DASSAULT AVIATION

Université Blaise Pascal

CNRS CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE

LASMEA

Topological navigation using sensory memory

SLAN versus SLAM

Dassault Aviation, Saint-Cloud, France, December 16th 2009

Professor Philippe Martinet

J. Courbon, Y. Mezouar, B. Thuilot

CLERMONT UNIVERSITÉ

LASMEA Laboratory

Blaise Pascal University

martinet@lasmea.univ-bpclermont.fr

http://wwwlasmea.univ-bpclermont.fr/Control

http://robots.lasmea.univ-bpclermont.fr

Program of the journey

10H	Accueil et présentation de la journée	B. Patin	*Dassault Aviation*
10H15	Les enjeux et problématiques du SLAM (Survey)	P. Rives	*Inria-Sophia Antipolis*
11H	Initialisation immédiate d-amers en SLAM monoculaire. Points et lignes	Juan Sola	*LAAS*
12H	**Navigation toplogique par mémoire sensorielle: SLAN vs SLAM**	**P. Martinet**	***LASMEA***
14H	Outils et méthodes d'estimation pour le SLAM - Application au Slam visuel	P. Rives	*Inria-Sophia Antipolis*
15H	Détection visuelle de fermeture de boucle et applications au SLAM métrique et topologique	D. Filliat	*ENSTA ParisTech/UEI Lab*
16H		N. Paparoditis	*IGN/MATIS*
17H	Cloture du séminaire	B. Patin	*Dassault Aviation*

Outline of the presentation

Introduction
- *LASMEA*
- *Sensor based Control*
- *Navigation*
- *SLAN vs SLAM*

A generic model
- *A unified model for camera*
- *Case of fisheye cameras*
- *Partial euclidian reconstruction*

Togological navigation system
- *Global system*
- *Case of vision*
- *Differents Modules*
- *SOVIN*

Illustrations
- *Indoor applications*
- *Outdoor applications*

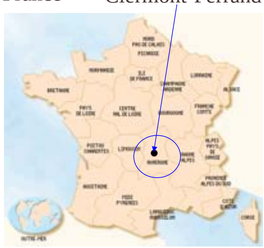
Discussion
- *Learning process*
- *Updating process*
- *Closing the loop problem*

Lasmea LASMEA Director : Michel Dhome

400km south of Paris
200km West of Lyon

CNRS/UBP
Mixed Unit 6602

France — Clermont-Ferrand



Universitary Campus — LASMEA

Lasmea LASMEA Director : Michel Dhome

400km south of Paris
200km West of Lyon

CNRS/UBP
Mixed Unit 6602

MATELEC

- *Optoelectronics, microelectronics*
- *Electromagnetism*
- *Gaz sensor*

33 teacher/researcher
1 research scientist
25 Phd

GRAVIR

- *Perception System*
- *Computer that See*
- *ROSACE*

25 teacher/researcher
2 research scientist,
49 Phd, 2 post-doc

Lasmea : GRAVIR-PERSYST

Perception System

PERception SYSTems — ***J.P. Derutin***

Multisensor Perception
- **Data fusion**
- **Scene understanding**
- ***SLAM***

Smart Sensors
- **Algorithm Architecture Adequation**
- ***Cognitive Vision***

Architecture and methods
- **Parallelism in vision**
- **High speed prototyping tools**
- **Tools for embedded applications**

Lasmea : GRAVIR-COMSEE

Artificial Vision

COMputer that SEE — *T. Chateau*

http://comsee.univ-bpclermont.fr

Geometry for visual perception — Metrology by vision; Automatic rigid scene reconstruction

Vision in Deformable Environments — Modeling and algorithm for deformable environment

Visual Recognition — Real time object tracking; Object recognition

Lasmea : GRAVIR-ROSACE

ROSACE

P. Martinet & Y. Mezouar

RObotics and Autonomous Complex systEms

http://wwwlasmea.univ-bpclermont.fr/Control
http://robots.lasmea.univ-bpclermont.fr

VISIR — *VIsual ServoIng of Robots*
- Omnidirectional visual servoing — *15 to 60Hz*
- Topological navigation through sensory memory
- Multi-sensor-based control

AGV — *Automatic Guided Vehicles*
- Control design under uncertain dynamics
- Hybrid control architecture — *15 to 60Hz*
- Multi-robot-system control — …

MICMAC — *Modeling, Identification and Control of Complex Machines*
- Vision-Based Control of Parallel Robots — *400Hz*
- Control of redundant robots — *1kHz*
- Control of High Dynamics System — …

Lasmea : GRAVIR

People involved in SLAM or near SLAM problems

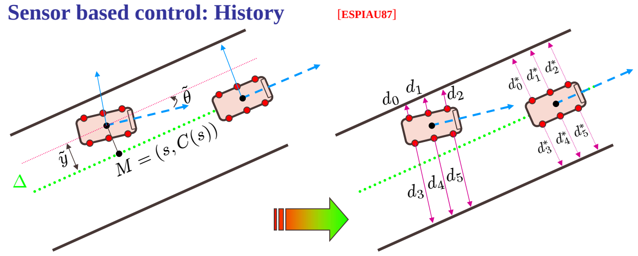
PERSYST — **F. Berry, R. Chapuis, F. Chausse, P. Checchin, J.P. Derutin, L. Trassoudaine**
(IMPALA/CITYVIP)
(CITYHOME/EURIPIDE/BRI)

COMSEE — **M. Dhome, M. Lhuillier , E. Royer**
(MOBIVIP/CITYVIP/BODEGA)
(FACT/CITYHOME/EURIPIDE/BRI)

ROSACE — **Y. Mezouar, P. Martinet, B. Thuilot**
(MOBIVIP/CITYVIP/BODEGA/R-DISCOVER/ARMEN)
(SAFEMPOVE, FACT/CITYHOME/EURIPIDE/BRI/)

Applications
- Autonomous navigation (RTK-GPS, Vision, Multi-sensor)
- Platoon: Autonomous leader (RTK-GPS, Vision, Multi-sensor)
- Platoon: Leader manually driven (RTK-GPS, Vision)
- 2D/3D maps building (vision, Radar, Range finder, Multi-sensor)

Sensor based control: History [ESPIAU87]



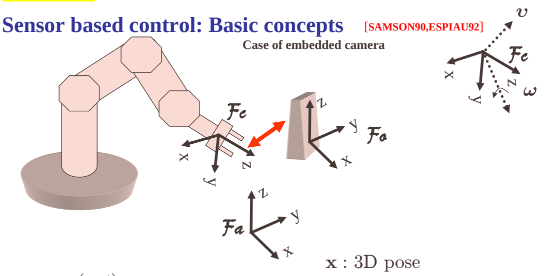
$d = (d_0, d_1, d_2, d_3, d_4, d_5)^T$

Lateral control in cartesian space: Bringing $(\tilde{y}, \tilde{\theta})$ to $(0,0)$. Explicite declaration of Δ and required cartesian localization

Lateral control in sensor space: Bringing d to d^*. No explicite declaration of Δ and do not required cartesian localization

Sensor based control: Basic concepts [SAMSON90,ESPIAU92]

Case of embedded camera



$\mathbf{x}$: 3D pose

$$\mathbf{s} = \mathbf{s}(\mathbf{x}, t)$$

$$\dot{\mathbf{s}} = \frac{\partial \mathbf{s}}{\partial \mathbf{x}}\frac{d\mathbf{x}}{dt} + \frac{\partial \mathbf{s}}{\partial t}$$

$$\dot{\mathbf{s}} = \frac{\partial \mathbf{s}}{\partial \mathbf{x}}\mathbf{v} + \frac{\partial \mathbf{s}}{\partial t}$$

Interaction matrix

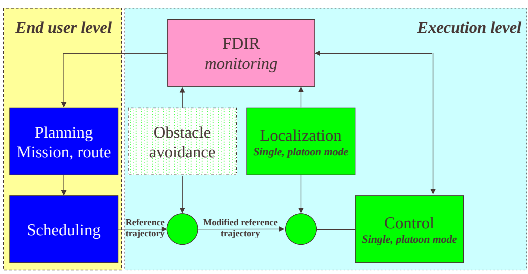
$$\dot{\mathbf{s}} = \mathbf{L}_s \mathbf{v} + \frac{\partial \mathbf{s}}{\partial t}$$

If (fixed object) $\frac{\partial \mathbf{s}}{\partial t} = 0$

$$\dot{\mathbf{s}} = \mathbf{L}_s \mathbf{v}$$

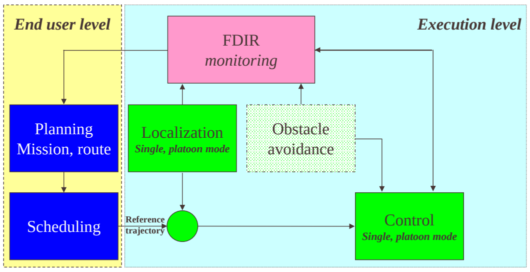
Navigation

Navigation scheme



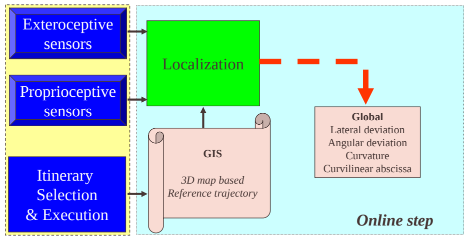
Navigation

Navigation scheme



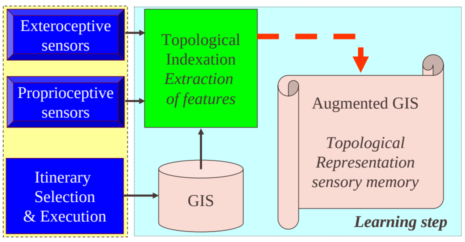
Navigation

3D Map based navigation (absolute navigation)



