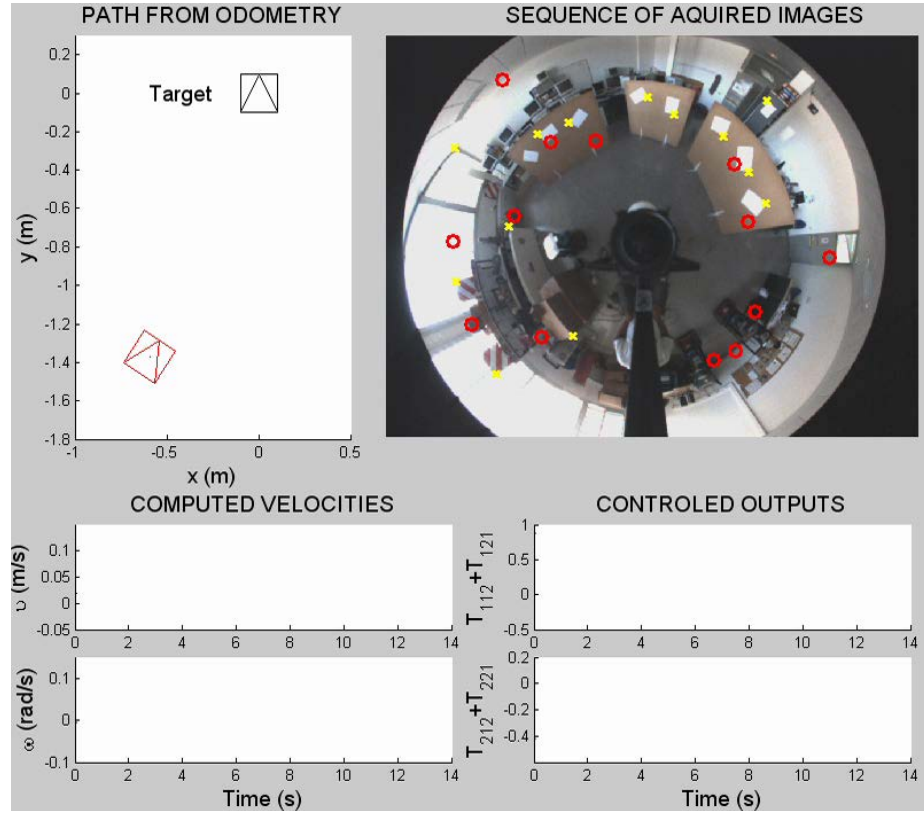
- When position correction has been reached $T_{122} = t_{y_1}\cos\phi_2$, and consequently, if $T_{122} = 0$ then $\phi_2 = n\pi$ with $n \in \boldsymbol{Z}$, and the orientation is corrected.

- Although only a rotation is needed, the same bounded translational velocity is used to maintain the longitudinal position under closed loop control.

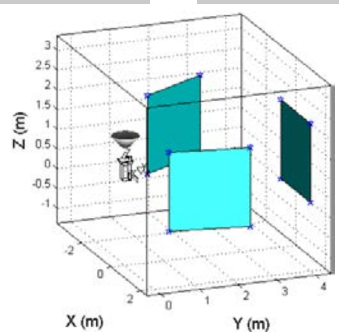
$$\upsilon = k_{\upsilon}\,\mathrm{sign}(s_1).$$

Visual control – TT based

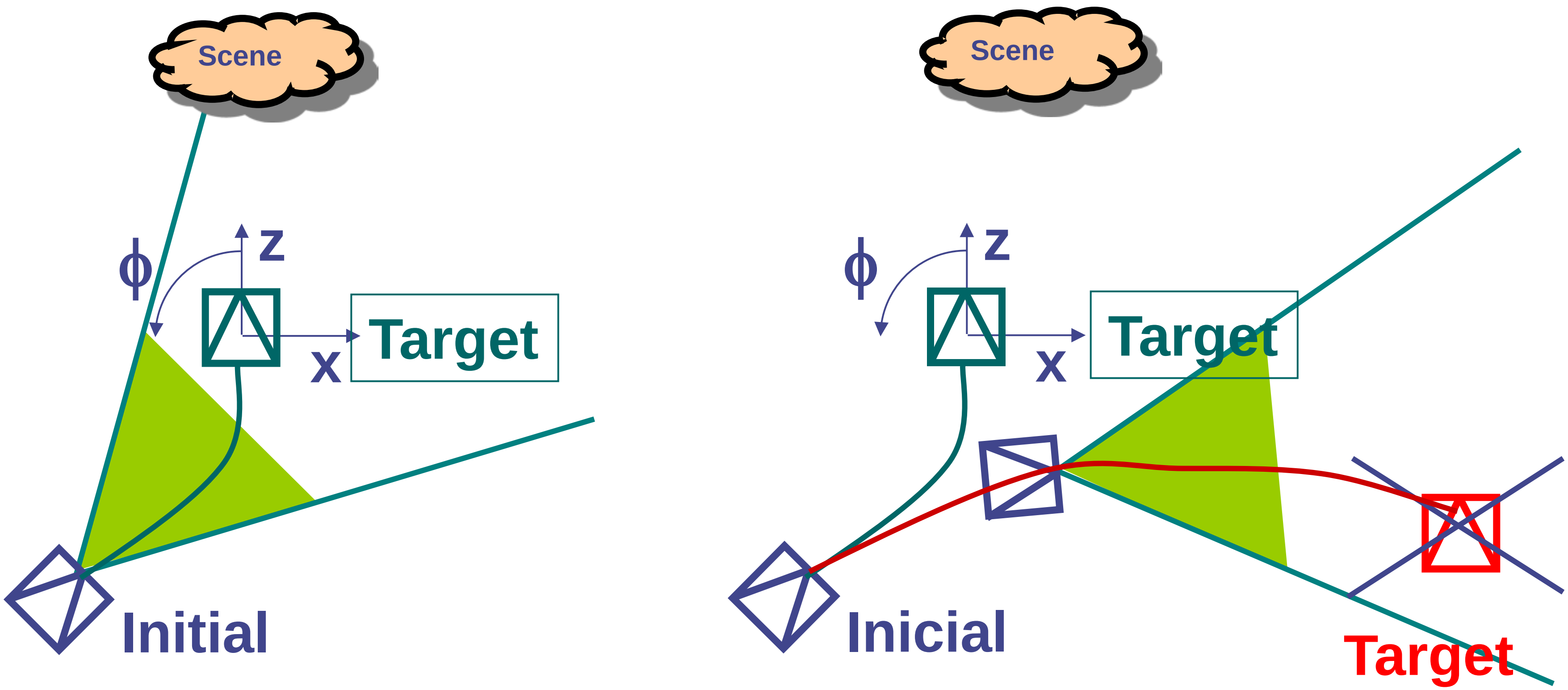




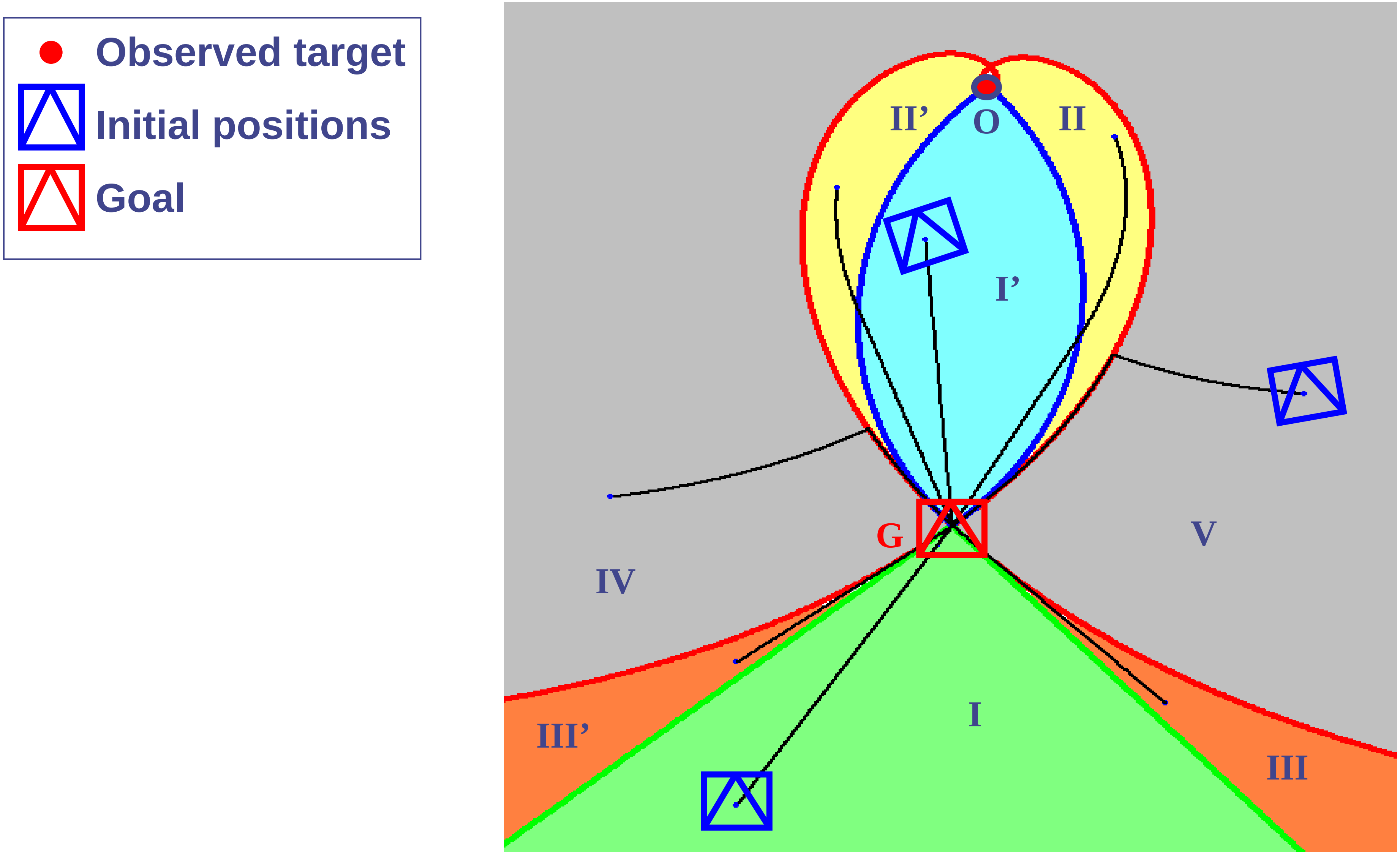
- The 1D-TT is computed from synthetic images of size 1024x768 pixels.
- The desired pose is (0,0,0º).
- Virtual scene:



Visual control with FoV constraints



Visual control with FoV constraints



Visual control with FoV constraints

- **The homography between two views is related to camera motion:**

$$\mathbf{H} = \mathbf{K}\left(\mathbf{R} - \mathbf{t}\,\frac{\mathbf{n}^T}{d}\right)\mathbf{K}^{-1}$$

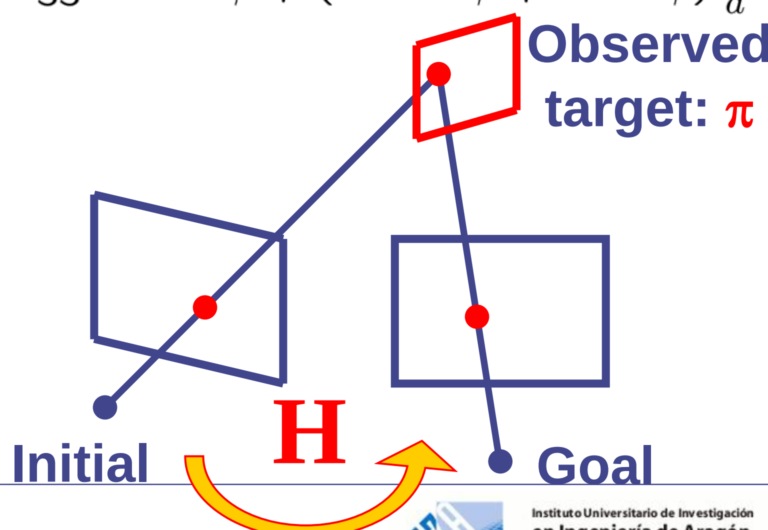
- **Planar motion:**

$$\mathbf{H} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ 0 & 1 & 0 \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$$

With:

$$\begin{cases} h_{11} = \cos\phi + (x\cos\phi + z\sin\phi)\frac{n_x}{d} \\ h_{12} = \frac{\alpha_x}{\alpha_y}(x\cos\phi + z\sin\phi)\frac{n_y}{d} \\ h_{13} = \alpha_x\left(\sin\phi + (x\cos\phi + z\sin\phi)\frac{n_z}{d}\right) \\ h_{31} = \frac{1}{\alpha_x}\left(-\sin\phi + (-x\sin\phi + z\cos\phi)\frac{n_x}{d}\right) \\ h_{32} = \frac{1}{\alpha_y}(-x\sin\phi + z\cos\phi)\frac{n_y}{d} \\ h_{33} = \cos\phi + (-x\sin\phi + z\cos\phi)\frac{n_z}{d} \end{cases}$$

