Efficient Visual Memory Based Navigation of Indoor Robot with a Wide-field of view Camera

Jonathan Courbon*†, Youcef Mezouar†, Laurent Eck*, Philippe Martinet†
*CEA, List
18 route du Panorama, BP6
F- 92265 FONTENAY AUX ROSES - FRANCE
Email: laurent.eck@cea.fr

†LASMEA
24 Avenue des Landais
63177 AUBIERE - FRANCE
Email: firstname.lastname@lasmea.univ-bpclermont.fr

***Abstract*—In this paper, we present a complete framework for autonomous indoor robot navigation. We show that autonomous navigation is possible in indoor situation using a single camera and natural landmarks. When navigating in an unknown environment for the first time, a natural behavior consists on memorizing some key views along the performed path, in order to use these references as checkpoints for a future navigation mission. The navigation framework for wheeled robots presented in this paper is based on this assumption. During a human-guided learning step, the robot performs paths which are sampled and stored as a set of ordered key images, acquired by an embedded camera. The set of these obtained visual paths is topologically organized and provides a visual memory of the environment. Given an image of one of the visual paths as a target, the robot navigation mission is defined as a concatenation of visual path subsets, called visual route. When running autonomously, the control guides the robot along the reference visual route without explicitly planning any trajectory. The control consists on a vision-based control law adapted to the nonholonomic constraint. The proposed framework has been designed for a generic class of cameras (including conventional, catadioptric and fish-eye cameras). Experiments with a AT3 Pioneer robot navigating in an indoor environment have been carried on with a fisheye camera. Results validate our approach.**

***Index Terms*—Autonomous robot, visual memory-based navigation, indoor environment, non-holonomic contraints**

I. Introduction

Often used among more "traditional" embedded sensors - proprioceptive sensors like odometers as exteroceptive ones like sonars - vision sensor provides accurate localisation methods. The authors of [1] accounts of twenty years of works at the meeting point of mobile robotics and computer vision communities. In many works, and especially those dealing with indoor navigation as in [2], computer vision techniques are used in a landmark-based framework. Identifying extracted landmarks to known references allows to update the results of the localisation algorithm. These methods are based on some knowledges about the environment, such as a given 3D model or a map built online. They generally rely on a complete or partial 3D reconstruction of the observed environment through the analysis of data collected from disparate sensors. The mobile robot can thus be localized in an absolute reference frame. Both motion planning and robot control can then be designed in this space. The results obtained by the authors of [3] leave to be forcasted that such a framework will be reachable using a single camera. However, although an accurate global localisation is unquestionably useful, our aim is to build a complete vision-based framework without recovering the position of the mobile robot with respect to a reference frame. In [1] this type of framework is ranked among qualitative approaches.

The principle of this approach is to represent the robot environment with a bounded quantity of images gathered in a set called visual memory. In the context of mobile robotics, [4] proposes to use a sequence of images, but recorded during a human teleoperated motion, and called View-Sequenced Route Reference. This concept underlines the close link between a human-guided learning and the performed paths during an autonomous run. However, the automatic control of the robot in [4] is not formulated as a visual servoing task. In [5] (respectively in [6]), we propose to achieve navigation task using a conventional camera (respectively an omnidirectional camera) looking at the ceiling toward recovering features observed during a learning stage. The observed features are artificial landmarks and are supposed to belong to a plane. In this paper, a fisheye camera pointing forward is embedded onto the robot and natural landmarks are extracted. A sequence of images, acquired during a human-guided learning step, allows to derive paths driving the robot from its initial to its goal locations. In order to reduce the complexity of the image sequences, only key views are stored and indexed on a visual path. The set of visual paths can be interpreted as a visual memory of the environment. A navigation task consists then in performing autonomously a visual route which is a concatenation of visual paths. The visual route connects thus in the sensor space the initial and goal configurations. The control law, adapted to the non-holonomic contraints of the robot, is computed from the matched points.