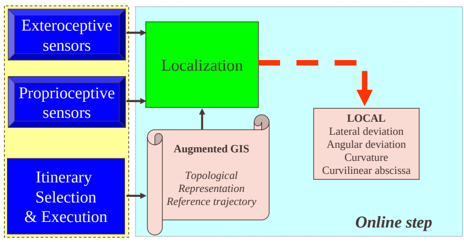
Navigation

Memory based navigation (relative navigation)



Navigation

Memory based navigation (relative navigation)



SLAN vs SLAM

	SLAM	SLAN
MAPS	Metric 2D/3D	Topological Semi-Metric
Localization	Absolute	Relative
Control space	Cartesian	Sensor space Cartesian (relative)

Outline of the presentation

Introduction
- LASMEA
- Sensor based Control
- Navigation
- SLAN vs SLAM

A generic model
- A unified model for camera
- Case of fisheye cameras
- Partial euclidian reconstruction

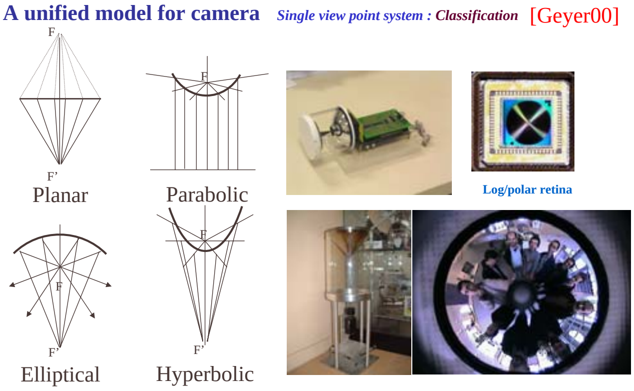
Togological navigation system
- Global system
- Case of vision
- Differents Modules
- SOVIN

Illustrations
- Indoor applications
- Outdoor applications

Discussion
- Learning process
- Updating process
- Closing the loop problem

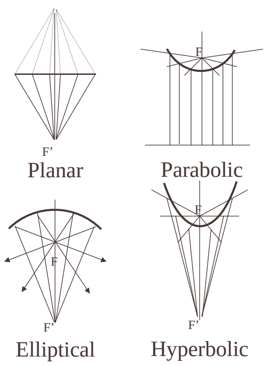
A unified model for camera

Single view point system : Classification [Geyer00]



A unified model for camera

Single view point system : Classification [Geyer00]



φ and ξ : Mirror parameters

Mirror type	ξ	φ
Parabolic	1	1+2p
Hyperbolic	$\frac{d}{\sqrt{d^2+4p^2}}$	$\frac{d+2p}{\sqrt{d^2+4p^2}}$
Elliptical	$\frac{d}{\sqrt{d^2+4p^2}}$	$\frac{d-2p}{\sqrt{d^2+4p^2}}$
Planar	0	1
Conventional	0	1

A unified model for camera

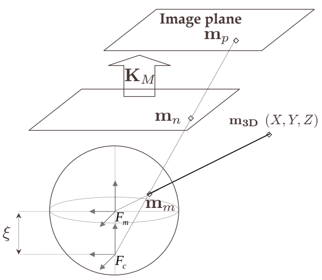
Single view point system : case of point

$$\mathbf{m_m} = \frac{1}{\rho}\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

$$\rho = \|\mathbf{m}\| = \sqrt{X^2+Y^2+Z^2}$$

$$\mathbf{m_n} = [\mathbf{x}^T \beta]^T = \begin{bmatrix} \frac{X}{Z+\xi\rho} \\ \frac{Y}{Z+\xi\rho} \\ \beta \end{bmatrix}$$

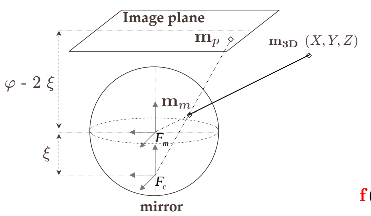
$$\mathbf{m_p} = \mathbf{K_M m_n}$$



Mirror : unitary sphere

A unified model for camera

Single view point system : case of point



φ and ξ : Mirror parameters

$$\mathbf{M} = \begin{bmatrix} \varphi-\xi & 0 & 0 \\ 0 & \varphi-\xi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{K} = \begin{bmatrix} f & fs & u_0 \\ 0 & fr & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathrm{f}(\mathbf{m_{3D}}) = \begin{bmatrix} \frac{X}{\epsilon Z+\xi\sqrt{X^2+Y^2+Z^2}} \\ \frac{Y}{\epsilon Z+\xi\sqrt{X^2+Y^2+Z^2}} \\ 1 \end{bmatrix}$$

Special cases :

$\epsilon = 1$ and $\xi = 0$ ➔ perspective projection

$\epsilon = 0$ and $\xi = 1$ ➔ spherical projection

Generic Projection function

$$\mathbf{m}_p = \underbrace{\mathbf{K}\ \mathbf{M}}_{\mathbf{K}_M}\ \mathrm{f}(\mathbf{m_{3D}})$$

Case of fisheye camera

Fisheye camera model : definition

★ $\mathcal{F}_m$: camera frame
★ $\mathcal{F}_i$: image frame

★ r = distance between X and the image point
★ θ = angle between the incoming ray and the principal axis

$\mathsf{X} = (X\ Y\ Z)^T$

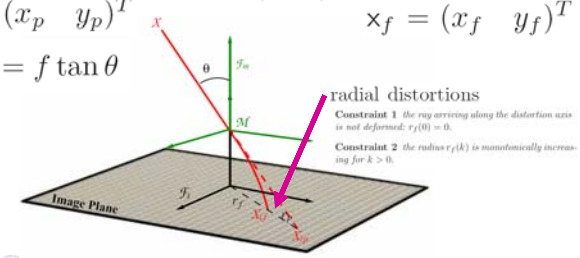
Pinhole camera

$$\mathsf{x}_p = (x_p \quad y_p)^T$$

$$r_p(\theta) = f\tan\theta$$

Fisheye camera

$$\mathsf{x}_f = (x_f \quad y_f)^T$$



Case of fisheye camera

Fisheye camera model :Pinhole based models

Mapping :

$$r_p \xrightarrow[T_1]{} r_f$$

$L(r_p, n) = 1+k_1 r_p^2+k_2 r_p^4+\ldots+k_n r_p^{2n}$ n parameters

★ $r_f^1(r_p) = r_p L(r_p, n)$ used for perspective cameras
[T. Pajdla97,Hartley00,Zhang98,Ma2006…]

★ $r_f^2(r_p) = \frac{r_p}{L(r_p,n)}$ handle high distortion
[Fitzgibbon01]

★ $r_f^3(r_p) = \frac{r_p}{1+k_1 r_p^2}$ one parameter
[Fitzgibbon01]

Case of fisheye camera

Fisheye camera model : Pinhole based models

Mapping : $r_p \xrightarrow[T_1]{} r_f$

★ $r_f^4(r_p) = r_p \frac{L_1(r_p,n_1)}{L_2(r_p,n_2)}$ rational model n1+n2 parameters [Li05]

★ $r_f^5(r_p) = s\log(1+\lambda r_p)$ logarithmic mapping [Basu95]

s is a scaling factor and λ a gain to control the amount of distortion

★ $r_f^6(r_p) = \frac{1}{\omega}\arctan\left(2r_p\tan\frac{\omega}{2}\right)$ Field-of-view (FOV) model [Devernay01]

★ $r_f(r_p) = \lambda_{r_p}$ [Hartley07]

The "distortion function" $\lambda = \{\lambda_1, \lambda_2, \dots \lambda_n\}$ can be then fitted by a parametric model

Case of fisheye camera

Fisheye camera model : Captured rays based models

Mapping : $\theta \xrightarrow[T_2]{} r_f$

★ $r_f^1(\theta) = f\theta$ cameras with limited distortions [Kingslake89]

★ $r_f^2(\theta) = 2f\tan\left(\frac{\theta}{2}\right)$ stereographic projection preserves circularity and thus project 3D local symmetries onto 2D local symmetries [Fleck94,Stevenson95]

★ $r_f^3(\theta) = f\sin\theta$ orthogonal or sine law projection [Ray94]

★ $r_f^4(\theta) = f\sin\left(\frac{\theta}{2}\right)$ equisolid angle projection [Smith92]

Case of fisheye camera

Fisheye camera model : Captured rays based models

Mapping : $\theta \xrightarrow[T_2]{} r_f$

★ $r_f^5(\theta) = f(k_1\theta + k_2\theta^3 + \cdots + k_n\theta^{2(n-1)+1})$

Improvment of the polynomial model accuracy

[xiong97,kannala04,schwalbe05,Scaramuzza06]

★ $r_f^6(\theta) = \alpha\sin(\beta\theta)$

★ α : scale factor

★ β : radial mapping parameter

[Kumler00]

★ $r_f^7(\theta) = a\tan(\theta/b) + c\sin(\theta/d)$

Combination of
- stereographic projection (with parameters a, b)
- equisolid angle projection (with parameters c, d)

[Bakstein02]