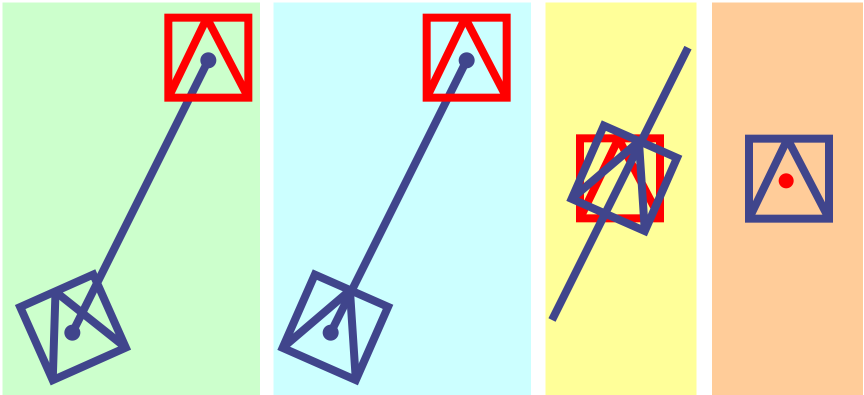
- **Target: Plane of the scene**
- **Goal:** $\mathbf{H} = \mathbf{I}$
- **Subgoals:** $\mathbf{H} = \ldots$



Visual control with FoV constraints

Particular homographies in particular positions

$$\mathbf{H}_{(x,z,\phi)} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ 0 & 1 & 0 \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$$



$$\mathbf{H}_{(x,z,\phi_t)} = \begin{bmatrix} \cos\phi_t & 0 & \alpha_x \sin\phi_t \\ 0 & 1 & 0 \\ \frac{-\sin\phi_t}{\alpha_x} + \frac{z n_x/d}{\alpha_x \cos\phi_t} & \frac{z n_y/d}{\alpha_y \cos\phi_t} & \frac{\cos^2\phi_t + z n_z/d}{\cos\phi_t} \end{bmatrix}$$

$$\mathbf{H}_{(0,0,\phi_t)} = \begin{bmatrix} \cos\phi_t & 0 & \alpha_x \sin\phi_t \\ 0 & 1 & 0 \\ \frac{-\sin\phi_t}{\alpha_x} & 0 & \cos\phi_t \end{bmatrix}$$

$$\mathbf{H}_{(0,0,0)} = \mathbf{I}$$

Visual control with FoV constraints

- **Switched control:**
  Three sequential steps

Step 1:

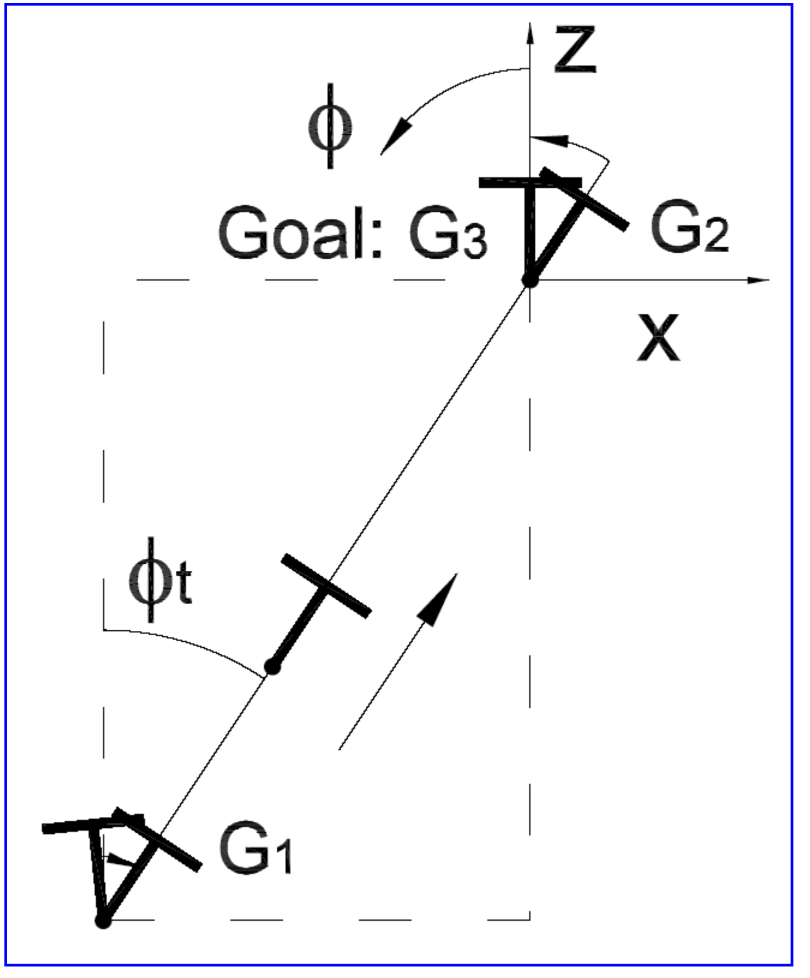
$$\begin{pmatrix} v_1 \\ \omega_1 \end{pmatrix} = \begin{pmatrix} 0 \\ -k_\omega(h_{11}^2 + h_{13}^2/\alpha_x^2 - 1) \end{pmatrix}$$

Step 2:

$$\begin{pmatrix} v_2 \\ \omega_2 \end{pmatrix} = \begin{pmatrix} -k_v(h_{11} - h_{33}) \\ -k_\omega(h_{11}^2 + h_{13}^2/\alpha_x^2 - 1) \end{pmatrix}$$

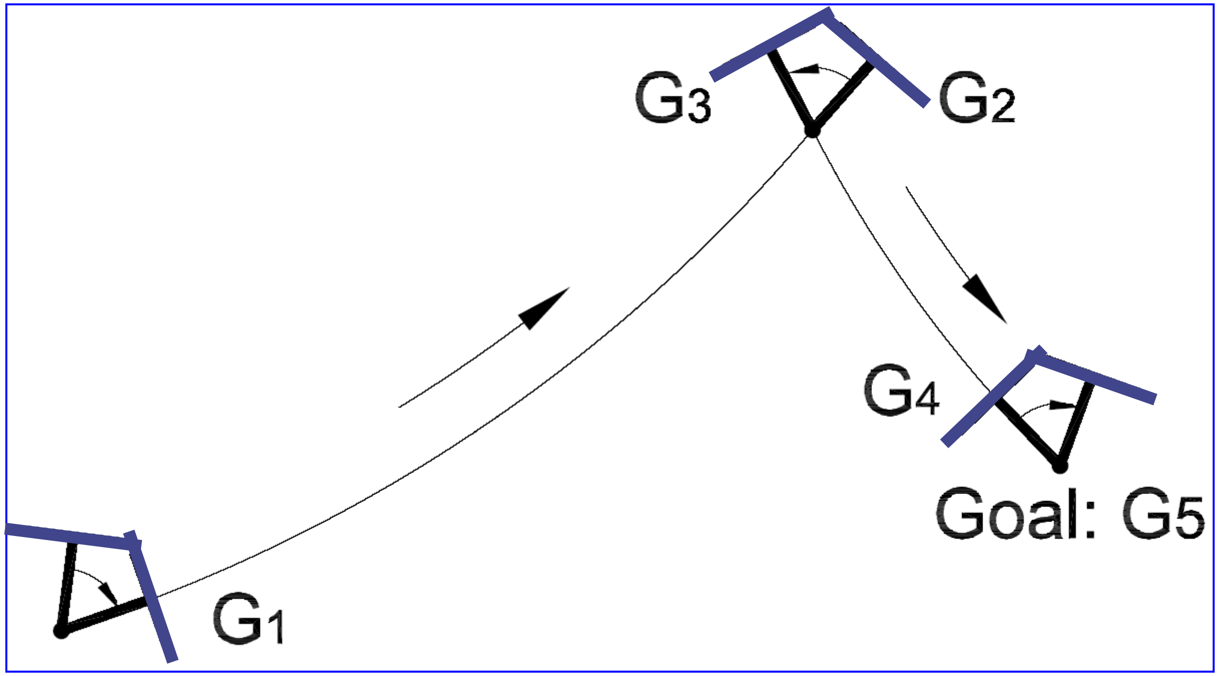
Step 3:

$$\begin{pmatrix} v_3 \\ \omega_3 \end{pmatrix} = \begin{pmatrix} 0 \\ -k_\omega h_{13} \end{pmatrix}$$



Visual control with FoV constraints

- **Switched control: Five sequential steps**



•**Subgoals**

- **G_1: Pure rotation until reaching the first T-curve**
- **G_2: Follow the first T-curve forward**
- **G_3: Pure rotation until reaching the second T-curve**
- **G_4: Follow the second T-curve backward**
- **G_5: Pure rotation until reaching desired Goal**

Visual control with FoV constraints

- **Switched control: Five sequential steps**

Step 1: $$\begin{pmatrix} v_1 \\ \omega_1 \end{pmatrix} = \begin{pmatrix} 0 \\ -k_\omega(h_{13} - h_{13}^{G_1}) \end{pmatrix}$$

Step 2: $$\begin{pmatrix} v_2 \\ \omega_2 \end{pmatrix} = \begin{pmatrix} -k_v(h_{33} - h_{33}^{G2}) \\ -k_\omega(h_{13} - h_{13}^{G2}) \end{pmatrix}$$

Step 3: $$\begin{pmatrix} v_3 \\ \omega_3 \end{pmatrix} = \begin{pmatrix} 0 \\ -k_\omega(h_{13} - h_{13}^{G_3}) \end{pmatrix}$$

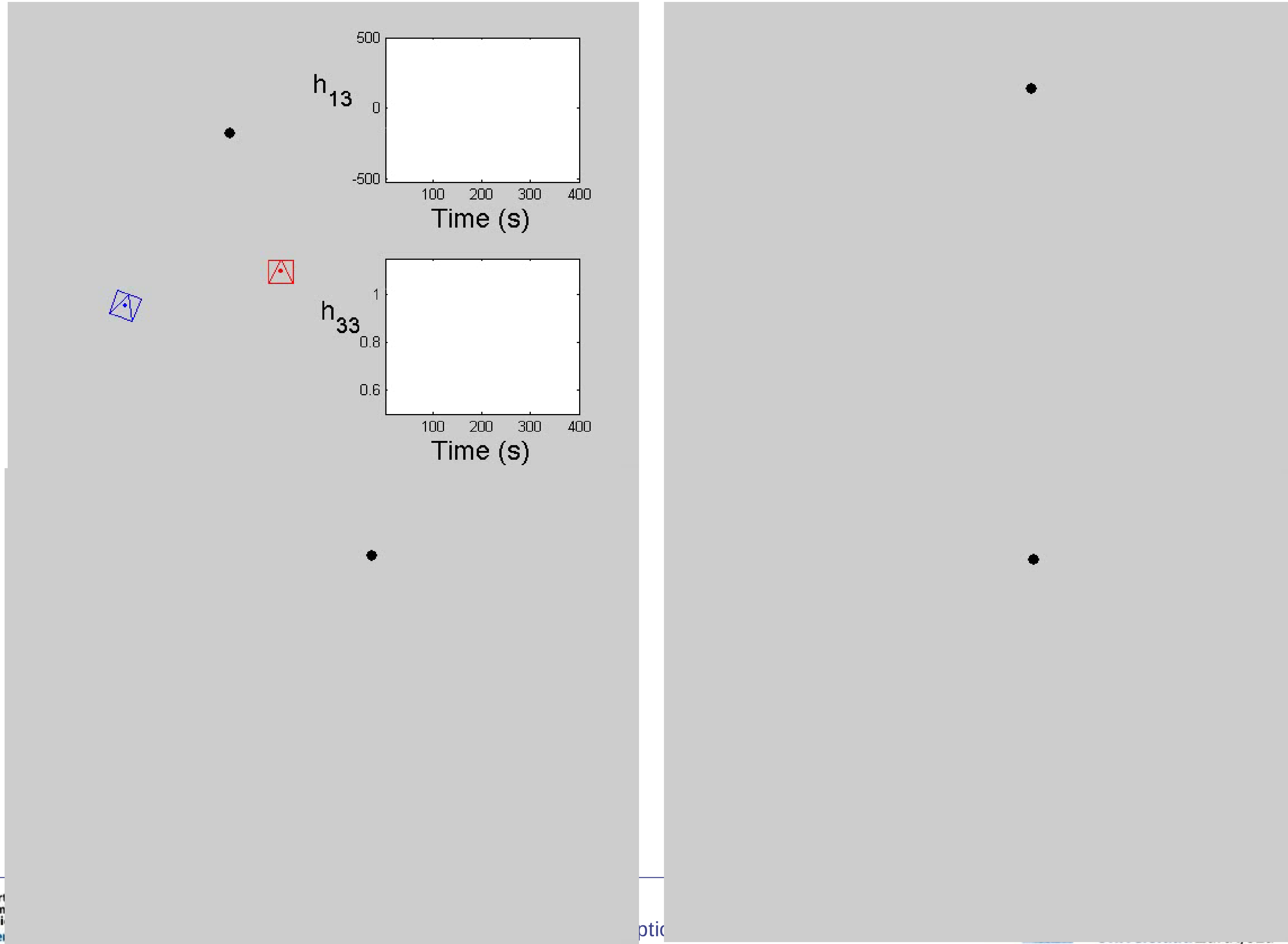
Step 4: $$\begin{pmatrix} v_4 \\ \omega_4 \end{pmatrix} = \begin{pmatrix} -k_v(h_{33} - h_{11}) \\ -k_\omega(h_{13} - h_{13}^{G_4}) \end{pmatrix}$$