Section II details the concept of visual memory. Panoramic views acquired by an omnidirectional camera are well adapted to this approach since they provide a large amount of visual features which can be exploited as well as for localisation than for visual servoing. The Section III deals with the omnidirectional vision-based control scheme designed to control the robot motions along a visual route. Finally, in Section IV, simulations and experiments on a Pioneer AT3 mobile robot (refer to Fig. 1) illustrate the proposed framework with two

978–1–4244–2287–6/08/$25.00

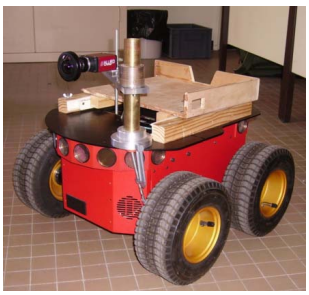


Fig. 1. Mobile robot Pioneer 3 equipped with a fisheye camera

experimentations.

II. Vision-based memory navigation strategy

Our approach can be divided in three steps 1) visual memory building, 2) localization into the visual memory, 3) navigation into the visual memory (refer to Fig. 2).

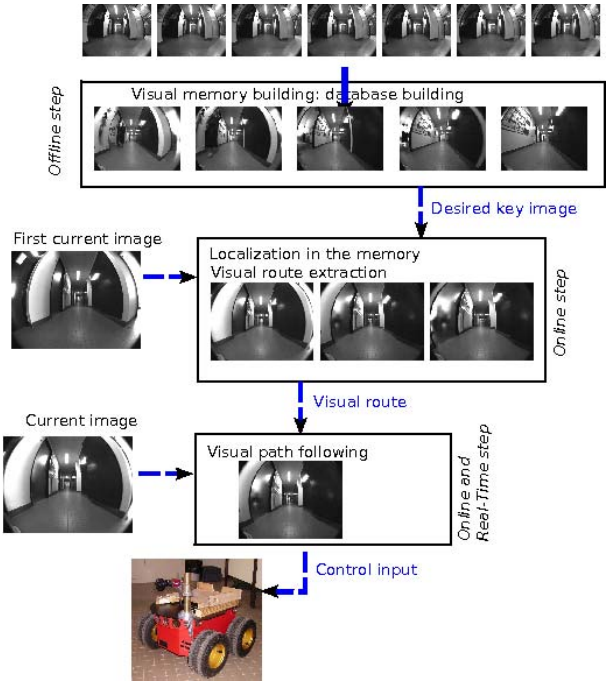


Fig. 2. Principle of our VMBN approach

A. Visual Memory Structure

The visual memory structure is summarized in the following (more details can be found in [6]). The learning stage relies on the human experience. The user guides the robot along one or several paths into each place p where the robot is authorized to go. A visual path Ψ^p is then stored. It is a weighted directed graph composed of n successive key images (*vertices*):

$$\Psi^p = \{\mathcal{I}_i^p | i = \{1, 2, \ldots, n\}\}$$

For control purpose, the authorized motions during the learning stage are assumed to be limited to those of a car-like robot, which only goes forward. Moreover, because the controller is vision-based, the robot is controllable from $\mathcal{I}_i^p$ to $\mathcal{I}_{i+1}^p$ *i.e.* the images contain a set $\mathcal{P}_i$ of matched visual features, which can be observed along a path performed between ${}^R\mathcal{F}_i$ and ${}^R\mathcal{F}_{i+1}$ and which allows the computation of the control law. The key images selection process, constrained by this condition, is detailled in Section II-C. Each visual path Ψ^p corresponds to an oriented edge which connects two configurations of the robot's workspace. The number of key images of a visual path is directly linked to the human-guided path complexity and to the key image selection process. Two visual paths Ψ^{p_1} and Ψ^{p_2} are connected when the terminal extremity of a visual path Ψ^{p_1} is the same key image as the initial extremity of another visual path Ψ^{p_2}. The visual memory structure is thus defined as a multigraph which vertices are key images linked by edges which are the visual paths (*directed graphs*). Note that more than one visual path may be incident to a node. It is yet necessary that this multigraph is strongly connected. Indeed, this condition warrants that any vertex of the visual memory is attainable from every others, through a set of visual paths.