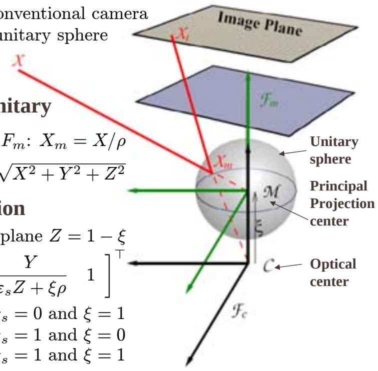
Case of fisheye camera

Generic camera model : Unified Spherical Model

★ F_c : frame attached to the conventional camera

★ F_m : frame attached to the unitary sphere

$X = [X\ Y\ Z]^T$ $\underline{x} = [x\ y\ 1]^T$



Step 1 : Projection on the unitary sphere

X_m in F_m: $X_m = X/\rho$

where $\rho = \|X\| = \sqrt{X^2+Y^2+Z^2}$

Step 2 : Perspective projection

on the normalized image plane $Z = 1-\xi$

$$\underline{x}\text{'}=f(X) = \begin{bmatrix} \frac{X}{\varepsilon_s Z + \xi\rho} & \frac{Y}{\varepsilon_s Z + \xi\rho} & 1 \end{bmatrix}^T$$

spherical projection : $\varepsilon_s = 0$ and $\xi = 1$

pinhole model : $\varepsilon_s = 1$ and $\xi = 0$

Catadioptric model : $\varepsilon_s = 1$ and $\xi = 1$

Case of fisheye camera

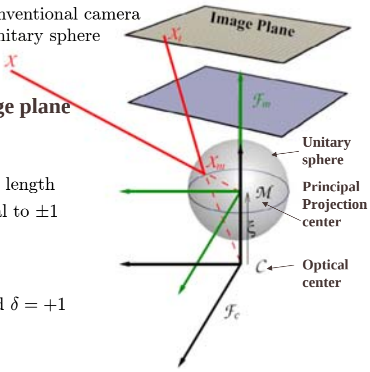
Generic camera model : Unified Spherical Model

★ F_c : frame attached to the conventional camera

★ F_m : frame attached to the unitary sphere

$X = [X\ Y\ Z]^T$ $\underline{x} = [x\ y\ 1]^T$

Step 3 : Projection onto image plane



$\underline{x} = M\,\underline{x}\text{'}$

$$M = \begin{pmatrix} f\kappa & 0 & 0 \\ 0 & \delta f\kappa & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

f : focal length

δ is equal to ± 1

$\kappa > 0$

catadioptric cameras : $\delta = -1$

perspective camera : $\kappa = 1$ and $\delta = +1$

Case of fisheye camera

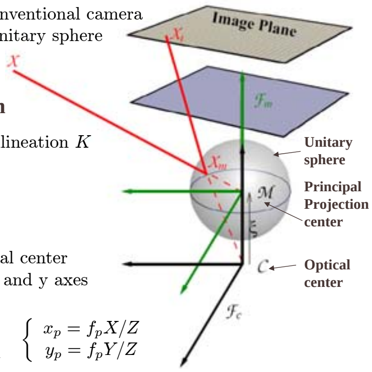
Generic camera model : Unified Spherical Model

★ F_c : frame attached to the conventional camera

★ F_m : frame attached to the unitary sphere

$X = [X\ Y\ Z]^T$ $\underline{x} = [x\ y\ 1]^T$

Step 4 : Pixel transformation



$m = K\,\underline{x}$ plane-to-plane collineation K

$$K = \begin{pmatrix} k_u & s_{uv} & u_0 \\ 0 & k_v & v_0 \\ 0 & 0 & 1 \end{pmatrix}$$

$(u_0, v_0)^T$: position of the optical center

k_u and k_v : scaling along the x and y axes

s_{uv} : skew

Pinhole camera setting $\xi = 0$, $\kappa = 1$ and $\delta = +1$

$$\begin{cases} x_p = f_p X/Z \\ y_p = f_p Y/Z \end{cases}$$

Case of fisheye camera

Generic camera model : Unified Spherical Model

For fisheye camera

$$r_f = r_f(r_p) = \frac{\frac{f_f}{f_p} r_p}{1+\xi\sqrt{\frac{r_p^2}{f_p^2}+1}}$$

is a T_1-mapping linking the perspective Radius r_p and the fisheye radius r_f

Constraint 1 : easily verified ($r_f(0)=0$)
Constraint 2 : $r_f(k)$ is monotonically increasing for $k>0$

$$r_f = r_f(\theta) = \frac{f_f \tan\theta}{1+\xi\sqrt{\tan^2\theta+1}}$$

is a T_2-mapping linking the radius r_f and the incidence angle θ

The generic camera model is candidate for fisheye camera modeling

Case of fisheye camera

Validation with calibration toolbox [MEI07]

Computation of f_f and ξ (Producers data)
Model fitting for all existing fisheye cameras
Calibration

List of fisheye cameras considered

Coastal Optical 4.88mm f/5.2
Nikon MF 6mm f/2.8
Coastal Optical 7.45mm f/2.8
Sigma 8mm f/4.0 < 1998
Peleng 8mm f/3.5
Nikkor 10.5mm f/2.8
ORIFL 190-3

List of models considered

Polynomial model using perspective projection
$$r_f(r_p) = r_p(1 + a_1 r_p^2 + \cdots + a_3 r_p^6)$$

Polynomial model using unified model projection
$$r_f(r_u) = r_u(1 + a_1 r_u^2 + \cdots + a_3 r_u^6)$$

Proposed unified model
$$r_f = r_u$$

The generic camera model is validated with an average of 0.2 pixels reprojection error for fisheye camera modeling

Only one parameter required

Partial euclidian reconstruction

Unified model - Case of points

Introducing $\eta = s\frac{\rho}{|Z|} = \sqrt{1+X^2/Z^2+Y^2/Z^2} \quad > 0$ s=sign(Z)

$$\begin{cases} x = \frac{X/Z}{1+\xi\eta} \\ y = \frac{Y/Z}{1+\xi\eta} \end{cases} \quad \begin{cases} (1+\xi\eta)\ x = \frac{X}{Z} \\ (1+\xi\eta)\ y = \frac{Y}{Z} \end{cases} \quad \left\{ \eta^2 - (x^2+y^2)(1+\xi\eta)^2 - 1 = 0 \right.$$

$$\eta = \frac{\pm\gamma_x + \xi(x^2+y^2)}{1-\xi^2(x^2+y^2)}$$

η computed without ambiguity

where $\gamma_x = \sqrt{1+(1-\xi^2)(x^2+y^2)}$

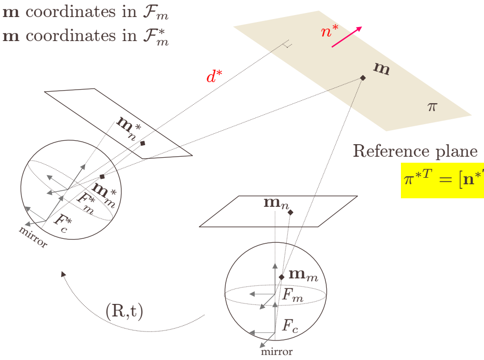
$$\mathbf{m_m} = (\eta^{-1}+\xi)\overline{\mathbf{m}} \quad \text{with} \quad \overline{\mathbf{m}} = [x \ y \ \tfrac{1}{1+\xi\eta}]^T$$

$\mathbf{m}_n$

Partial euclidian reconstruction

Unified model - Case of points

$[X\ Y\ Z]^T$ $\mathbf{m}$ coordinates in $\mathcal{F}_m$
$[X^*\ Y^*\ Z^*]^T$ $\mathbf{m}$ coordinates in $\mathcal{F}_m^*$



Reference plane π in F_m^*

$$\pi^{*T} = [\mathbf{n}^{*\mathbf{T}} \quad -d^*]$$

Partial euclidian reconstruction

Unified model - Case of points

$$\mathrm{m}_m = (\eta^{-1}+\xi)\overline{\mathbf{m}} = \tfrac{1}{\rho}\left[\ X \quad Y \quad Z\ \right]^T$$
$$\mathbf{m}_\mathbf{m}^* = (\eta^{*-1}+\xi)\overline{\mathbf{m}}^* = \tfrac{1}{\rho^*}\left[\ X^* \quad Y^* \quad Z^*\ \right]^T$$

$$\overline{\mathbf{m}} = [x\ y\ \tfrac{1}{1+\xi\eta}]^T$$
$$\overline{\mathbf{m}}^* = [x^*\ y^*\ \tfrac{1}{1+\xi\eta^*}]^T$$

Homogeneous coordinates:

$\underline{\mathbf{m}} = [X\ Y\ Z\ H]^T$ and $\underline{\mathbf{m}}^* = [X^*\ Y^*\ Z^*\ H^*]^T$

$$\rho(\eta^{-1}+\xi)\overline{\mathbf{m}} = \left[\ \mathbf{I_3} \quad 0\ \right]\underline{\mathbf{m}} = \left[\ \mathbf{R} \quad \mathbf{t}\ \right]\underline{\mathbf{m}}^*$$

The distance from the world point m to the plane (π)

$$d(m,\pi) = \pi^{*T}\cdot\underline{\mathbf{m}}^* = [\mathbf{n}^{*\mathbf{T}} \quad -d^*][X^*\ Y^*\ Z^*\ H^*]^T = \rho^*(\eta^{*-1}+\xi)\mathbf{n}^{*T}\overline{\mathbf{m}}^* - d^*H^*$$