Step 5: $$\begin{pmatrix} v_5 \\ \omega_5 \end{pmatrix} = \begin{pmatrix} 0 \\ -k_\omega h_{13} \end{pmatrix}$$

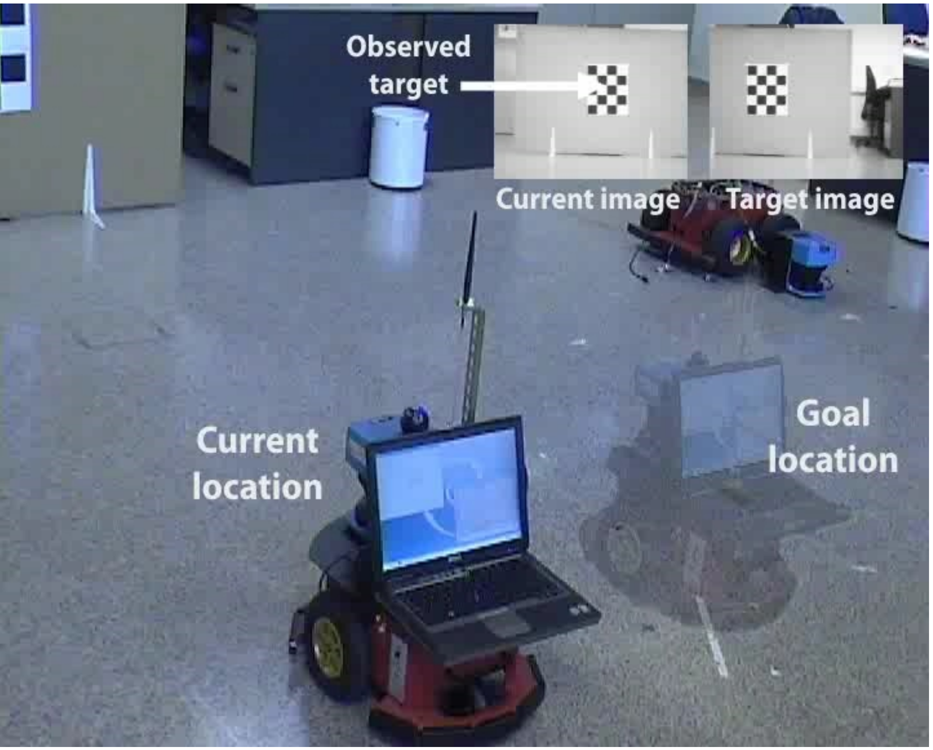
- **Subgoals:**
  - **Defined in terms of homography parameters**
  - **Decomposition of the homography**

$$G_i \quad (i = 1..5) \begin{cases} h_{13}^{G_i} = \dfrac{(\frac{h_{13}}{\alpha_x} - \sin\phi)(\rho^{G_i}\cos\phi^{G_i} + \sin\phi^{G_i})}{(\rho\cos\phi + \sin\phi)\rho_z \,/\, \alpha_x} + \alpha_x \sin\phi^{G_i} \\ h_{33}^{G_i} = \dfrac{(h_{33} - \cos\phi)(-\rho^{G_i}\sin\phi^{G_i} + \cos\phi^{G_i})}{(-\rho\sin\phi + \cos\phi)\rho_z} + \cos\phi^{G_i} \end{cases}$$

Visual control with FoV constraints



Visual control with FoV constraints

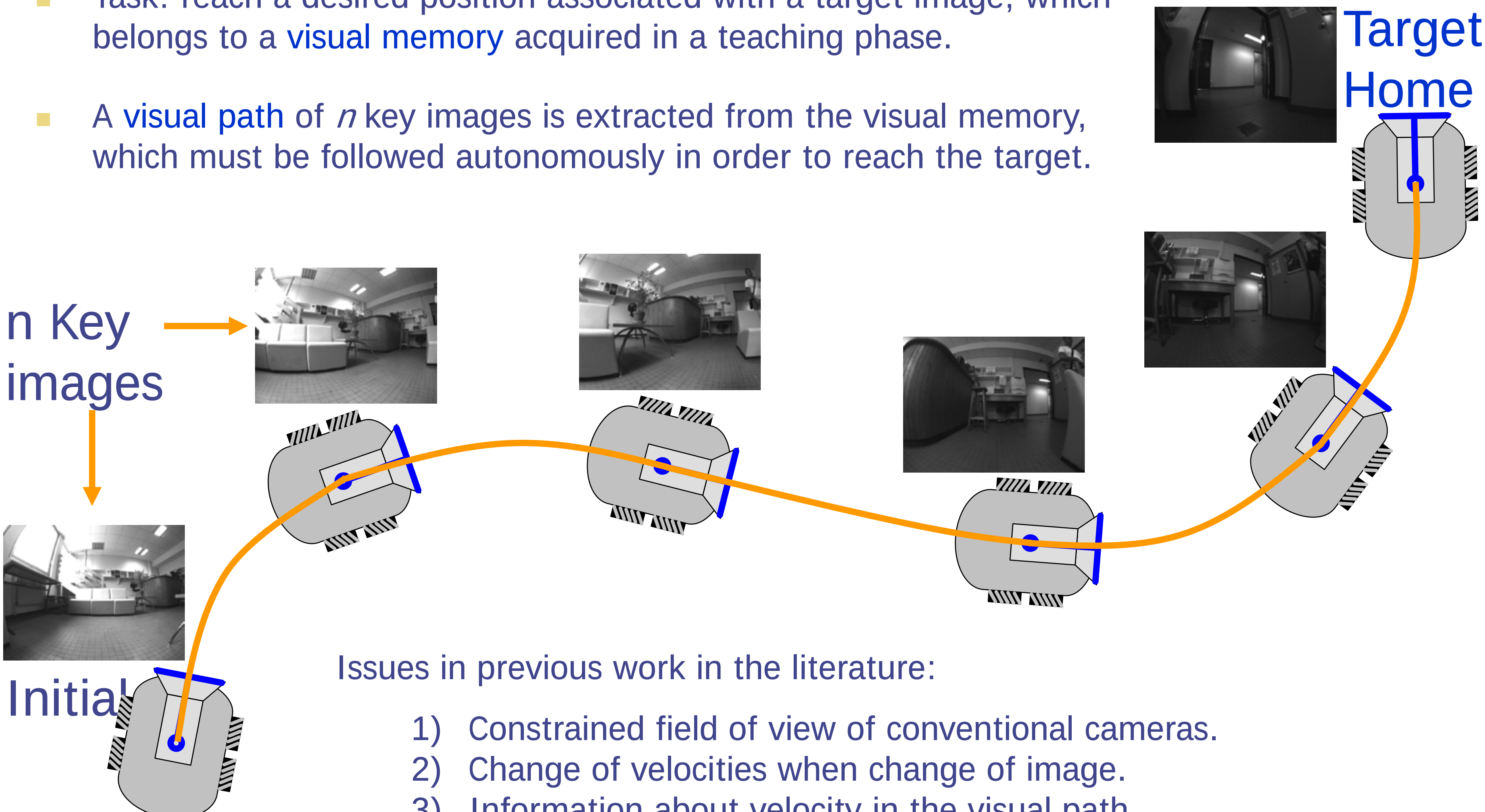


An Optimal Homography-Based Control Scheme for Mobile Robots with Nonholonomic and Field-of-View Constraints

G. López-Nicolás, N. Gans, S. Bhattacharya,
C. Sagüés, J.J. Guerrero and S. Hutchinson

Long term navigation

- Task: reach a desired position associated with a target image, which belongs to a visual memory acquired in a teaching phase.
- A visual path of *n* key images is extracted from the visual memory, which must be followed autonomously in order to reach the target.

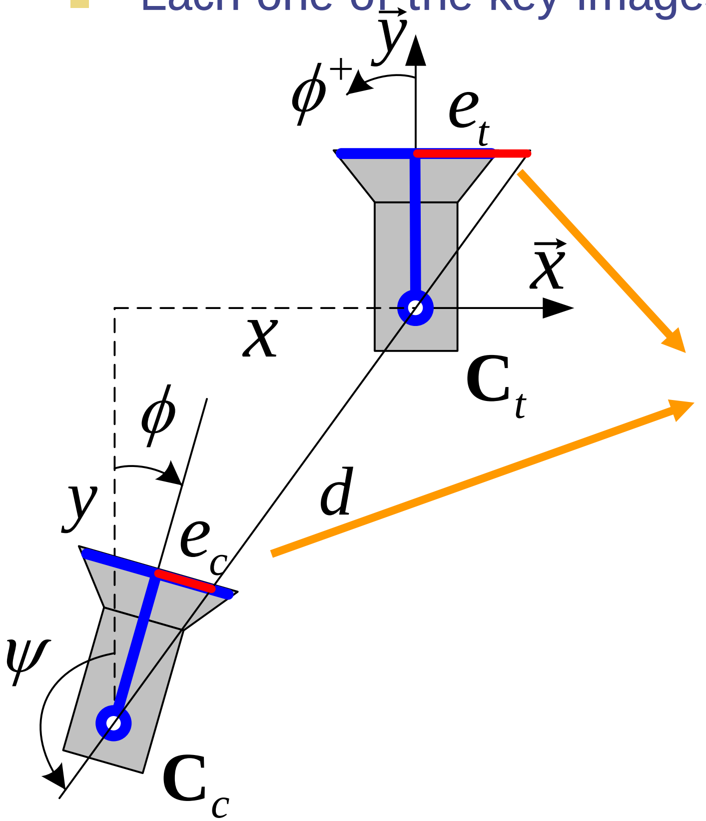


Issues in previous work in the literature:

1) Constrained field of view of conventional cameras.
2) Change of velocities when change of image.
3) Information about velocity in the visual path.

Long term navigation

- The omnidirectional cameras can be virtually represented as conventional cameras when working with points on the sphere.
- Each one of the key images is used as target image accordingly.



Target location ⇨ $\mathbf{C}_t = (0,0,0)$

Current location ⇨ $\mathbf{C}_c = (x, y, \phi)$

Epipoles

$$\begin{cases} e_c = \alpha_x \dfrac{x\cos\phi + y\sin\phi}{y\cos\phi - x\sin\phi}, \\ e_t = \alpha_x \dfrac{x}{y}. \end{cases}$$

- Interaction with the robot velocities:

$$\dot{e}_c = -\frac{\alpha_x \mathbf{sin}(\phi-\psi)}{d\,\mathbf{cos}^2(\phi-\psi)}\upsilon + \frac{\alpha_x}{\mathbf{cos}^2(\phi-\psi)}\omega,$$

$$\dot{e}_t = -\frac{\alpha_x \mathbf{sin}(\phi-\psi)}{d\,\mathbf{cos}^2(\psi)}\upsilon$$

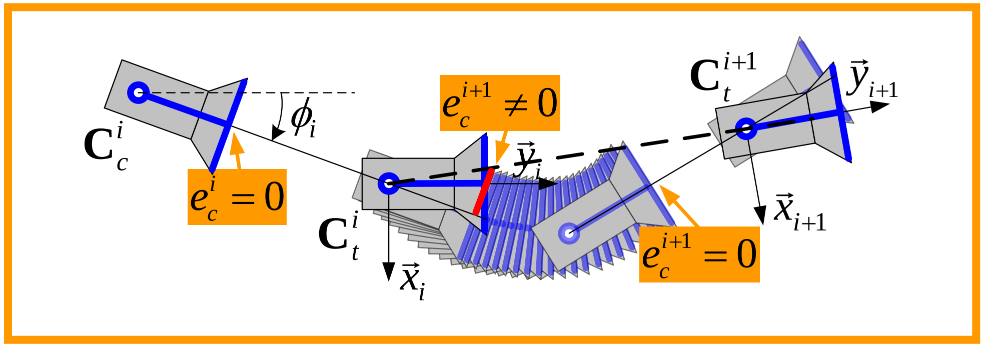
Long term navigation

- The current epipole gives information of the translation direction and it is directly related to the required robot rotation to be aligned with the target.
- Use of the *x*-coordinate of the current epipole as feedback information to control the robot heading and so, to correct the lateral deviation.