B. Visual route

A visual route describes the robot's mission in the sensor space. Given the images $\mathcal{I}_c^*$ and $\mathcal{I}_g$, a visual route is a set of key images which describes a path from $\mathcal{I}_c^*$ to $\mathcal{I}_g$. $\mathcal{I}_c^*$ is the closest key image to the current image of the robot. This current image is determined by extracting the image $\mathcal{I}_c^*$ from the visual memory during a localization step, as described in Section II-D. A visual route is chosen as a path of the visual memory connecting two vertices associated to $\mathcal{I}_c^*$ and $\mathcal{I}_g$.

C. Key-images selection

A central clue for implementation of almost all the vision-based frameworks relies on point matching. Indeed this process takes places in all steps of the proposed navigation framework. It allows key images selection during the learning stage, of course it is also usefull during the autonomous navigation to provide the necessary input for state estimation. We use a similar process that the one proposed in [3] and successfully applied for the metric localization of autonomous robots in outdoor environment. Interest points are detected in each image with Harris corner detector [7]. For an interest point $\mathcal{P}_1$ at coordinates $(x\, y)$ in image $\mathcal{I}_i$, we define a search region in image $\mathcal{I}_{i+1}$. The search region is a rectangle whose center has coordinates $(x\, y)$. For each interest point $\mathcal{P}_2$ inside the search region in image $\mathcal{I}_{i+1}$, we compute a score between the neighborhoods of $\mathcal{P}_1$ and $\mathcal{P}_2$ using a Zero Normalized Cross Correlation. The point with the best score and upper than a threshold is kept as a good match and the unicity constraint is used to reject matches which have become impossible. This method is illumination invariant and its computational cost is small.

The first image of the video sequence is selected as the first key frame $\mathcal{I}_1$. A key frame $\mathcal{I}_{i+1}$ is then chosen so that there are as many video frames as possible between $\mathcal{I}_i$ and $\mathcal{I}_{i+1}$ while there are at least $\mathcal{M}$ common interest points tracked between $\mathcal{I}_i$ and $\mathcal{I}_{i+1}$.

D. Localization

The localization step consists on finding the image of the memory which best fits the current image acquired by the embedded camera. It is thus necessary to match the current image with all the images of the memory. Two main strategies exist to match images: the image can be represented by a single descriptor (*global approaches*) [8] or alternatively by a set of descriptors defined around visual features (*landmarks-based* or *local approaches*) [9]. In this paper, we use a landmark-based approach. Points between the current and a key image are matched. The distance between these images is the inverse of the number of matched points.

III. Routes following using an omnidirectional camera

As the camera is embedded and fixed, the control of the camera is based on wheeled mobile robots control theory [10]. A visual route following can be considered as a sequence of visual-servoing tasks. A stabilization approach could thus be used to control the camera motions from a key image to the next one. However, a visual route is fundamentally a path. To design the controller, described in the sequel, the key images of the reference visual route are considered as consecutive checkpoints to reach in the sensor space. The control problem is formulated as a path following to guide the nonholonomic mobile robot along the visual route.

A. Model and assumptions

Let $\mathcal{I}_i$, $\mathcal{I}_{i+1}$ be two consecutive key images of a given visual route to follow and $\mathcal{I}_c$ be the current image. Let us note $\mathcal{F}_i = (O_i, \mathbf{X_i}, \mathbf{Y_i}, \mathbf{Z_i})$ and $\mathcal{F}_{i+1} = (O_{i+1}, \mathbf{X_{i+1}}, \mathbf{Y_{i+1}}, \mathbf{Z_{i+1}})$ the frames attached to the robot when $\mathcal{I}_i$ and $\mathcal{I}_{i+1}$ were stored and $\mathcal{F}_c = (O_c, \mathbf{X_c}, \mathbf{Y_c}, \mathbf{Z_c})$ a frame attached to the robot in its current location. Figure 3 illustrates this setup. The origin O_c of $\mathcal{F}_c$ is on the axle midpoint of a car-like robot, which evolutes on a perfect ground plane.