Partial euclidian reconstruction

Unified model - Case of points

$$H^* = \frac{\rho^*(\eta^{*-1}+\xi)}{d^*}\mathbf{n}^{*T}\overline{\mathbf{m}}^* - \frac{d(m,\pi)}{d^*}$$
$$\rho^*(\eta^{*-1}+\xi)\overline{\mathbf{m}}^* = \left[\ X^* \quad Y^* \quad Z^*\ \right]^T$$

$$\underline{\mathbf{m}}^* = \rho^*(\eta^{*-1}+\xi)\mathbf{A}_\pi^*\overline{\mathbf{m}}^* + \mathbf{b}_\pi^*$$

where

$$\begin{cases} \mathbf{A}_\pi^* = \left[\mathbf{I_3} \quad \frac{\mathbf{n}^*}{d^*}\right]^T \\ \mathbf{b}_\pi^* = \left[\mathbf{0}_{1\times 3} \quad -\frac{d(m,\pi)}{d^*}\right] \end{cases}$$

$$\rho(\eta^{-1}+\xi)\overline{\mathbf{m}} = \left[\ \mathbf{R} \quad \mathbf{t}\ \right]\underline{\mathbf{m}}^*$$

$$\rho(\eta^{-1}+\xi)\overline{\mathbf{m}} = \rho^*(\eta^{*-1}+\xi)\mathbf{H}_\pi\overline{\mathbf{m}}^* + \alpha\mathbf{t}$$

with $\mathbf{H}_\pi = \mathbf{R} + \frac{\mathbf{t}}{d^*}\mathbf{n}^{*T}$ and $\alpha = -\frac{d(m,\pi)}{d^*}$

Euclidian Homography

Partial euclidian reconstruction
Unified model - Case of points

$m \in \pi$

$$\overline{\mathbf{m}} = \beta_{x,x^*} \mathbf{H}_\pi \overline{\mathbf{m}}^*$$

where $\beta_{x,x^*} = \frac{\eta^{*-1}+\xi}{\eta^{-1}+\xi}\frac{\rho^*}{\rho}$

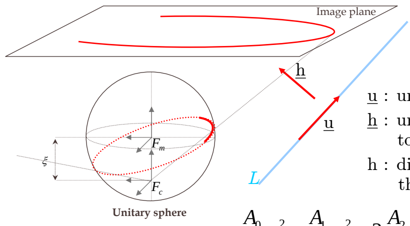
Linear form

$\overline{\mathbf{m}} \times \mathbf{H}_\pi \overline{\mathbf{m}}^* = \mathbf{0}$ → $\mathbf{H}_\pi$ → $\mathbf{R}$ and $\mathbf{t}_{d^*} = \frac{\mathbf{t}}{d^*}$

$$\sigma = \frac{\rho}{\rho^*} = (1 + \mathbf{n}^{*T}\mathbf{R}^T\mathbf{t}_{d^*})\frac{(\eta^{*-1}+\xi)n^{*T}\overline{\mathbf{m}}^*}{(\eta^{-1}+\xi)n^{*T}\mathbf{R}^T\overline{\mathbf{m}}}$$

Partial euclidian reconstruction
Unified model - Case of lines

Similar results are obtained from lines *(using polar lines)*



Plücker coordinates

$$L = (\underline{u}, \underline{h}, h)^T$$

- $\underline{u}$: unitary vector along the line $\mathbf{L}$
- $\underline{h}$: unitary vector orthogonal to the interpretation plane
- h : distance between the origin of the frame and the line

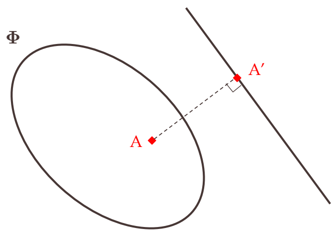
$$\frac{A_0}{A_5}x^2 + \frac{A_1}{A_5}y^2 + 2\frac{A_2}{A_5}xy + 2\frac{A_3}{A_5}x + 2\frac{A_4}{A_5}y + 1 = 0$$

$$B_i(\underline{h}, \xi, K_M) = \frac{A_i}{A_5}$$

Partial euclidian reconstruction
Unified model - Case of lines

Similar results are obtained from lines *(using polar lines)*

Polar lines



- ϕ a 2D conic curve
- A a point in the definition plane of ϕ
- $\mathbf{l_i}$: the corresponding polar line

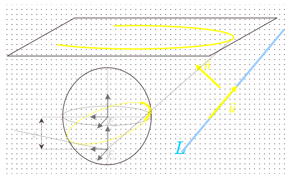
$$\mathbf{l_i} \propto \mathbf{\Phi A}$$

Partial euclidian reconstruction
Unified model - Case of lines

Similar results are obtained from lines *(using polar lines)*

Polar lines

$$\Omega = \begin{pmatrix} h_x^2 - \xi^2(1-h_y^2) & h_x h_y(1-\xi^2) & h_x h_z \\ h_x h_y(1-\xi^2) & h_y^2 - \xi^2(1-h_x^2) & h_y h_z \\ h_x h_z & h_y h_z & h_z^2 \end{pmatrix}$$



$$\underline{\mathbf{x_i}}^T \underbrace{\mathbf{K}^{-T}\mathbf{\Omega}\mathbf{K}^{-1}}_{\mathbf{\Omega_i}} \underline{\mathbf{x_i}} = 0$$

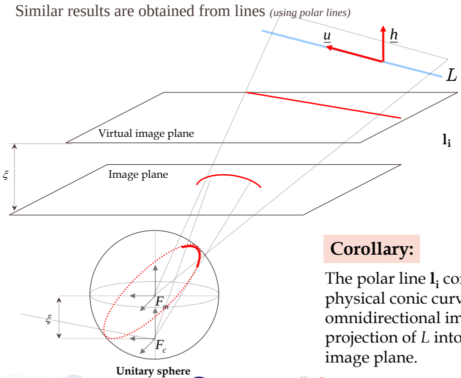
The polar line of the optical center with respect to the conic $\mathbf{\Omega_i}$ is given by:

$$\mathbf{l_i} \propto \mathbf{\Omega_i O_i}$$

principal point
$\mathbf{O_i} = [u_0 \; v_0 \; 1]^T = \mathbf{K}[0 \; 0 \; 1]^T$

Partial euclidian reconstruction
Unified model - Case of lines

Similar results are obtained from lines *(using polar lines)*



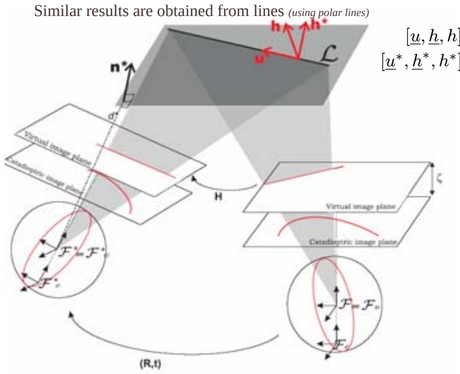
$$\mathbf{l_i} \propto \mathbf{K}^{-T}\mathbf{\Omega}\mathbf{K}^{-1}\mathbf{O_i} \propto \mathbf{K}^{-T}\mathbf{\Omega}\mathbf{K}^{-1}\mathbf{K}\begin{pmatrix}0\\0\\1\end{pmatrix} \propto \mathbf{K}^{-T}\begin{pmatrix}h_x\\h_y\\h_z\end{pmatrix}$$

Corollary:

The polar line $\mathbf{l_i}$ computed from the physical conic curve projection of L in the omnidirectional image is the perspective projection of L into the virtual camera image plane.

Partial euclidian reconstruction
Unified model - Case of lines

Similar results are obtained from lines *(using polar lines)*



$[\underline{u}, \underline{h}, h]$ $\mathbf{L}$ coordinates in $\mathcal{F}_m$

$[\underline{u}^*, \underline{h}^*, h^*]$ $\mathbf{L}$ coordinates in $\mathcal{F}_m^*$

Reference plane π in F_m^*

$$\pi^{*T} = [\mathbf{n}^{*\mathbf{T}} \; -d^*]$$

Partial euclidian reconstruction

Unified model - Case of lines

Similar results are obtained from lines *(using polar lines)*

$\mathcal{X}_1$ and $\mathcal{X}_2$ are two points in the 3D space lying on the line $\mathcal{L}$

$$\underline{\mathbf{h}} \propto \mathbf{H}^{-\top}\underline{\mathbf{h}}^*$$

$$\mathbf{l_i} \propto \mathbf{K}^{-T}\underline{\mathbf{h}} \rightarrow \mathbf{l_i} \propto \mathbf{K}^{-T}\mathbf{H}^{-T}\underline{\mathbf{h}}^*$$

$$\mathbf{l_i}^* \propto \mathbf{K}^{-T}\underline{\mathbf{h}}^* \rightarrow \mathbf{l_i} \propto \mathbf{K}^{-T}\mathbf{H}^{-T}\mathbf{K}^{T}\mathbf{l_i}^*$$

$$\mathbf{l_i} \propto \mathbf{G}^{-\top}\mathbf{l_i}^*$$

$$\mathbf{G} = \mathbf{KHK}^{-1}$$

Linear form

$$\mathbf{l_i} \times \mathbf{G}^{-\top}\mathbf{l_i}^* = \mathbf{0}$$

with 4 couples (l_i, l_i^*) $\rightarrow \mathbf{H} \rightarrow$ $\mathbf{R}$ and $\mathbf{t}_{d^*} = \frac{\mathbf{t}}{d^*}$

$$\mathbf{H} = \mathbf{R} + \frac{\mathbf{t}}{d^*}\mathbf{n}^{*T}$$

$$r = \frac{h}{h^*} = (1 + \mathbf{t}_d^{*\top}\mathbf{R}^\top\mathbf{n}^*)\frac{\|\mathbf{n}^*\times\mathbf{K}^\top\mathbf{l_i}^*\|}{\|\mathbf{Rn}^*\times\mathbf{K}^\top\mathbf{l_i}\|}$$