Non-null translational velocity $\upsilon \neq 0$

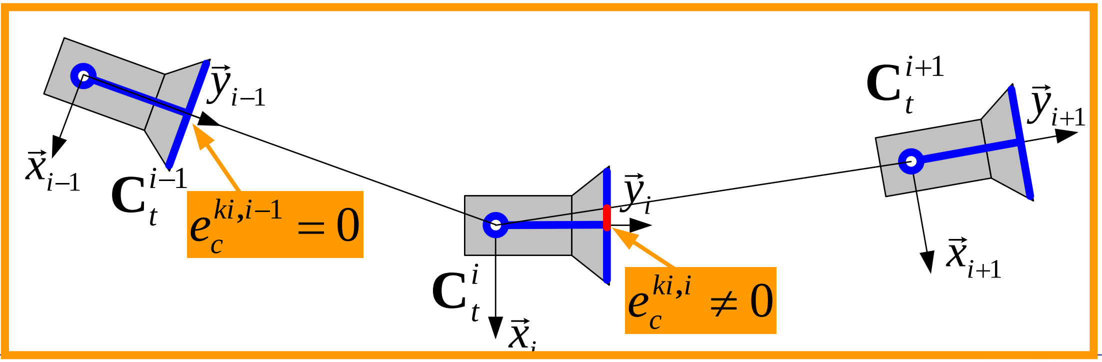
$$\omega^{ce} = k_t \omega_{rt}^{ce} + \overline{\omega}^{ce}.$$

First component of the rotational velocity



$\omega \Rightarrow f(e_c)$

Second component of the rotational velocity



$\omega \Rightarrow f(e_c^{ki})$

Long term navigation

- Let us define a tracking error to drive the epipole smoothly to zero for every segment between key images

$$\zeta_{ce} = e_c - e_c^d(t) = 0.$$

where $e_c^d(t) = \frac{e_c(0)}{2}\left(1 + \cos\left(\frac{\pi}{\tau}t\right)\right), 0 \le t \le \tau$ with $\tau = \frac{d_{\min}}{\upsilon}$.

$e_c^d(t) = 0, \quad t > \tau$

- Control goal – Stabilization of the error system:

$$\dot{\zeta}_{ce} = -\frac{\alpha_x \sin(\phi - \psi)}{d\cos^2(\phi - \psi)}\upsilon + \frac{\alpha_x}{\cos^2(\phi - \psi)}\omega_{rt}^{ce} - \dot{e}_c^d.$$

- Considering that the translational velocity is known, the following rotational velocity, referred as reference tracking (RT) control, stabilizes the error system

$$\omega_{rt}^{ce} = \frac{\sin(\phi - \psi)}{d}\upsilon + \frac{\cos^2(\phi - \psi)}{\alpha_x}\left(\dot{e}_c^d - k_c\zeta_{ce}\right).$$

with $k_c > 0$.

Long term navigation

- A varying translational velocity according to the shape of the path can be computed depending on the epipoles between key images.

$$\upsilon^{ce} = \upsilon_{\mathbf{max}} + \upsilon_{\mathbf{min}} + \frac{\upsilon_{\mathbf{max}} - \upsilon_{\mathbf{min}}}{2} \mathbf{tanh}\left(1 - \frac{\left|e_c^{ki} / d_{\mathbf{min}}\right|}{\sigma}\right).$$

- We propose the following nominal rotational velocity, which is computed from the epipoles between key images:

$$\overline{\omega}^{ce} = \frac{k_m \upsilon^{ce}}{d_{\mathbf{min}}} e_c^{ki}.$$

- So that, the complete rotational velocity (RT+ control) is given as:

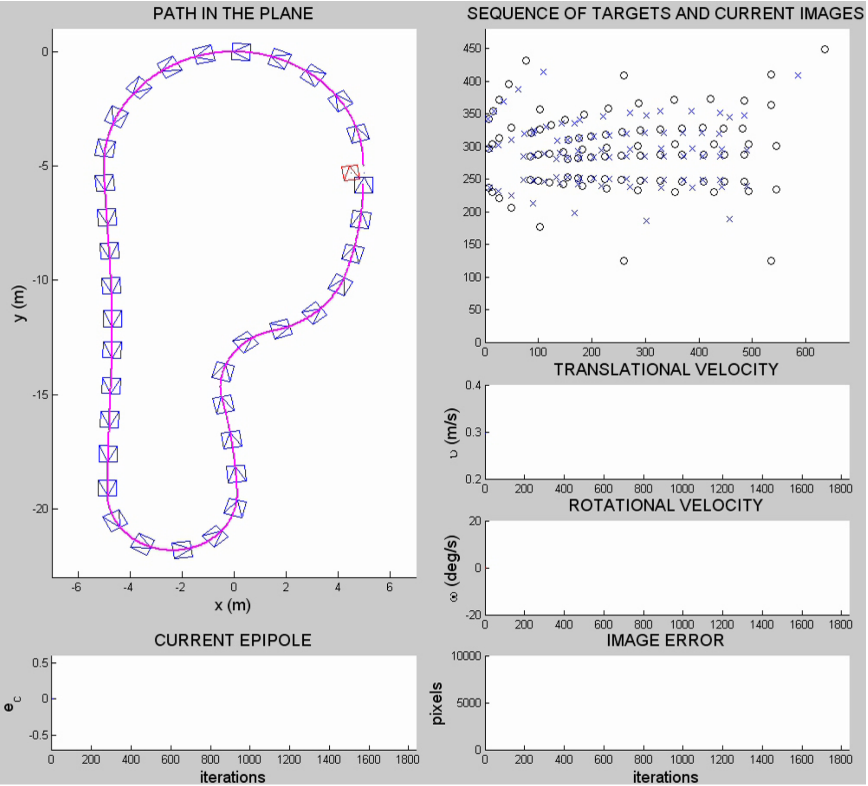
$$\omega^{ce} = k_t \omega_{rt}^{ce} + \overline{\omega}^{ce}.$$

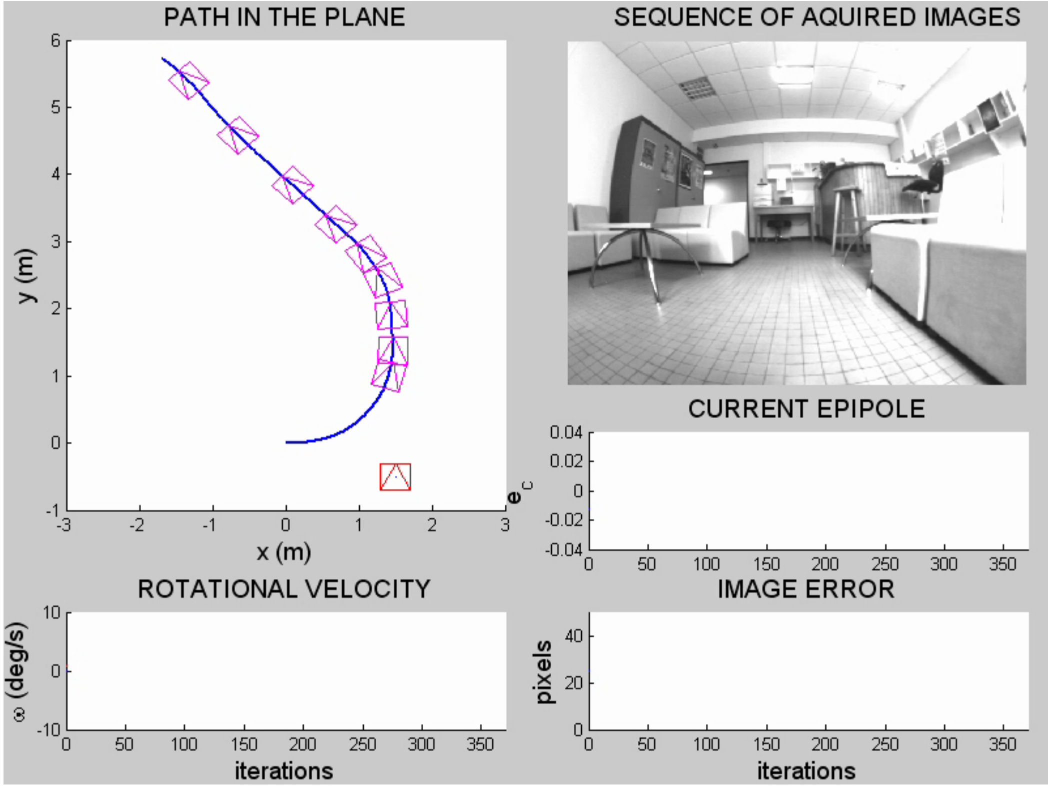
Switching and stop condition

- The switching condition to the next key image or to stop the task is given when the image error starts to increase, which is defined as follows:

$$\varepsilon = \frac{1}{r} \sum_{j=1}^{r} \left\| \mathbf{p}_j - \mathbf{p}_{i,j} \right\|.$$

Long term navigation



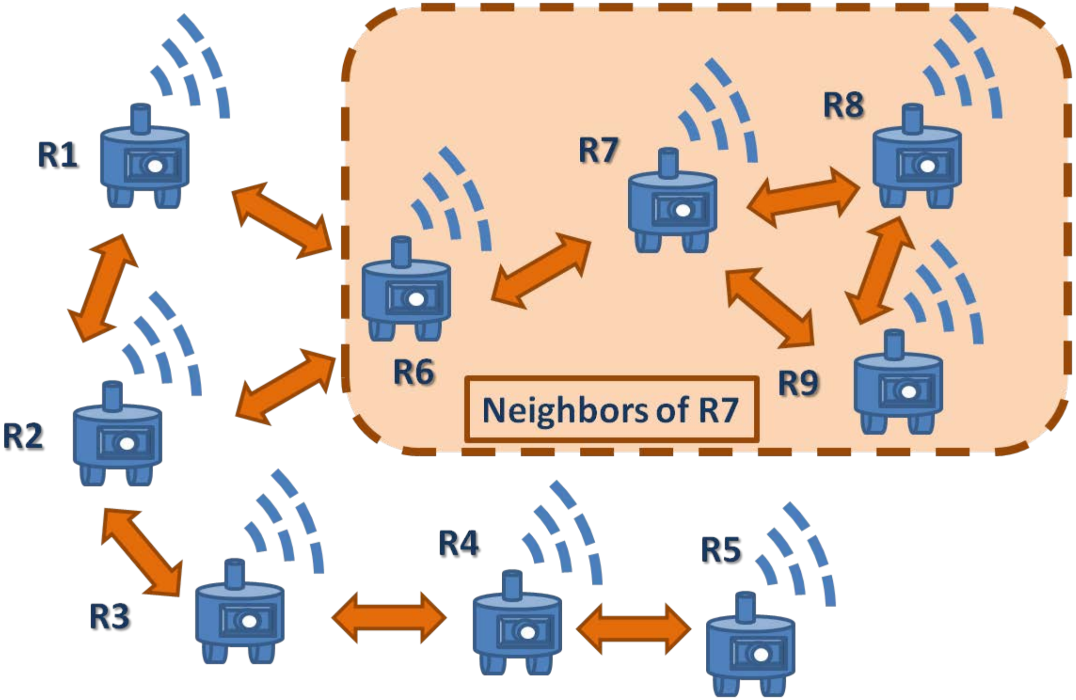


Index

- Features. FM, H, TT (Fundamental Matriz, Homography and Trifocal Tensor)
- Visual mobile robot control
  - FM based
  - H based
  - TT based
  - Long term navigation
- Control of Multi-robot systems
  - Data association
  - Coordinated motion with epipoles
  - Central decision with flying camera on scene - Homography

Multi-Robot Systems

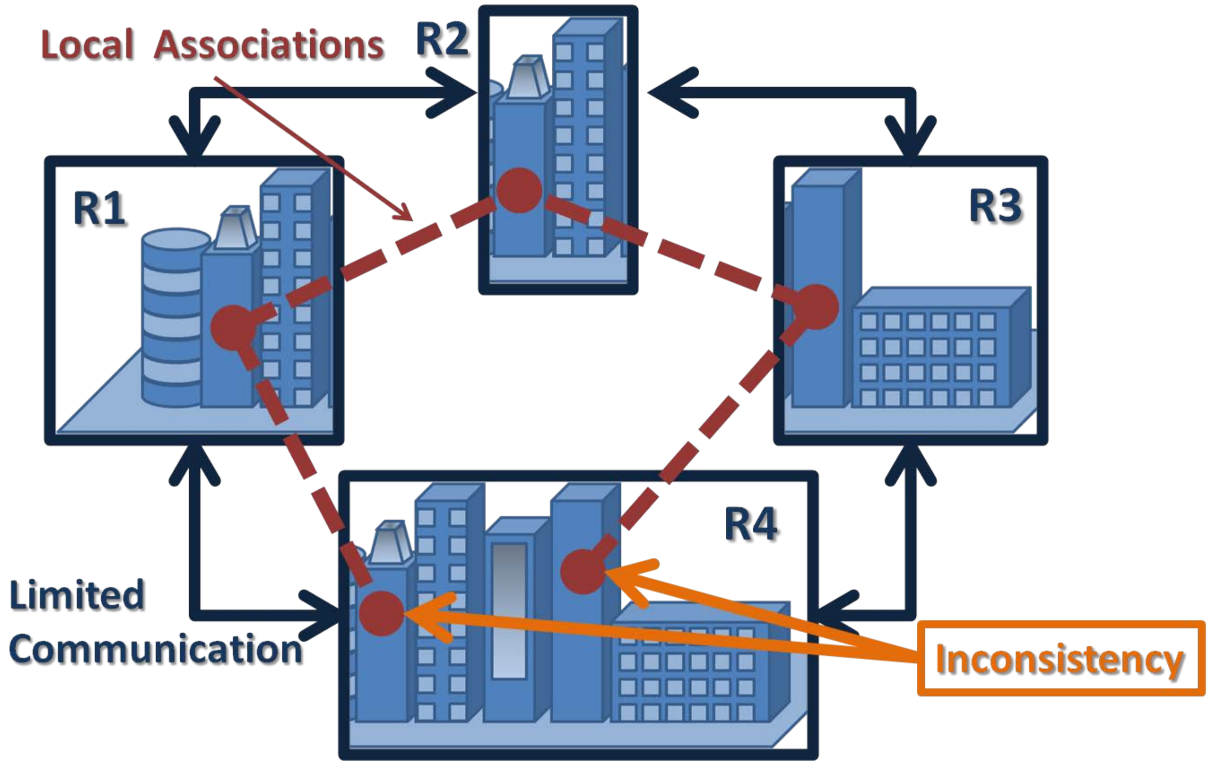
- Robots communication is limited
  - Wireless network.
  - Range-limited.
  - Visibility (Comm.)
- Communication graphs
  - Nodes: the robots
  - Edges: link between robots that can exchange data