The control vector of the considered car-like robot is $\mathbf{u} = [V, \omega]^T$ where V is the longitudinal velocity along the axle $\mathbf{Y_c}$ of $\mathcal{F}_c$, and ω is the rotational velocity around $\mathbf{Z_c}$. The hand-eye parameters (*i. e.* the rigid transformation between $\mathcal{F}_c$ and the frame attached to the camera) are supposed to be known.

The state of a set of visual features $\mathcal{P}_i$ is known in the images $\mathcal{I}_i$ and $\mathcal{I}_{i+1}$. Moreover $\mathcal{P}_i$ has been tracked during the learning step along the path ψ between $\mathcal{F}_i$ and $\mathcal{F}_{i+1}$. The state of $\mathcal{P}_i$ is also assumed available in $\mathcal{I}_c$ (*i.e* $\mathcal{P}_i$ is in the camera field of view). The task to achieve is to drive the state of $\mathcal{P}_i$ from its current value to its value in $\mathcal{I}_{i+1}$.

B. Principle

Consider the straight line $\Gamma' = (O_{i+1}, \mathbf{Y}_{i+1})$ (see Figure 3). The control strategy consists in guiding $\mathcal{I}_c$ to $\mathcal{I}_{i+1}$ by regulating asymptotically the axle $\mathbf{Y}_c$ on Γ'. The control objective is achieved if $\mathbf{Y}_c$ is regulated to Γ' before the origin of $\mathcal{F}_c$ reaches the origin of $\mathcal{F}_{i+1}$. This can be done using chained systems. Indeed in this case chained system properties are very interesting. A chained system results from

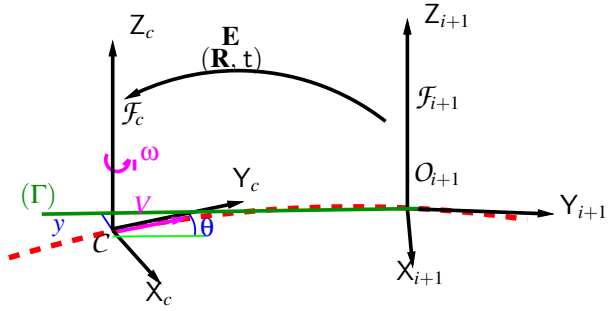


Fig. 3. The frame $\mathcal{F}_{i+1}$ along the trajectory Γ is the frame where the desired image $\mathcal{I}_{i+1}$ was acquired. The current image $\mathcal{I}_c$ is situated at the frame $\mathcal{F}_c$.

a conversion of a mobile robot non linear model into an almost linear one [10]. As long as the robot longitudinal velocity V is non zero, the performances of path following can be determined in terms of settling distance [11]. The settling distance has to be chosen with respect to robot and perception algorithm performances.

The lateral and angular deviations of $\mathcal{F}_c$ with respect to Γ' to regulate can be obtained through partial Euclidean reconstructions as described in the next section.

C. Control law

Exact linearization of nonlinear models of wheeled mobile robot under the assumption of rolling without slipping is a well known theory, which has already been applied in many robot guidance applications, as in [11] for a car-like robot, and in our previous works (see [5]). The used state vector of the robot is $\mathsf{X} = \begin{bmatrix} s & y & \theta \end{bmatrix}^\top$, where s is the curvilinear coordinate of a point $\mathcal{M}$, which is the orthogonal projection of the origin of $\mathcal{F}_c$ on Γ'. The derivative of this state give the following state space model:

$$\begin{cases} \dot{s} = V\cos\theta \\ \dot{y} = V\sin\theta \\ \dot{\theta} = \omega_c \end{cases} \tag{1}$$

The state space model (1) may be converted into a 3-dimensional chained system: $\begin{bmatrix} a_1 & a_2 & a_3 \end{bmatrix}^T$. Deriving this system with respect to a_1 provides an almost linear system. As proposed in [5] and thanks to classical linear automatics theory, an asymptotically stable guidance control law can be designed by choosing $a_1 = s$ and $a_2 = y$:

$$\omega(y, \theta) = -V\cos^3\theta K_p y - |V\cos^3\theta| K_d \tan\theta \tag{2}$$

The control law (2) is theorically independant to the longitudinal velocity V. K_p and K_d are two positive gains which set the performances of the control law. Their choice determine a settling distance for the control, *i. e.* the impulse response of y with respect to the covered distance by the point $\mathbf{M}$ on Γ'.