Outline of the presentation

Introduction
- LASMEA
- Sensor based Control
- Navigation
- SLAN vs SLAM

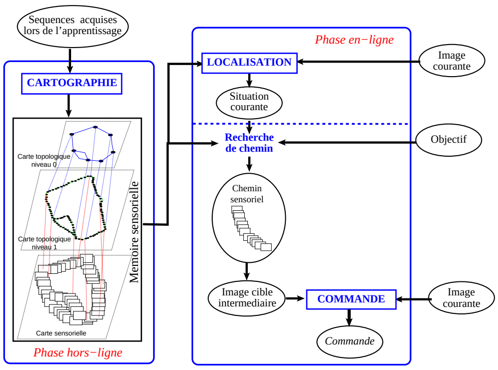
A generic model
- A unified model for camera
- Case of fisheye cameras
- Partial euclidian reconstruction

Togological navigation system
- Global system
- Case of vision
- Differents Modules
- SOVIN

Illustrations
- Indoor applications
- Outdoor applications

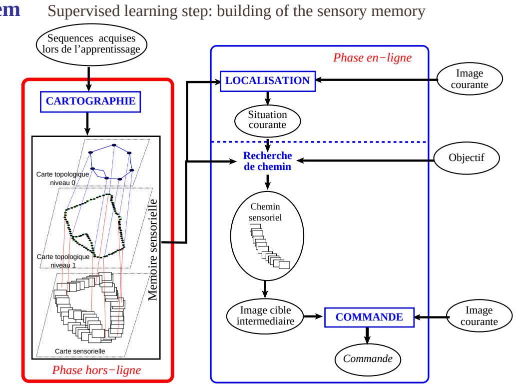
Discussion
- Learning process
- Updating process
- Closing the loop problem

Global system



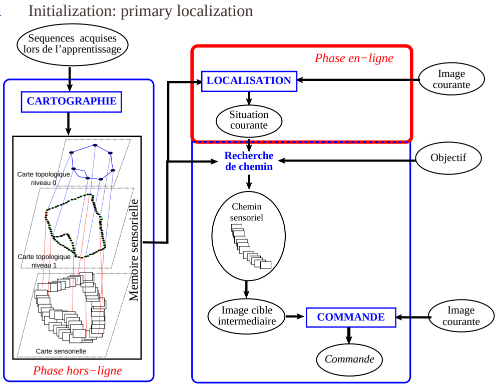
Global system

Supervised learning step: building of the sensory memory



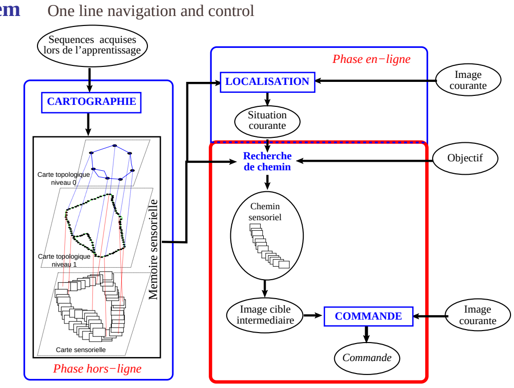
Global system

Initialization: primary localization

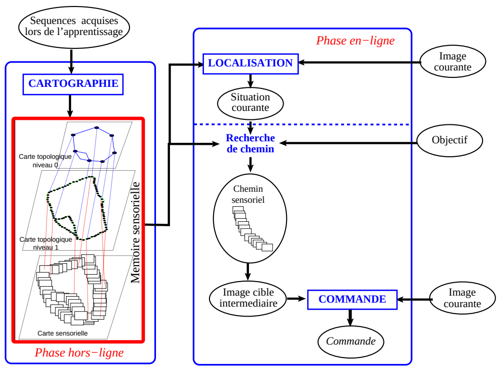


Global system

One line navigation and control



Case of vision



Case of vision

Topological map level 0 (CT0)

- ✓High level representation : corridor, routes, ..

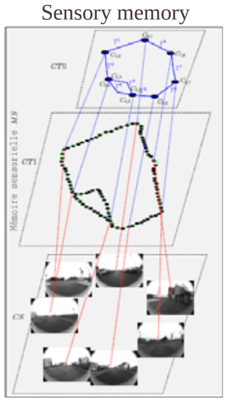
Topological map level 1 (CT1)

- ✓Image network representation
- ✓Navigability
- ✓Commandability
- ✓Plannification

Sensory map (CS)

- ✓Initial localization
- ✓Sensor based control
- ✓Vizualization of the environment
- ✓Human interface

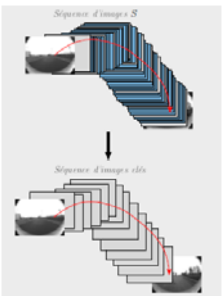
Sensory memory



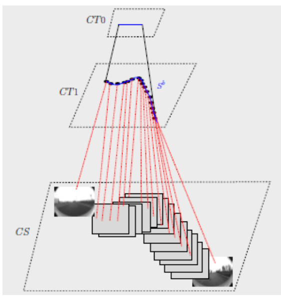
Different modules

Learning step (first arc, first two nodes)

Key images selection

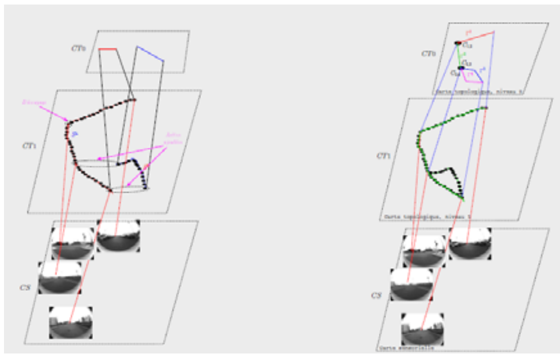


Building the sensory memory



Different modules

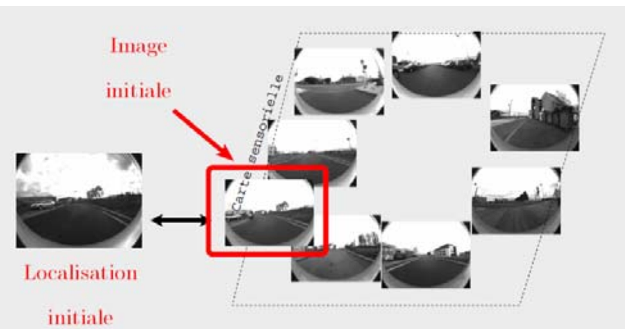
Learning step (expanding the processus…)



Differents modules

Initial localization

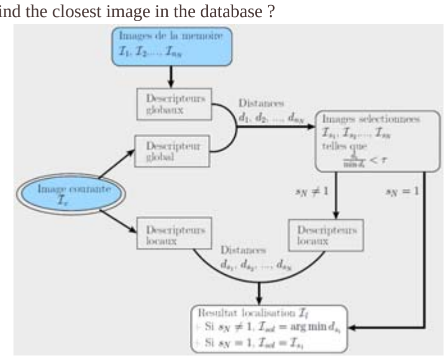
How to find the closest image in the database ?



Differents modules

Initial localization

How to find the closest image in the database ?

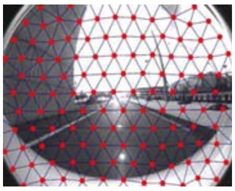


Differents modules Initial localization

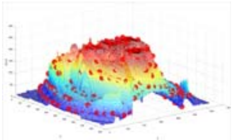
How to find the closest image in the database ?

Use of a regular mesh [PERSSON04]
Cubic interpolation of the image surface

Global descriptor : vector of the interpolated grey level pixels at the position of the control point



Control point distribution (one example)



Differents modules Initial localization

How to find the closest image in the database ?

Local descriptor : interest points detected by Harris Stephen detector

Use of patches 11x11

Matching done by using the centered cross correlation score ZNCC

Use of the number of matched point to compute the similarity between 2 images

Differents modules Initial localization

How to find the closest image in the database ?

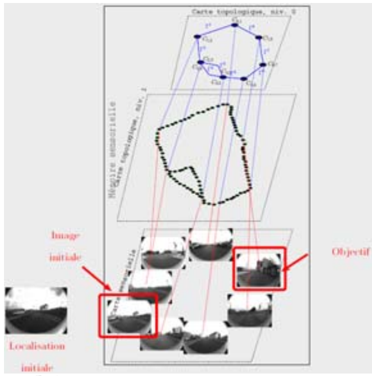
Evaluation of the approach [ICRA08] using three image databases regarding 3 main criteria: Used memory size, percentage of success, computation time for localization

Caméra	Taille image	Nb images	Images
Catadioptrique	1024 × 768	978	
Fisheye 110 deg	384 × 288	188	
Fisheye 190 deg	640 × 480	445	

Several approaches were compared : GONZ [GONZALES02], PHLAC [LINAKER04], CUB, SURF, SIFT, HARRIS, CUBHAR

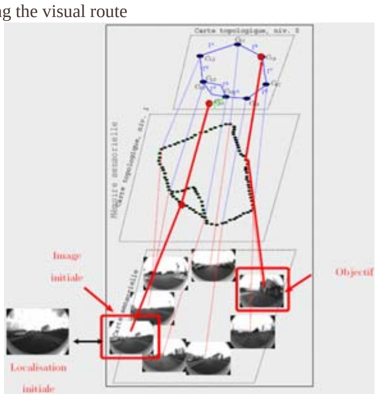
Differents modules Planifying the visual route

How to find the optimal path ?