- Each robot exchange data with its one-hop neighbors
- Robots are moving: new edges may appear / previous links disappear
  - Communication graphs with switching topology

Distributed Data Association

- Limited communication: Locally associate features with neighbors
- Propagate local associations through the network
- Inconsistent global associations



- Distributed algorithms:
  - propagate local associations
  - detect inconsistencies
  - resolve them
- Additionally, establish global labels for the features

Distributed Data Association

- Each robot $i \in \{1, \ldots, n\}$ in the team has a set $\mathcal{S}_i = \{f_1^i, \ldots, f_{m_i}^i\}$ of m_i features .
- It has executed a local association method F to match its features $\mathcal{S}_i$ and its neighbors' ones $\mathcal{S}_j$, for $j \in \mathcal{N}_i$

$$F(\mathcal{S}_i, \mathcal{S}_j) = \mathbf{A}_{ij} = \mathbf{A}_{ji}^T = (F(\mathcal{S}_j, \mathcal{S}_i))^T \qquad F(\mathcal{S}_i, \mathcal{S}_i) = \mathbf{A}_{ii} = \mathbf{I}$$

$$[\mathbf{A}_{ij}]_{r,s} = \begin{cases} 1 & \text{if } f_r^i \text{ and } f_s^j \text{ are associated,} \\ 0 & \text{otherwise,} \end{cases}$$

$$r = 1, \ldots, m_i \textit{ and } s = 1, \ldots, m_j.$$

- This information can be represented with graph, where the nodes are the features of all the robots, and there is a link between two features if they have been locally matched by F .
- The adjacency matrix of this graph is with

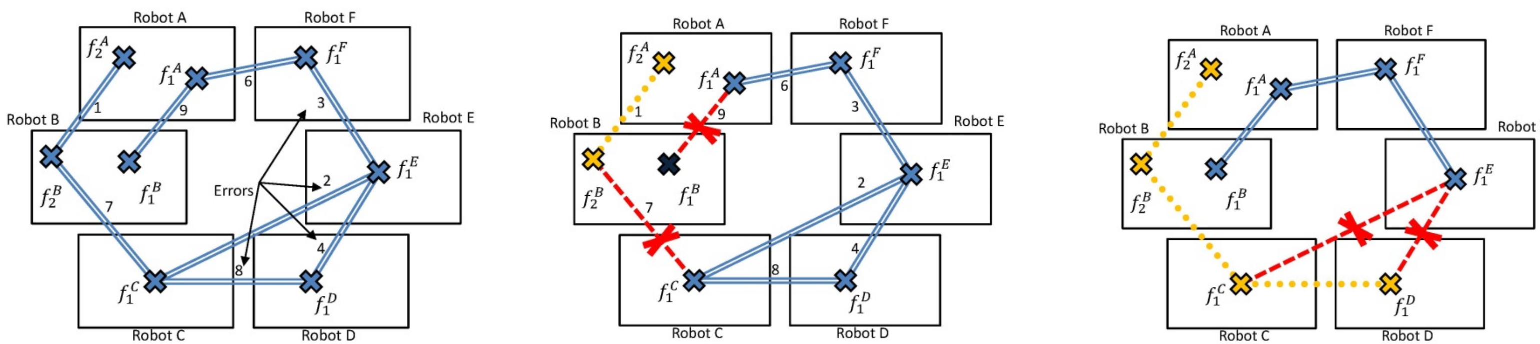
$$\mathbf{A}_{ij} = \begin{cases} F(\mathcal{S}_i, \mathcal{S}_j) & \text{if } j \in \{\mathcal{N}_i \cup i\}, \\ \mathbf{0} & \text{otherwise.} \end{cases} \qquad \mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \ldots & \mathbf{A}_{1n} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{n1} & \ldots & \mathbf{A}_{nn} \end{bmatrix}$$

Distributed Data Association

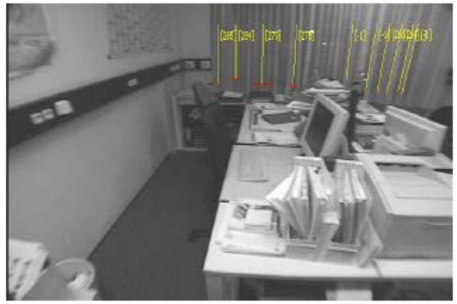
- **Goal** (robot **i**). Discover for each the features f_r^i , all the other features which are connected to f_r^i through a path.
- **Idea**. If there is a link between features f_r^i and f_s^j , then the features connected to f_r^i and to f_s^j through a path are the same.
- **Formal**. Distributed computation of the powers of the adjacency matrix,
  - Each robot maintain $\mathbf{A}^t$ the rows of the adjacency matrix power associated to its own features, and updates them using data from its neighbors
  - For each of this features f_r^i , each robot **i** obtains all f_s^j connected to f_r^i through a path, and detects the **inconsistent** ones.

Distributed Data Association

- **Idea**: break local associations so that there are no two features from the same robot related by a path.
  - Note that each inconsistency is motivated by, at least, one spurious local link (false positives).
- All local links are equal ⟹ Resol. algorithm based on Trees
  - For each conflictive feature belonging to the same robot, use it as root of its tree and incrementally add features linked to it.
  - If a feature already belongs to a tree, or receives requests from more than a tree, it selects one of the trees and erases links to the others.
- Links with quality information ⟹ Maximum Error Cut
  - For each pair of inconsistent features belonging to same robot, select and erase the link with the largest error that breaks the inconsistency.

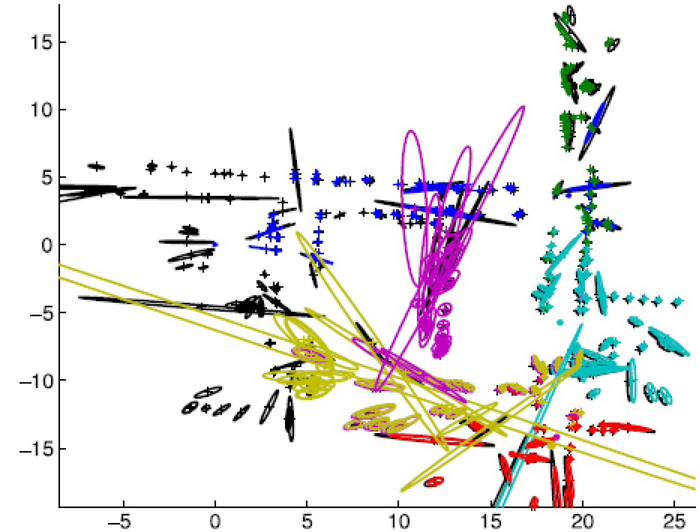


Distributed Data Association

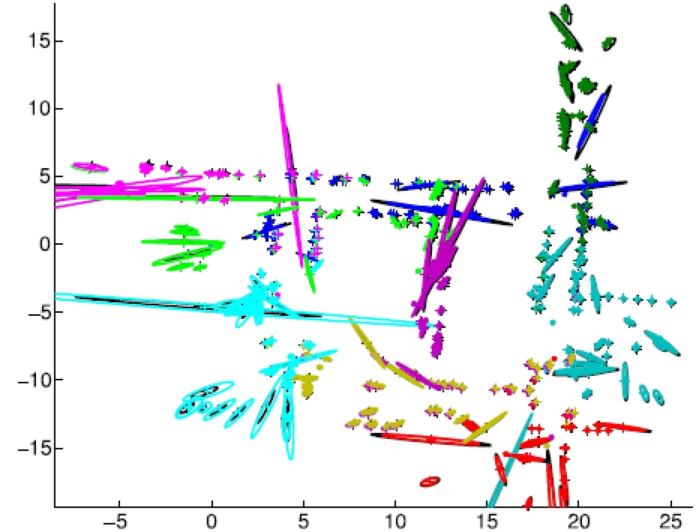


- Compute the robot positions in a common reference frame
- Each robot measures the relative position of its neighbors

- Distributed map merging scenario
  - Local maps aligned before merging
  - It only needs to be computed once

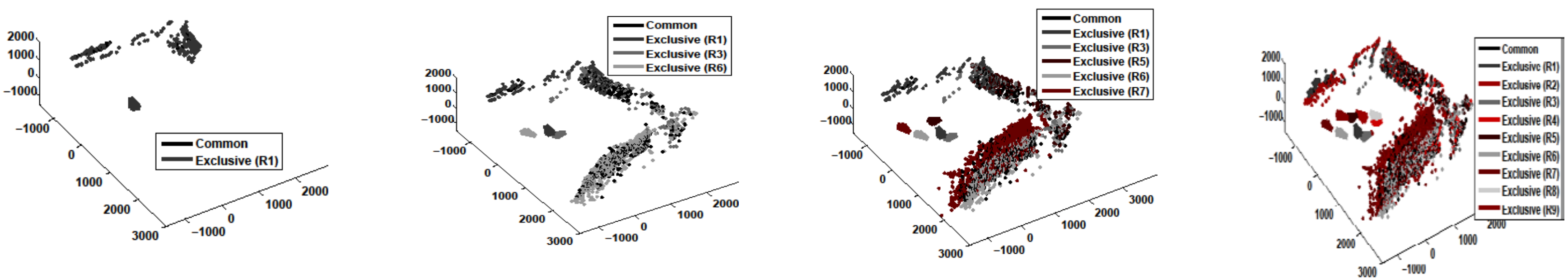


(a) Merged map after 5 iterations



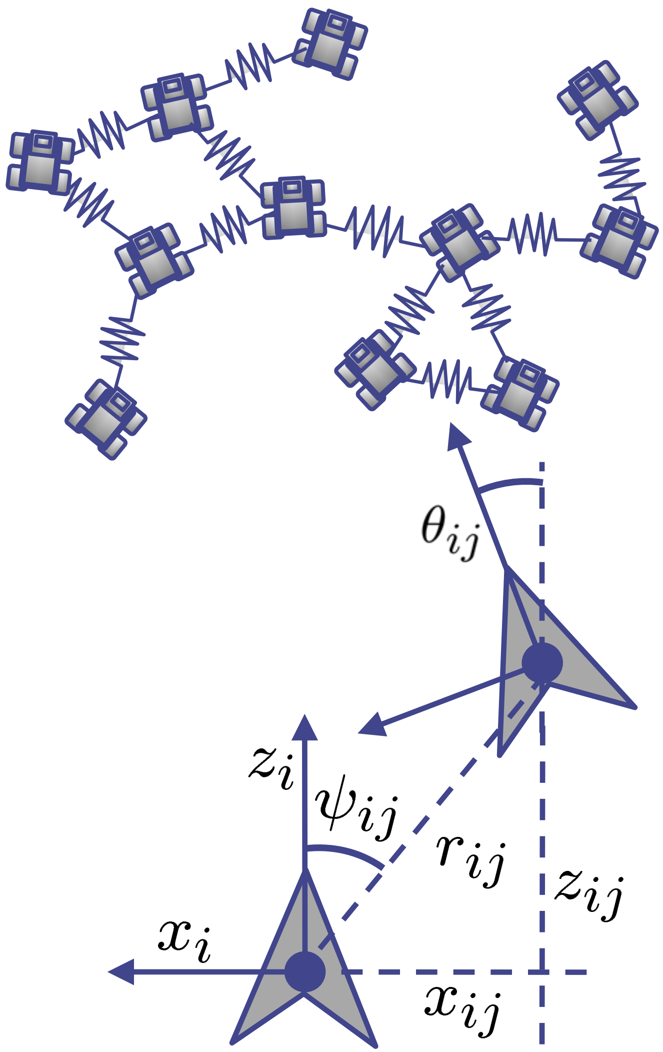
(b) Merged map after 20 iterations

Distributed Data Association



Multi robot control based on epipoles

- Coordinated control for attitude sincronization



Modeled with an undirected graph

$$\mathcal{G} = (\mathcal{V}, \mathcal{E})$$

$$\mathcal{N}_i = \{j \in \mathcal{V} \mid (i,j) \in \mathcal{E}\}$$

Non holonomic motion on the plane

$$\begin{bmatrix} \dot{x}_i \\ \dot{z}_i \\ \dot{\theta}_i \end{bmatrix} = \begin{bmatrix} \sin(\theta_i) & 0 \\ \cos(\theta_i) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_i \\ w_i \end{bmatrix}$$