D. State estimation from the unified model of camera on the sphere

In our application, a fisheye camera has been used to obtain omnidirectional views. A classical model for catadioptric cam-

eras is the unified model on the sphere [12]. It has been shown in [13] that this model is also available for fisheye cameras in the case of robotic applications. First, the model of such a sensor is briefly described and then the epipolar geometry is exploited to extract the state of the robot.

1) Camera model: The projection can be modeled by a central projection onto a virtual unitary sphere followed by a perspective projection onto an image plane [12] (refer to Figure 4). This generic model is parametrized by the parameter ξ describing the type of sensor.

The coordinates x_i of the point in the image plane corresponding to the 3D point $\mathcal{X}$ are obtained after two steps:

Step 1 : Projection of the 3D point $\mathcal{X}$ of coordinates $\mathsf{X} = [X\ Y\ Z\ ]^T$ on the unit sphere: $\mathsf{X_m} = \mathsf{X}/\|\mathsf{X}\|$

Step 2 : Perspective projection on the normalized image plane $Z = 1 - \xi$ and plane-to-plane collineation. The coordinates $\underline{\mathsf{x}}_\mathsf{i}$ in the image plane is

$$\underline{\mathsf{x}}_i = \mathbf{K}_c\mathbf{M}\left[\frac{X}{Z+\xi\|\mathsf{X}\|} \quad \frac{Y}{Z+\xi\|\mathsf{X}\|} \quad 1\right] \tag{3}$$

$\mathbf{K}_c$ contains the usual intrinsic parameters of the perspective camera and $\mathbf{M}$ contains the parameters of the frames changes.

We notice that X_m can be computed as a function of the coordinates in the image $\bar{\mathsf{x}}$ and the sensor parameter ξ:

$$\mathsf{X}_m = (\eta^{-1} + \xi)\bar{\mathsf{x}} \tag{4}$$
$$\bar{\mathsf{x}} = \left[\mathsf{x}^T \quad \frac{1}{1+\xi\eta}\right]^T$$

with: $\begin{cases} \eta = \dfrac{-\gamma - \xi(x^2+y^2)}{\xi^2(x^2+y^2)-1} \\ \gamma = \sqrt{1+(1-\xi^2)(x^2+y^2)} \end{cases}$.

2) Scaled Euclidean reconstruction: Let $\mathcal{X}$ be a 3D point with coordinates $\mathsf{X_c} = [X_c\ Y_c\ Z_c\ ]^T$ in the current frame $\mathcal{F}_c$ and $\mathsf{X}^* = [X_{i+1}\ Y_{i+1}\ Z_{i+1}\ ]^T$ in the frame $\mathcal{F}_{i+1}$. Let $\mathsf{X_m}$ and X^*_m be the coordinates of those points, projected onto the unit sphere (refer to Fig. 4). The epipolar plane contains the projection centers O_c and O_{i+1} and the 3D point X. $\mathsf{X_m}$ and X^*_m clearly belong to this plane. The coplanarity of those points is traduced by the relation:

$$\mathsf{X_m}^T\mathbf{R}(\mathsf{t}\times\mathsf{X_m}^{*T}) = \mathsf{X_m}^T\mathbf{R}\,[\mathsf{t}]_\times\,\mathsf{X_m}^{*T} = 0 \tag{5}$$

where $\mathbf{R}$ and t represent the rotational matrix and the translational vector between the current and the desired frames. Similarly to the case of pinhole model, the relation (5) can be written:

$$\mathsf{X_m}^T\mathbf{E}\mathsf{X_m}^{*T} = 0 \tag{6}$$

where $\mathbf{E} = \mathbf{R}\,[\mathsf{t}]_\times$ is the essential matrix [14]. The essential matrix $\mathbf{E}$ between two images is estimated using five couples of matched points as proposed in [15] if the camera calibration (matrix $\mathbf{K}$) is known. Outliers are rejected using a random sample consensus (RANSAC) algorithm. From the essential matrix, the camera motion parameters (that is the rotation $\mathbf{R}$ and the translation t up to a scale) can be determined. Finally, the estimation of the input of the control law (2), *i.e* the angular deviation θ and the lateral deviation y, are computed straightforwardly from $\mathbf{R}$ and t.

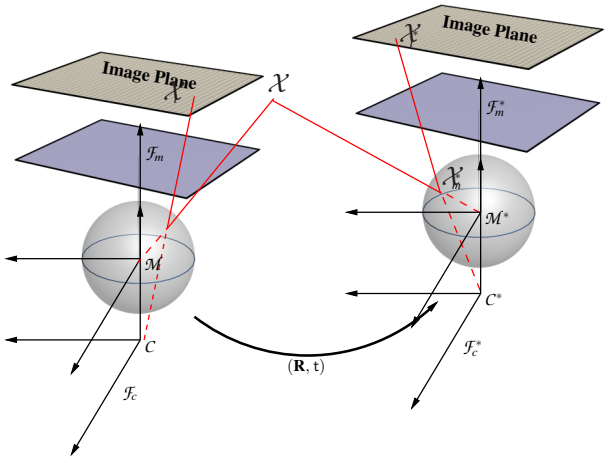


Fig. 4. Geometry of two views.

IV. Experimentations

Our experimental robot is a Pioneer 3AT (refer to Fig. 1). Vision and guidance algorithms are implemented in C^{++} language on a laptop using RTAI-Linux OS with a 2GHz Centrino processor. The Fujinon fisheye lens, mounted onto a Marlin F131B camera, has a field-of-view of $185\,\mathrm{deg}$. The image resolution in the experiments was 800×600 pixels. It has been calibrated using the Matlab toolbox presented in [16]. The camera, looking forward, is situated at approximately 30cm from the ground. The parameters of the rigide transformation between the camera and the robot control frames are roughly estimated. Grey level images are acquired at a rate of 15fps. A learning stage has been conducted off-line and images have been memorized as proposed in Section II. Several paths have been memorized. In a first experimentation, the robot is navigating indoor while in the second, it starts indoor and ends outdoor.

The navigation task has been started near the visual route to follow. First, the robot localized itself in the memory as explained in Section II-D: for each image of the visual memory, points are extracted and matched to the image points of the current image. The image with the smaller distance (*i.e* the higher number of matched features) is kept as the localization result. Given a goal image, a visual path has been extracted.

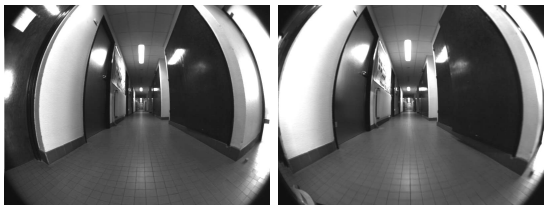
At each frame, points are extracted from the current image and matched with the desired key image. A robust partial reconstruction is then applied using the current, desired and the former desired images of the memory. Angular and lateral errors are extracted and allow the computation of the control law (5). A key image is supposed to be reached when one of the "image errors" is smaller than a fixed threshold. In our experiment, we have considered two "image errors": the longer distance between an image point and its position in the desired

key image (errImageMax) and the mean distance between those points (errPoints), expressed in pixels. The longitudinal velocity V has been fixed to $200\,mms^{-1}$. K_p and K_d have been set in order that error presents a double pole located at value 0.3. For safety, the absolute value of the control input is bounded to 10 degree by second.

A. First experimentation

The current image (Fig. 5 (a)) is localized in the image (Fig. 5 (b)) with 306 matchings in 6.5 s. This visual route to follow contains 280 key images.