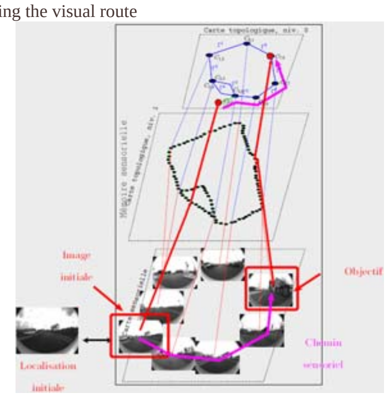
Differents modules Planifying the visual route

How to find the optimal path ?



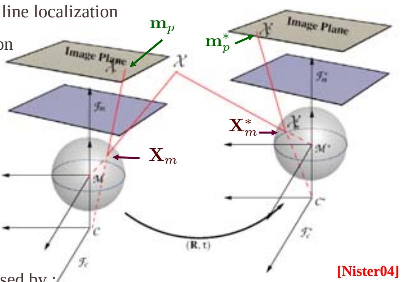
Differents modules Planifying the visual route

How to find the optimal path ?



Differents modules One line localization

Scaled euclidian reconstruction



Epipolar constraint can be expressed by :

$$X_m{}^T \mathbf{R}(t \times X_m{}^{*T}) = X_m{}^T \mathbf{R}\,[t]_\times X_m{}^{*T} = 0$$

For pinhole model $X_m{}^T \mathbf{E} X_m{}^{*T} = 0$

[Nister04]

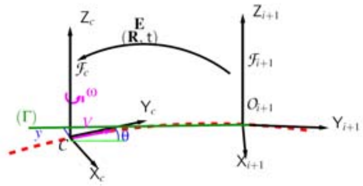
From five couples of points E can be estimated

Outliers are rejected by using RANSAC

Differents modules State estimation and control (one example)

State model of the mobile robot

$$(s, y, \theta)$$



★ $\mathcal{F}_i = (O_i, \mathbf{X_i}, \mathbf{Y_i}, \mathbf{Z_i})$

★ $\mathcal{F}_{i+1} = (O_{i+1}, \mathbf{X_{i+1}}, \mathbf{Y_{i+1}}, \mathbf{Z_{i+1}})$ the frames attached to the robot when $\mathcal{I}_i$

Kinematic model of the mobile robot

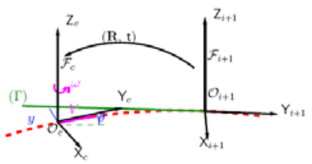
$$\begin{cases} \dot{s} = v\dfrac{\cos\theta}{1-c(s)y} \\ \dot{y} = v\sin\theta \\ \dot{\theta} = v\left(\dfrac{\tan\delta}{l} - \dfrac{c(s)\cos\theta}{1-c(s)y}\right) \end{cases}$$

Chained System theory

[IAV04]

Differents modules State estimation and control (one example)

Path following approach

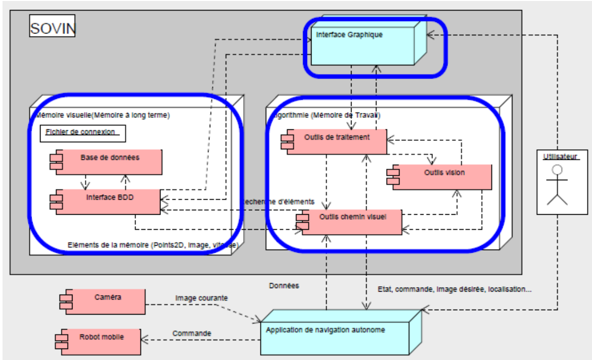


Goal: $y \longrightarrow 0$, $\theta \longrightarrow 0$ before $\mathcal{F}_c$ reaches $\mathcal{F}_{i+1}$

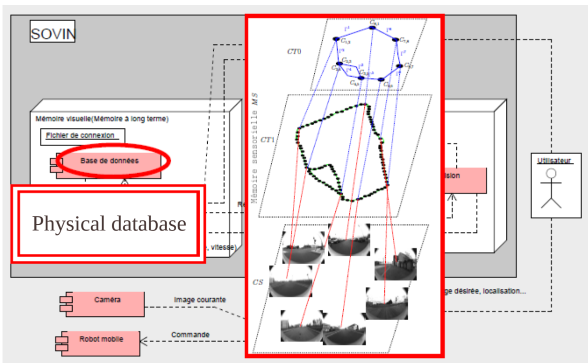
Asymptotically stable guidance control law based on the chain system approach:

$$\delta(y,\theta) = \arctan\left(-l\left[\cos^3\theta\,(-K_d\tan\theta - K_p y)\right]\right)$$

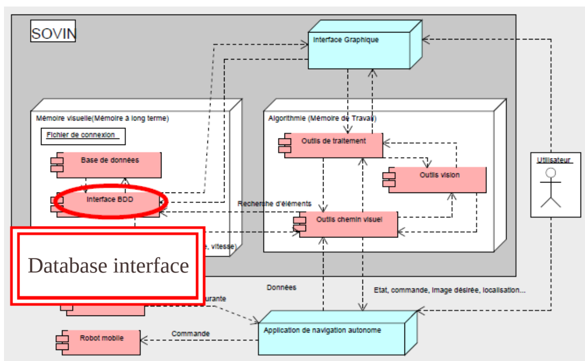
SOVIN A global software environment for topological navigation in wide spaces



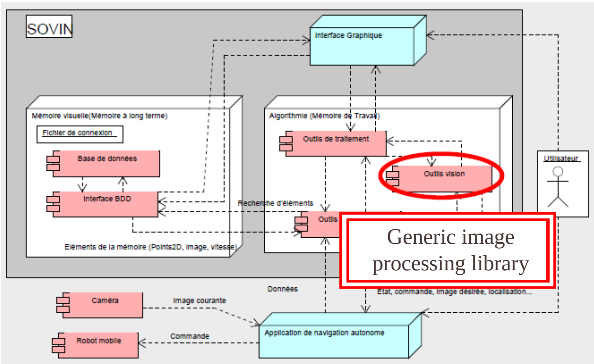
SOVIN A physical database organized in three levels



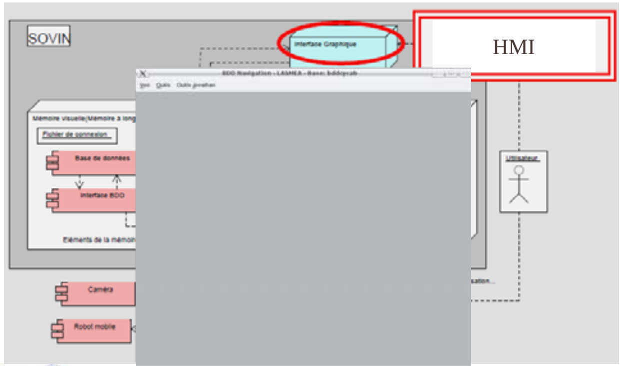
SOVIN An efficient database interface for on line and real time applications



SOVIN A generic library (generic visual sensor) for visual algorithm processing



SOVIN An efficient Human Machine Interface for managing applications



Outline of the presentation

Introduction
- *LASMEA*
- *Sensor based Control*
- *Navigation*
- *SLAN vs SLAM*

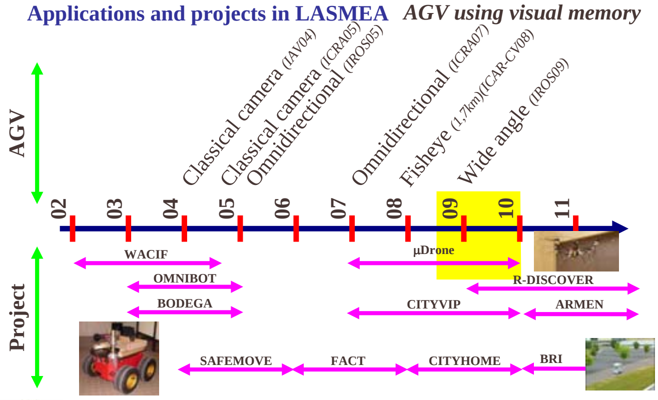
A generic model
- *A unified model for camera*
- *Case of fisheye cameras*
- *Partial euclidian reconstruction*

Togological navigation system
- *Global system*
- *Case of vision*
- *Differents Modules*
- *SOVIN*

Illustrations
- *Indoor applications*
- *Outdoor applications*

Discussion
- *Learning process*
- *Updating process*
- *Closing the loop problem*

Applications and projects in LASMEA *AGV using visual memory*



Indoor applications : WACIF project (2002-2005)

Autonomous Robot for Telepresence navigation, localization, learning, warning and communication capabilities

- **Study and development of a demonstrator** like a personal robot fully integrated in the context of « wireless home »
- **An innovant domotic HMI** dedicated for software useful for human, family and home applications

Wireless communication and services

- **Telepresence and telesurveillance tasks**
home supervision through any wireless point, abnormal event detection in family environment (intrusion, noise, …)

Autonomous navigation strategy

Indoor applications : WACIF project (2002-2005)

Global navigation strategy

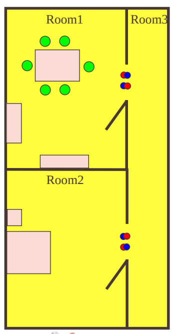
Topological representation
Predefined trajectory
Graph representation
Visual memory concept

Hypothesis : knowledge

$\text{HS} = \text{HOME Space}$

$\text{RS}_i = \text{Room Space } i$

$\text{HS} = \sum_i RS_i$



- **Starting point**
- **Ending point**

Topological representation step is a part of the knowledge we get from the HOME environment