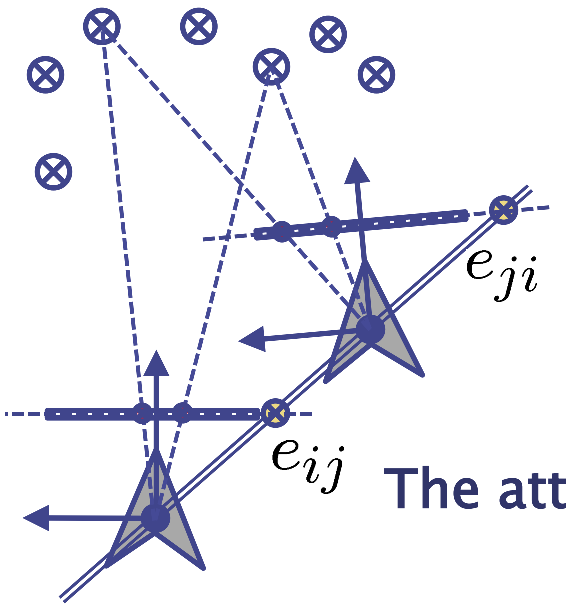
Polar coordinates

$$r_{ij} = \sqrt{x_{ij}^2 + z_{ij}^2} \in \mathbb{R}_{\geq 0},$$

$$\psi_{ij} = \arctan(x_{ij}/z_{ij}) \in (-\pi/2, \pi/2],$$

$$\theta_{ij} = \theta_j - \theta_i \in (-\pi, \pi],$$

Multi robot control based on epipoles



The robots exchange the visual features

Correspondences satisfy the epipolar constraint

$$\mathbf{p}_i^T \mathbf{F}_{ij} \mathbf{p}_j = 0$$

The epipoles are the null space of $\mathbf{F}_{ij}$ and $\mathbf{F}_{ij}^T$

$$e_{ij} = \alpha \tan(\psi_{ij})$$
$$e_{ji} = \alpha \tan(\psi_{ij} - \theta_{ij})$$

The attitude consensus implies the epipoles to be equal

$$\theta_{ij} = 0 \Rightarrow e_{ij} = e_{ji}$$

Note that the opposite is not necessarily true

$$\theta_{ij} = \pi \Rightarrow e_{ij} = e_{ji}$$

Multi robot control based on epipoles

Define

$$d_{ij} = \text{arctan}(\frac{e_{ij}}{\beta}) - \text{arctan}(\frac{e_{ji}}{\beta}) \in (-\pi, \pi], \beta > 0$$

The "geodesic" in the epipole domain

$$w_{ij} = \begin{cases} d_{ij} & \text{if } |d_{ij}| \leq \frac{\pi}{2} \\ -\text{sign}(d_{ij})(\pi - |d_{ij}|) & \text{otherwise} \end{cases},$$

If the calibration is known, then choosing $\beta = \alpha$ the exact relative orientation can be computed and we have a standard consensus problem

Multi robot control based on epipoles

The distributed controller used by the robots is

$$w_i = K \sum_{j \in \mathcal{N}_i} w_{ij}, \ K > 0$$

Properties of the controller

$$w_{ij} = -w_{ji}$$

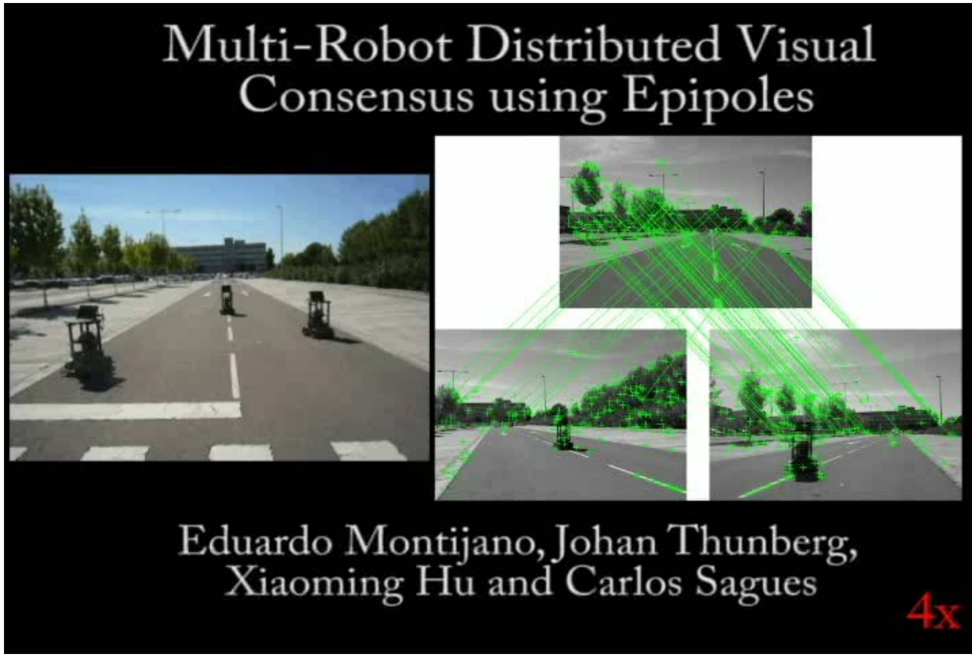
$$\sum_{i \in \mathcal{V}} w_i = 0$$

$$\text{sign}(e_{ij}) = \text{sign}(e_{ji}) \Rightarrow |d_{ij}| < \pi/2$$

Multi robot control based on epipoles

Multi-Robot Distributed Visual Coordination using Epipoles

Eduardo Montijano, Johan Thunberg,
Xiaoming Hu and Carlos Sagues

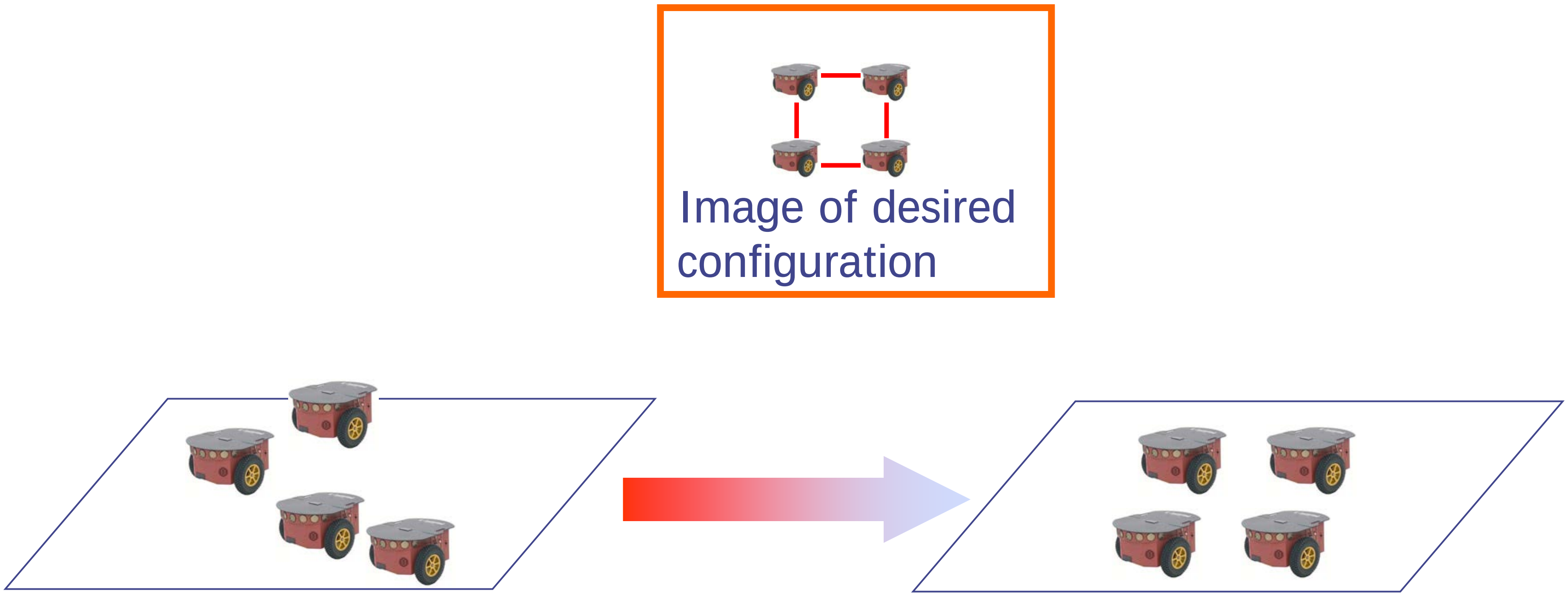


Multi-Robot Distributed Visual Coordination using Epipoles

Eduardo Montijano, Johan Thunberg,
Xiaoming Hu and Carlos Sagues

Multi robot control with flying camera (H)

- What? Visual control of mobile robots
  - Desired configuration defined by an image
  - Task: Navigate to the desired configuration



Initial configuration

Desired configuration

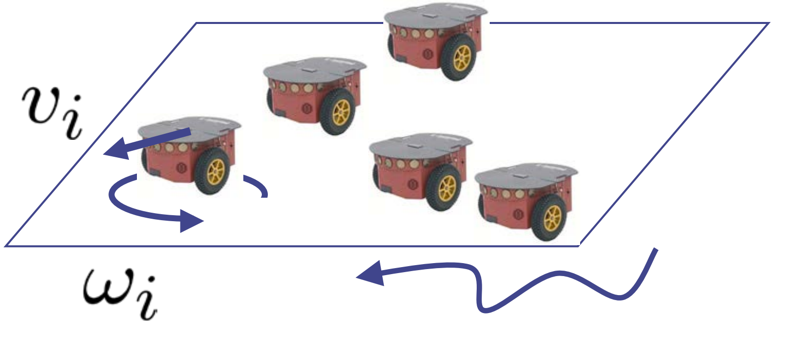
Multi robot control with flying camera (H)

- What? Visual control of mobile robots
- Who? Set of nonholonomic vehicles
  - Nonholonomic kinematics
    - Cartesian coordinates

$$\dot{x} = -v \sin\phi$$
$$\dot{y} = v \cos\phi$$
$$\dot{\phi} = \omega$$

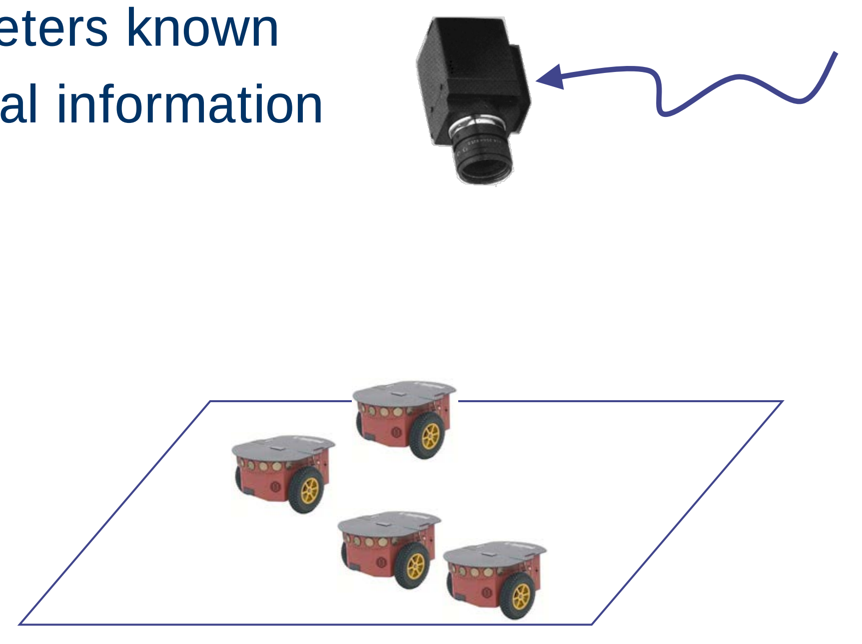
    - Polar coordinates

$$\dot{\rho} = v \cos\alpha$$
$$\dot{\alpha} = \omega - \frac{v}{\rho}\sin\alpha$$
$$\dot{\phi} = \omega$$