Indoor applications : WACIF project (2002-2005)

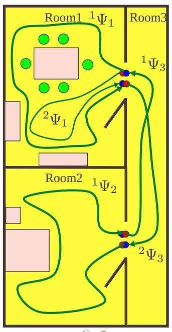
Global navigation strategy

Topological representation

Predefined trajectory

Graph representation

Visual memory concept



Predefined trajectory step
is totally supervised
using robot teleoperation

Indoor applications : WACIF project (2002-2005)

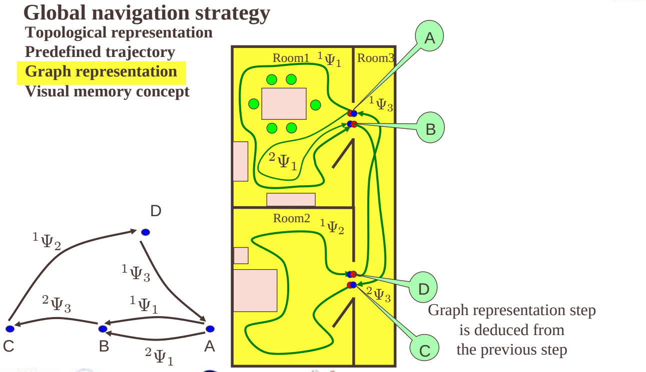
Global navigation strategy

Topological representation

Predefined trajectory

Graph representation

Visual memory concept



Graph representation step
is deduced from
the previous step

Indoor applications : WACIF project (2002-2005)

Global navigation strategy

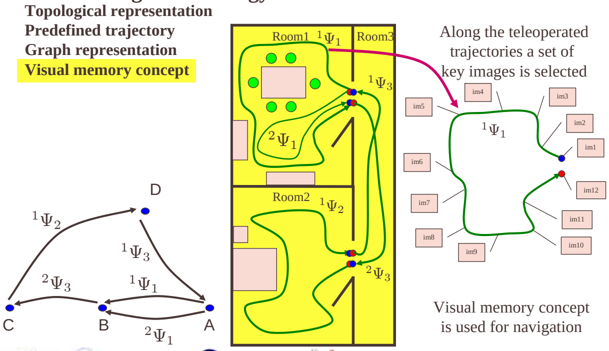
Topological representation

Predefined trajectory

Graph representation

Visual memory concept

Along the teleoperated
trajectories a set of
key images is selected



Visual memory concept
is used for navigation

Indoor applications : WACIF project (2002-2005)

Global navigation strategy

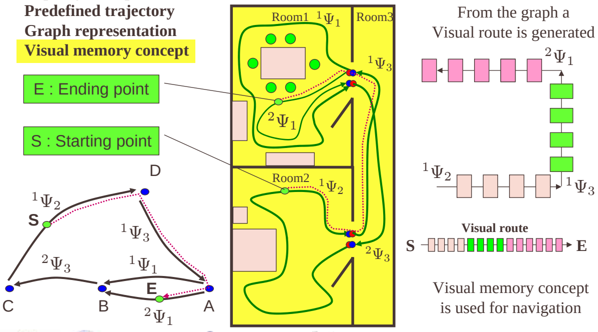
Topological representation

Predefined trajectory

Graph representation

Visual memory concept

From the graph a
Visual route is generated

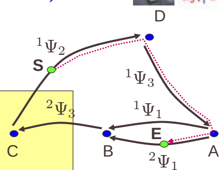


Visual memory concept
is used for navigation

Indoor applications : WACIF project (2002-2005)

Global navigation strategy

Functional decomposition



Learning Off line

Define each ${}^{i}\Psi_j$: learn the trajectory

- Robot teleoperation
- Models features extraction and selection
- Mosaic construction

Re-run of trajectory : On line

- Extract the reference model from the visual route
- Select and track models in current images
- Localization in the mosaic
- Lateral control of the non holonomous robot

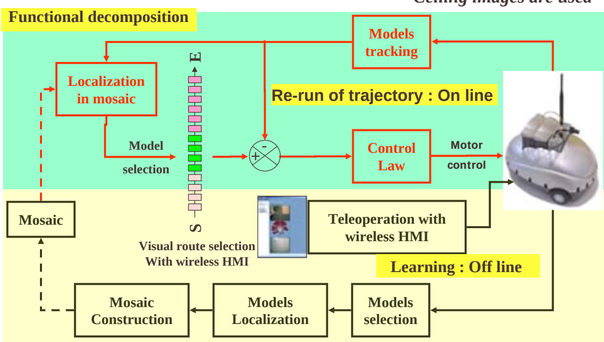
Ait Ader [04]
Blanc [04]
Blanc [05]

Indoor applications : WACIF project (2002-2005)

Global navigation strategy

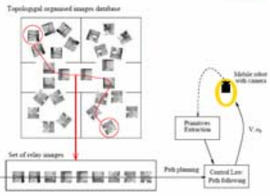
Ceiling images are used

Functional decomposition



Indoor applications : WACIF project (2002-2005)

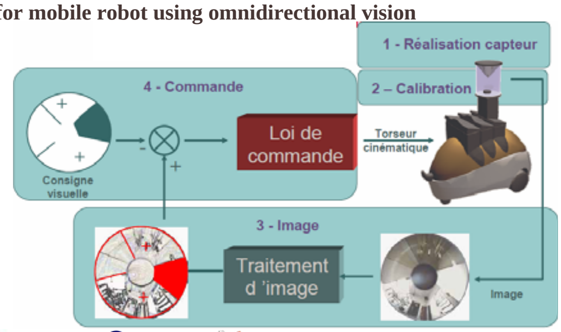
Navigation using visual memory

Indoor applications : OMNIBOT project

From omnidirectional vision to mobile robot control

Development of a complete perception and control approach for mobile robot using omnidirectional vision

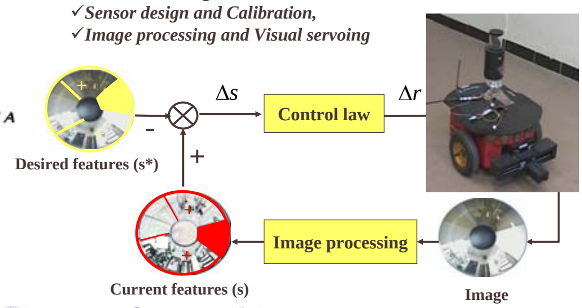


Indoor applications : OMNIBOT project

From omnidirectional vision to mobile robot control

Development of a complete perception and control approach for mobile robot using omnidirectional vision :

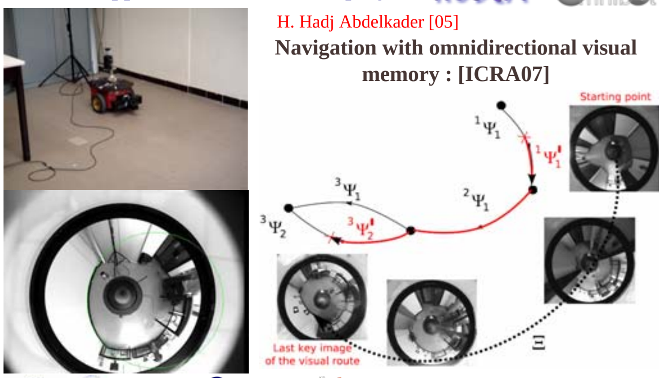
- ✓*Sensor design and Calibration,*
- ✓*Image processing and Visual servoing*



Indoor applications : OMNIBOT project

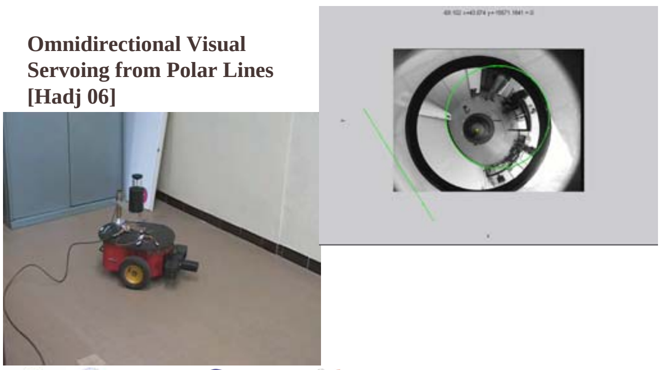
H. Hadj Abdelkader [05]

Navigation with omnidirectional visual memory : [ICRA07]



Indoor applications : OMNIBOT project

Omnidirectional Visual Servoing from Polar Lines [Hadj 06]



Indoor applications : ROSACE project

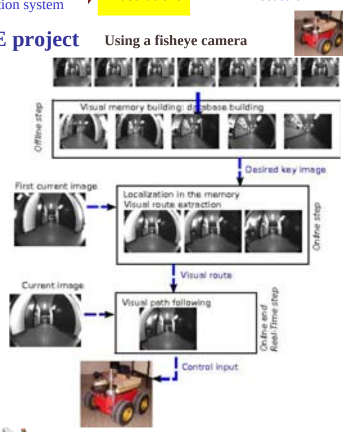
Using a fisheye camera

General framework :

Vision based memory navigation strategy :

3 steps :

1. *Visual memory building*
2. *Localization into the visual memory*
3. *Navigation into the visual memory*



Indoor applications : ROSACE project

Using a fisheye camera

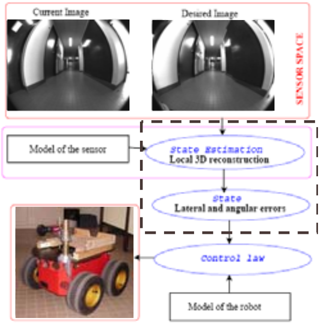
Localization

Vision based memory navigation strategy :

Localization into the visual memory

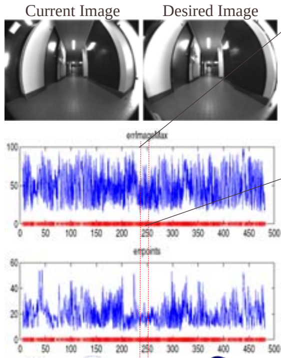
a- Global localization
b- Selection of key images
c- Relative pose computation

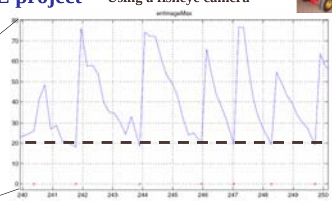
(y, θ) evaluation



Indoor applications : ROSACE project

Using a fisheye camera





Global localisation (6.5s) *306 matchings*
Visual route (280 key images) – Matching mean=108
- ErrImageMax:longest distance between an image point and its position in desired image
- Errpoints:Mean distance between those points

When errImageMax reaches 20 we decide to change key image

Indoor applications : ROSACE project

Using a fisheye camera

[Courbon08]

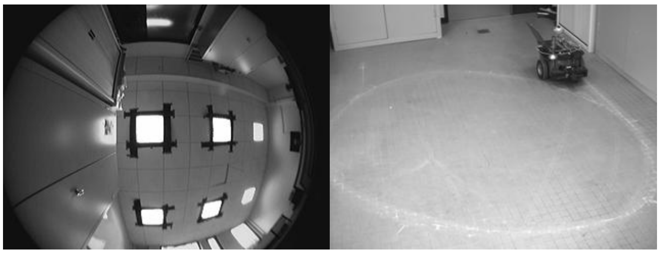
Indoor applications : ROSACE project

Using a fisheye camera

J. Courbon [07]

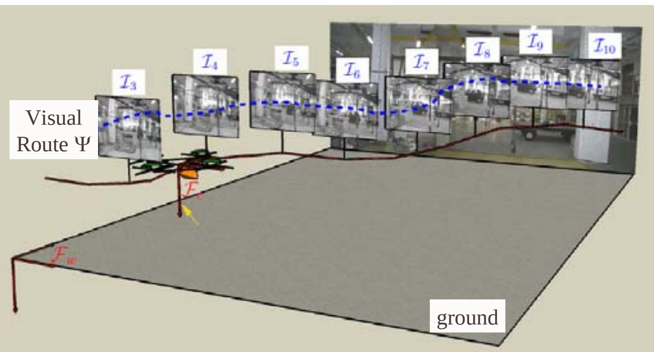
Automatic Guided Vehicles
robot control using a fisheye lens

Autonomous navigation using a visual memory



Indoor applications : µdrone project

Flying robot
[IROS09]



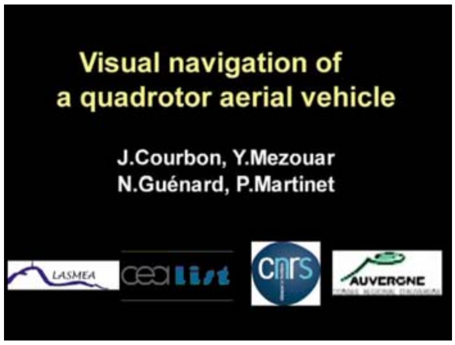
Indoor applications : µdrone project

Flying robot
[IROS09]

Topological navigation using visual memory

CEA 2009
Fontenay-aux-Roses

Using vision only
Clermont-Ferrand
LASMEA-GRAVIR



Outdoor applications : MOBIVIP project

urban context **[Royer04]**

Visual memory

Reference Trajectory

and

Image features (3D points)

Experimental data Compiegne city center (BODEGA/MOBIVIP)

Outdoor applications : MOBIVIP project

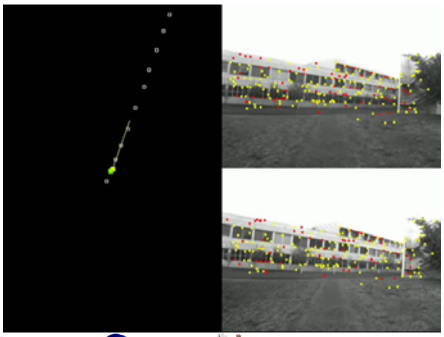
urban context **[Royer04]**

Perception Localization using monocular camera

Reference Trajectory

and

localization



Key images And features

Current images and features

Outdoor applications : MOBIVIP project

urban context **[Royer06]**

Perception and control

2006: Les Cezeaux Campus CFd

2006: Inner city of Clermont-Fd

Outdoor applications : CITYVIP project

[ICRA09] **urban context**

General framework :

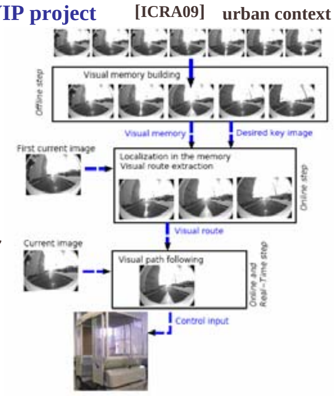
Vision based memory navigation strategy :

3 steps :

1. Visual memory building

2. Localization into the visual memory

3. Navigation into the visual memory



Outdoor applications : CITYVIP project

[ICRA09] **urban context**

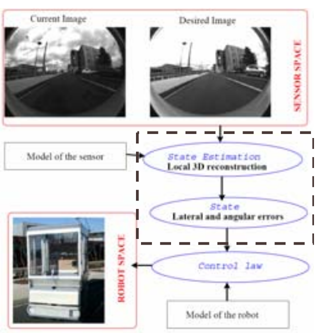
Localization

Vision based memory navigation strategy :

Localization into the visual memory

a- Global localization
b- Selection of key images
c- Relative pose computation

(y, θ) evaluation



Outdoor applications : CITYVIP project

[ICRA09] **urban context**

Testbed

Experimental robot is a Robucab from Robosoft

Algorithms are implemented in C^{++} language on a laptop using RTAI-Linux OS with a 2GHz Centrino processor

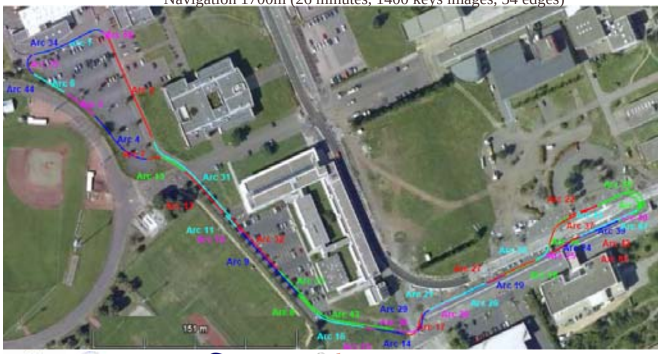
Fujinon fisheye lens, mounted onto a
Marlin F131B camera
Field-of-view of 185 deg
Image resolution in the experiments was 800×600 pixels
Frame rate of 15fps

Longitudinal velocity V has been fixed to $1\,ms^{-1}$
K_p and K_d are tuned regarding a double pole located at value 0.3

Outdoor applications : CITYVIP project [ICRA09] urban context

Experiment

Loop 1200 m – 35 edges
Navigation 1700m (26 minutes, 1400 keys images, 54 edges)

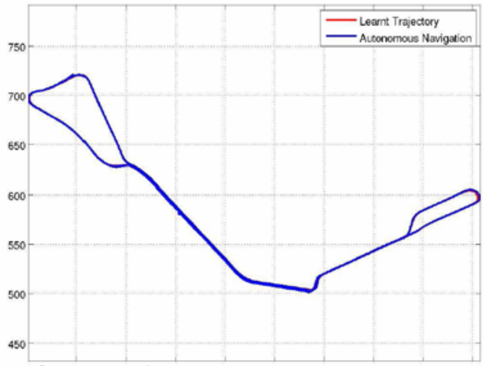


Outdoor applications : CITYVIP project [ICRA09] urban context

Experiment

Lateral error:
- mean: 23 cm
-standard deviation: 30 cm.

Navigation stops after 1700m for security reasons (small number of matched points)



Outdoor applications : CITYVIP project [ICRA09] urban context

Video

Topological navigation using visual memory

Cityvip
Clermont-Fd 2008

Using vision only
Clermont-Ferrand
LASMEA-GRAVIR

Outline of the presentation

Introduction
- LASMEA
- Sensor based Control
- Navigation
- SLAN vs SLAM

A generic model
- A unified model for camera
- Case of fisheye cameras
- Partial euclidian reconstruction

Togological navigation system
- Global system
- Case of vision
- Differents Modules
- SOVIN

Illustrations
- Indoor applications
- Outdoor applications

Discussion
- Learning process
- Updating process
- Closing the loop problem

Learning process

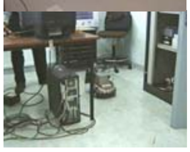
Part of the next step: R-DISCOVER project
Automatic discovering & learning the environment

Updating process

Part of the next step: R-DISCOVER project
Automatic updating the environment

Closing the loop problem

No a priori needs for closing the loop.
Can be useful for improving topological maps



Thanks for your attention

Any questions

http://wwwlasmea.univ-bpclermont.fr/Control
http://wwwlasmea.univ-bpclermont.fr/Personnel/Philippe.Martinet

martinet@lasmea.univ-bpclermont.fr