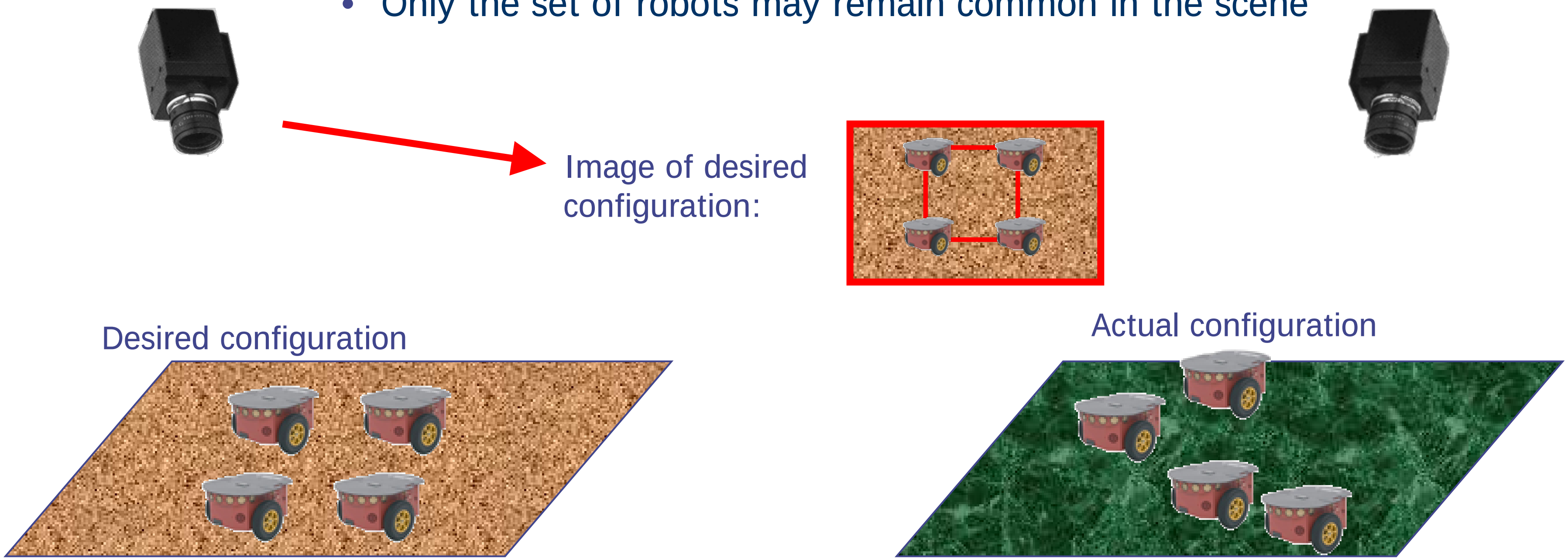
Multi robot control with flying camera (H)

- What? Visual control of mobile robots
- Who? Set of nonholonomic vehicles
- How? Flying camera
    - Flying camera looking downward
    - Camera motion unknown
    - Intrinsic camera parameters known
    - Homography: Only visual information



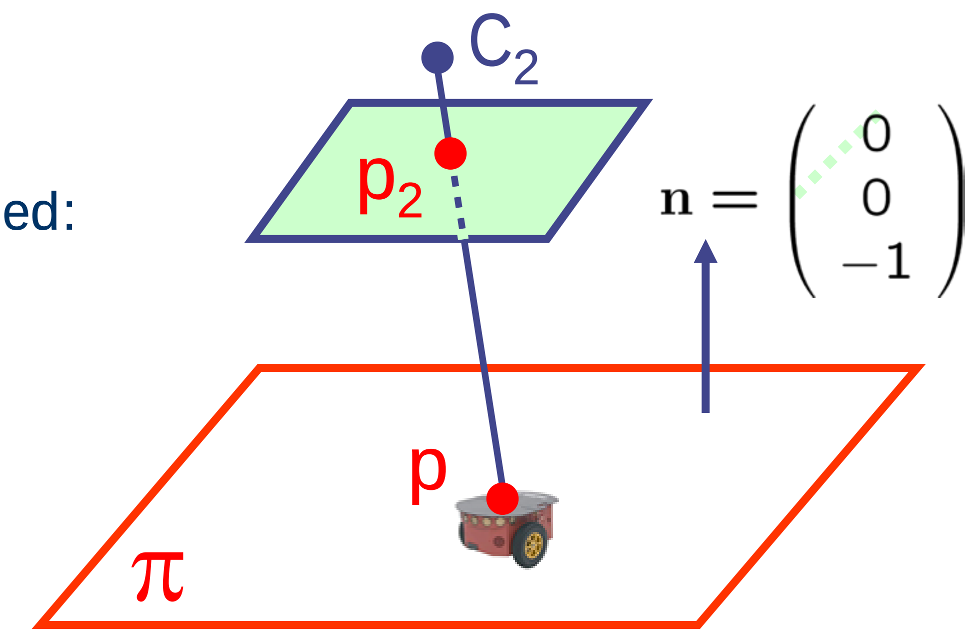
Multi robot control with flying camera (H)

- What? Visual control of mobile robots
- Who? Set of nonholonomic vehicles
- How? Flying camera
- Where? Motion occurs in a planar floor
  - This gives additional constraints on the homography
  - Only the set of robots may remain common in the scene



Multi robot control with flying camera (H)

- The homography in our framework:
  - Multi-robot motion in a planar floor
  - Points = Robots => Homography
  - Camera flies parallel to the floor
- Then, the homography is constrained:



$$\mathbf{H} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{H} = \begin{bmatrix} \cos\phi & \sin\phi & -t_x/d \\ -\sin\phi & \cos\phi & -t_y/d \\ 0 & 0 & 1 \end{bmatrix}$$

- This homography can be computed from a minimal set of two points/robots

Multi robot control with flying camera (H)

$\mathbf{H}_{rigid}$

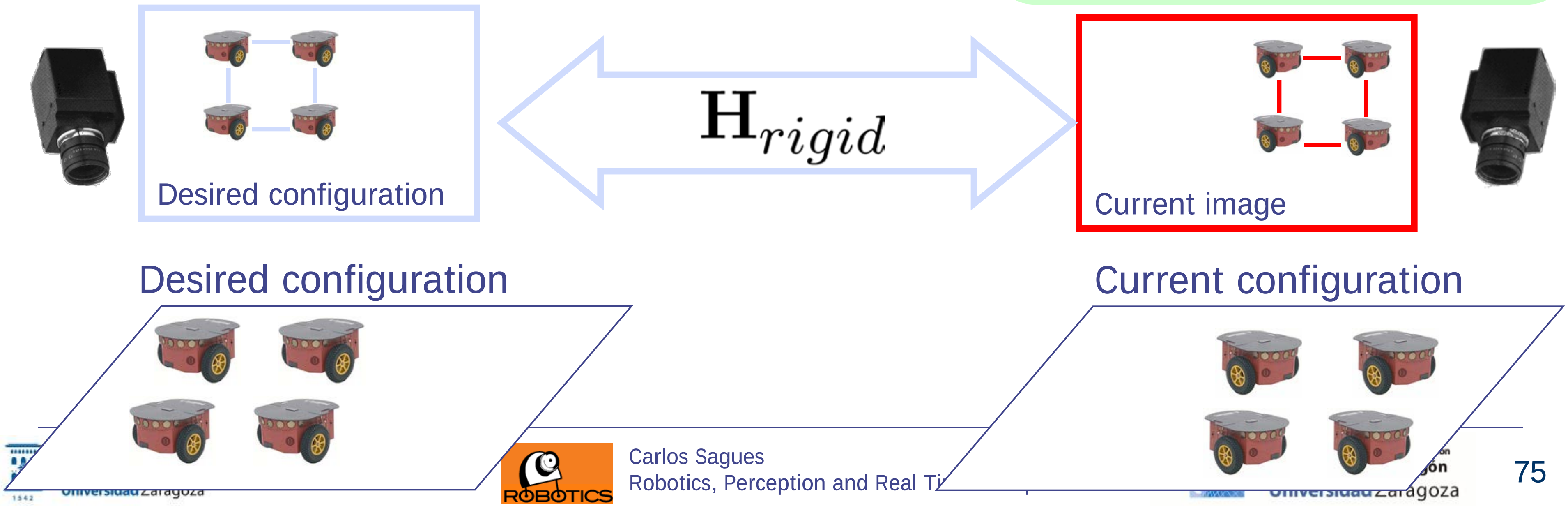
- If the robots are in the desired configuration:
  - The homography is conjugate to a planar Euclidean transformation
  - The homography is not the identity matrix

$$\mathbf{H}_{rigid} = \begin{bmatrix} \cos\phi & \sin\phi & h_{13} \\ -\sin\phi & \cos\phi & h_{23} \\ 0 & 0 & 1 \end{bmatrix}$$

Extracting Motion parameters

$$\mathbf{n} = (0, 0, -1)^T$$
$$\mathbf{x} = (x, y, 0)^T$$

Which is coherent with a rigid motion. So, the robots are in the desired formation



Multi robot control with flying camera (H)

$\mathbf{H}_{nonrigid}$

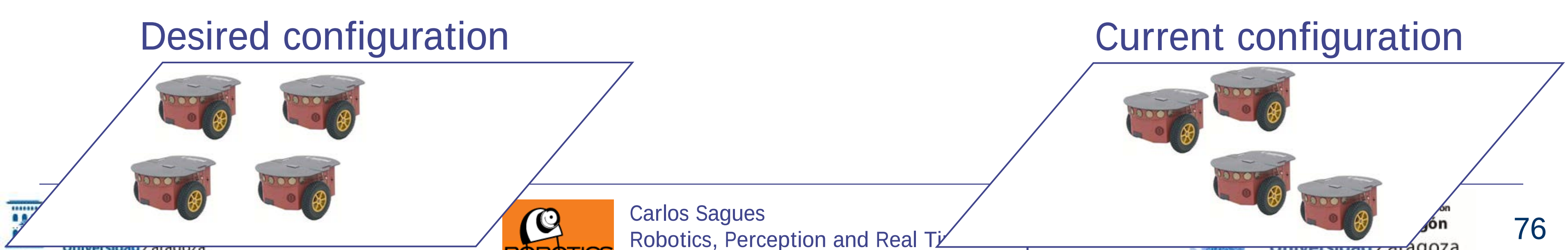
- If the robots are NOT in the desired configuration:
  - The homography is a similarity transformation with isotropic scaling s
  - The H computation with the 2-point method

$$\mathbf{H}_{nonrigid} = \begin{bmatrix} s\cos\phi & s\sin\phi & h_{13} \\ -s\sin\phi & s\cos\phi & h_{23} \\ 0 & 0 & 1 \end{bmatrix}$$

Extracting Motion parameters

$$\mathbf{n} = (0, 0, -1)^T$$
$$\mathbf{x} = (x, y, (s-1)d^2)^T$$

Which is NOT coherent with a rigid motion. So, the robots are not in formation



Multi robot control with flying camera (H)

$\mathbf{H}_{nonrigid}$ — controller → $\mathbf{H}_{rigid}$

- **We have**
  - Robots not in formation
  - Nonrigid homography
  - Each pair of robots induces a different Homography, valid but not coherent

$$\mathbf{H}_{nonrigid} = \begin{bmatrix} s\cos\phi & s\sin\phi & h_{13} \\ -s\sin\phi & s\cos\phi & h_{23} \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{p}' = \mathbf{H}_{nonrigid}\,\mathbf{p}$$

- **We want**
  - Robots in formation
  - Rigid homography
  - Every pair of robots induce the same Homography

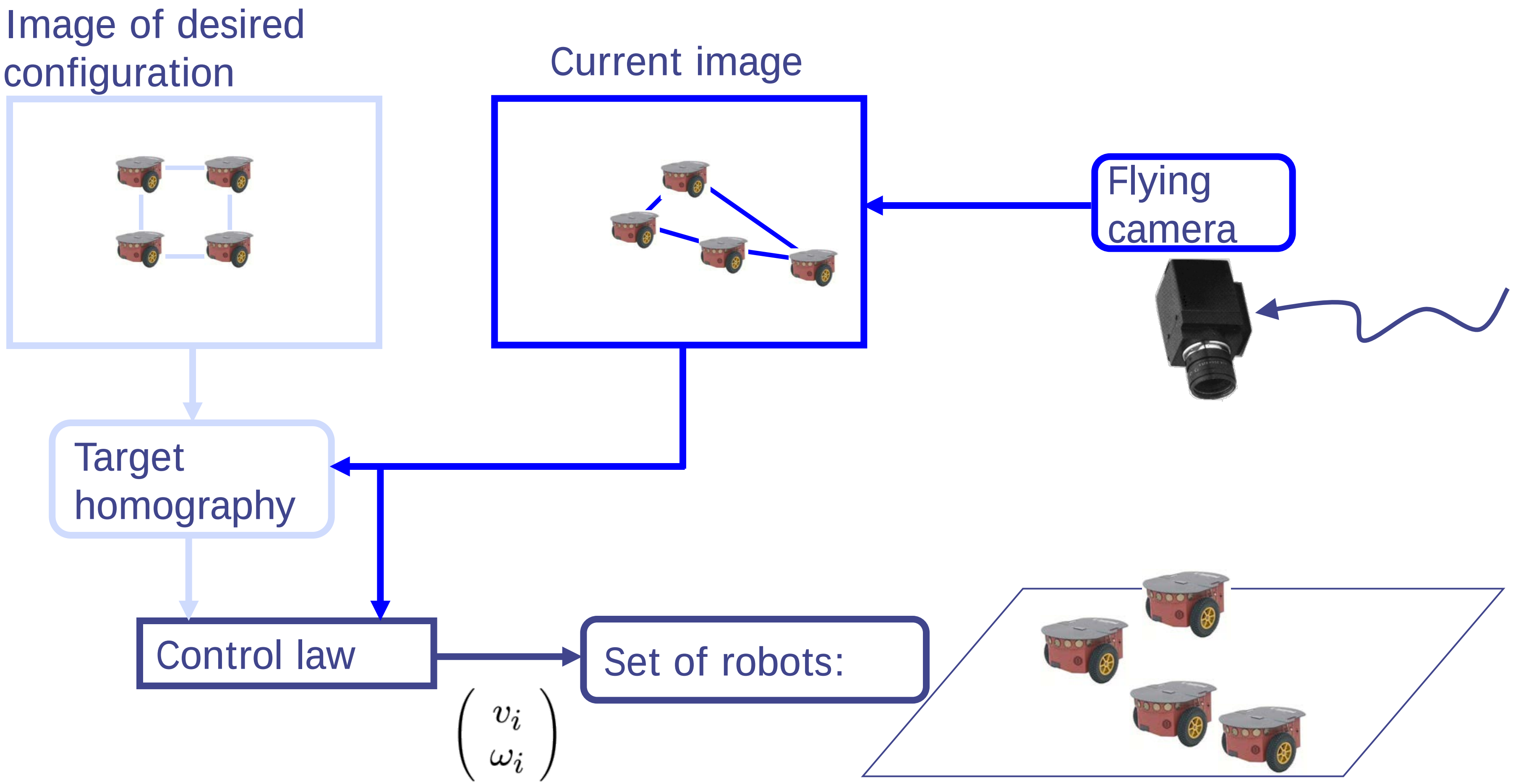
$$\mathbf{H}_{rigid} = \begin{bmatrix} \cos\phi & \sin\phi & h_{13} \\ -\sin\phi & \cos\phi & h_{23} \\ 0 & 0 & 1 \end{bmatrix}$$

- **We define a desired homography**
  - Like the nonrigid homography but being induced by keeping the motion constraints
  - The task is to drive the robots to the desired homography
  - The desired homography is not constant and depends on the robots and camera motion

$$\mathbf{H}^d = \mathbf{H}_{nonrigid} \begin{bmatrix} 1/s & 0 & 0 \\ 0 & 1/s & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{p}^d = (\mathbf{H}^d)^{-1}\,\mathbf{p}'$$

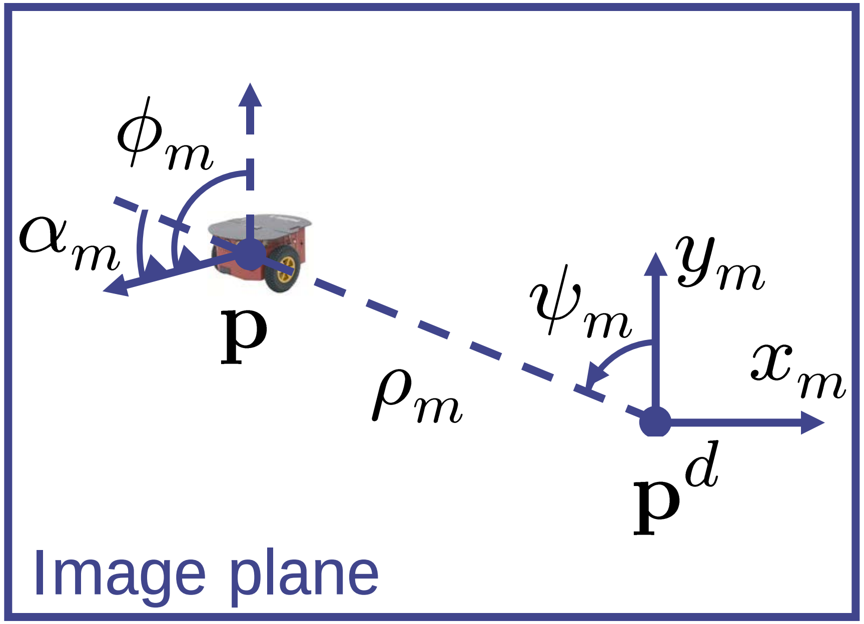
Multi robot control with flying camera (H)



$$\begin{pmatrix} v_i \\ \omega_i \end{pmatrix}$$

Multi robot control with flying camera (H)

- Image-based control law
- Control error:
  - Current state of the robots on the image vs desired states given by the desired homography
- Switched control consisting of three sequential steps:



$$\text{Step 1}\begin{cases} v = 0 \\ \omega = \dot{\psi}_c - k_\omega\,(\alpha_m - \pi) \end{cases}$$

$$\text{Step 2}\begin{cases} v = \dot{\rho}_d - k_v\,\rho_m \\ \omega = \dot{\psi}_c - k_\omega\,(\alpha_m - \pi) \end{cases}$$

$$\text{Step 3}\begin{cases} v = 0 \\ \omega = -k_\omega\,\left((\phi_m - \psi_{Fm}) - (\phi_m^0 - \psi_{Fm}^0)\right) \end{cases}$$

$$\rho_m = \sqrt{(p_x - p_x^d)^2 + (p_y - p_y^d)^2}$$

$$\psi_m = atan2\left(-(p_x - p_x^d), (p_y - p_y^d)\right)$$

$$\psi_{Fm} = atan2\left(-(p_x^i - p_x^j), (p_y^i - p_y^j)\right)$$

$$\mathbf{x}^d(t) = (x^d, y^d, \phi^d)^T$$

$$\dot{\rho}_d = \partial\rho_c/\partial\mathbf{x}^d$$

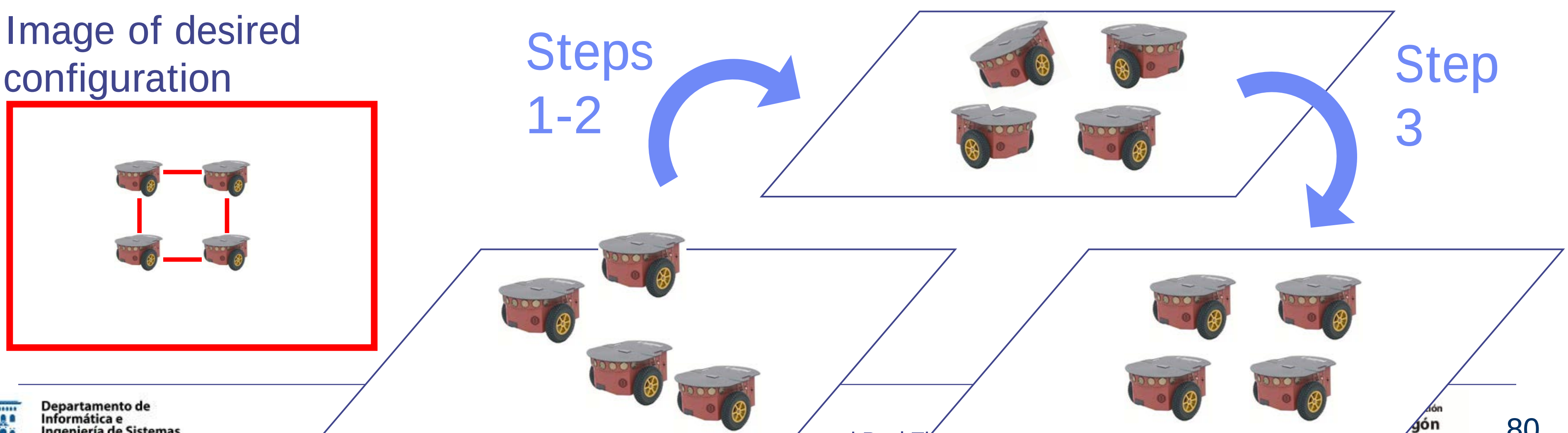
Multi robot control with flying camera (H)

- Steps 1-2 orientate and drive the robots toward their target locations. In practice, they are carried out simultaneously:

$$\text{Step 1 and 2}\begin{cases} v = \dot{\rho}_d - k_v\,\rho_m \\ \omega = \dot{\psi}_c - k_\omega\,(\alpha_m - \pi) \end{cases}$$

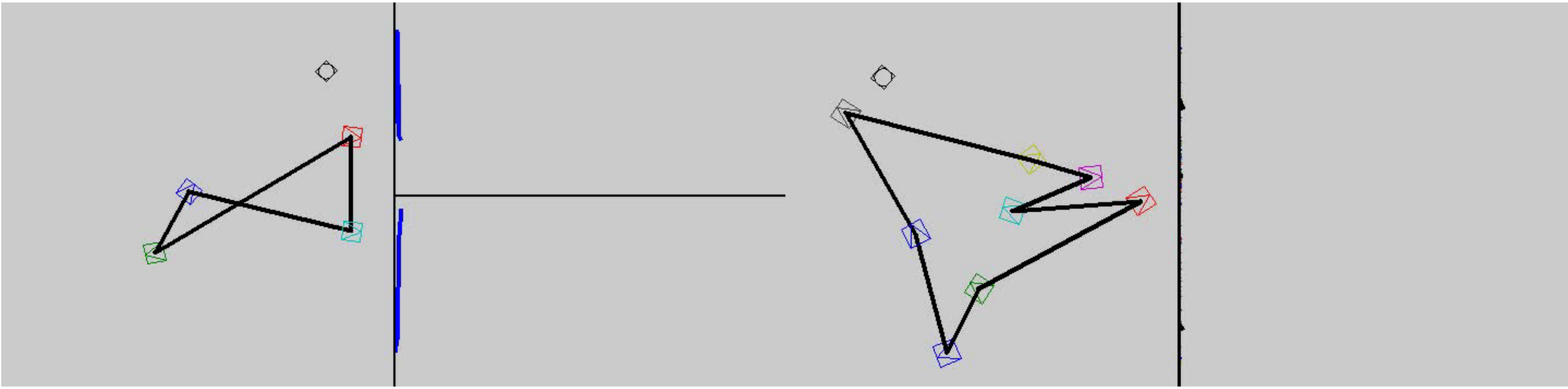
- Step 3 rotates the robots until they are in the required relative orientation within the formation

$$\text{Step 3}\begin{cases} v = 0 \\ \omega = -k_\omega\,\left((\phi_m - \psi_{Fm}) - (\phi_m^0 - \psi_{Fm}^0)\right) \end{cases}$$

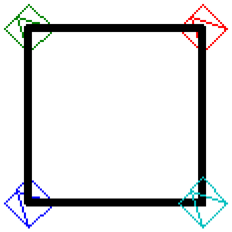


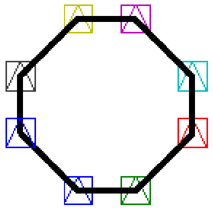
Multi robot control with flying camera (H)

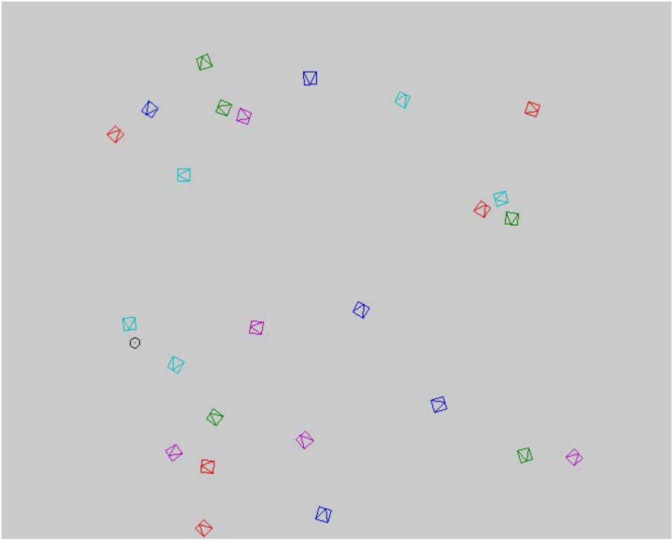
Top view | Linear velocity: v | Top view | Homography entries



Angular velocity: ω

Desired configuration: 

Desired configuration: 



Universidad de Zaragoza

Control de robots y sistemas multi-robot basado en visión

Ciclo de conferencias
Master y Programa de Doctorado en
"Ingeniería de Sistemas y de Control"

UNED – ETS Ingeniería Informática

April -2014

Colaboradores:
Gonzalo López Nicolás
Héctor Manuel Becerra
Rosario Aragüés
Eduardo Montijano
Miguel Aranda

Carlos Sagues
Universidad de Zaragoza
http://www.unizar.es/~csagues