The image errors are represented in Fig. 6. A mean of 108



(a) (b)

Fig. 5. Result of the localization step: the current image (a) and the nearest key image (b) have 306 matched points.

good matching for each frame has been found. For the state estimations, the mean reprojection error in the three views was around 2.7 pixels. The image error is decreasing until a new key image is reached (refer to Fig. 7).

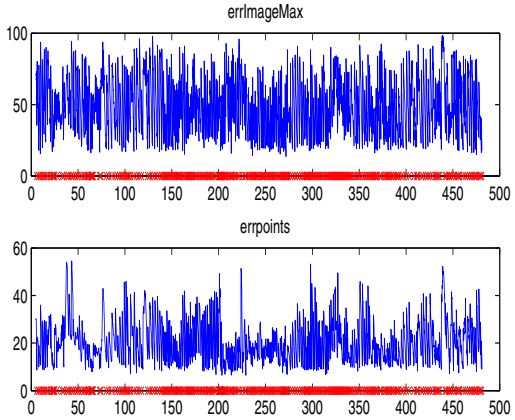


Fig. 6. Image errors: errImageMax and errPoints (expressed in pixels) vs time (in s).

Lateral and angular errors as well as control input are represented in Fig. 8. As it can be noticed, those errors are well regulated to zero for each key view. The discontinuities are due to the transition between two successive key images. At the beginning of the navigation, some errors are large because the robot starts far from the trajectory done offline.

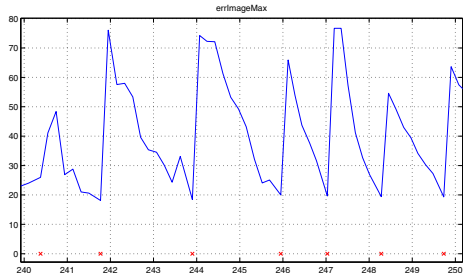


Fig. 7. A zoom on the image error (errImageMax). One of the condition for key image change was when errImageMax<20 pixels.

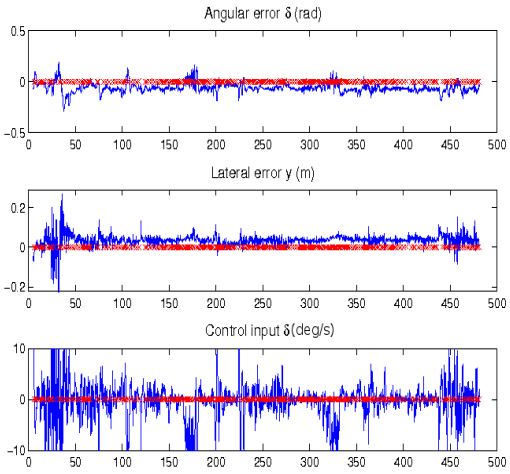
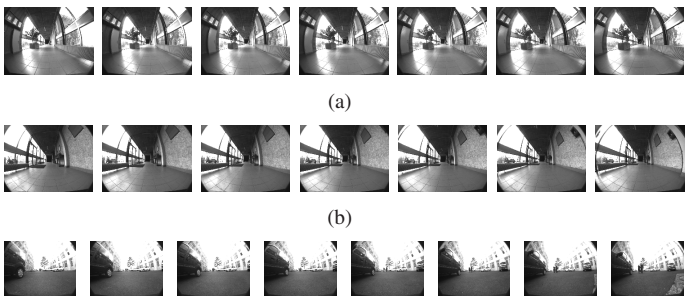


Fig. 8. Lateral y and angular θ errors and control input δ vs time (in s).

B. Second experimentation

The robot is positionned into a building (refer to Fig. 9 (a) and (b)) and then go out (Fig. 9 (c)).

Lateral and angular errors as well as control input are



(a) (b) (c)

Fig. 9. Parts of the visual path to follow (2nd experimentation).

represented in Fig. 11. As in the first experimentation, those errors are well regulated to zero for each key view. The image errors are also decreasing before reaching the key views (refer to Fig. 10).

As it can be noticed in Fig. 12, our method is robust to

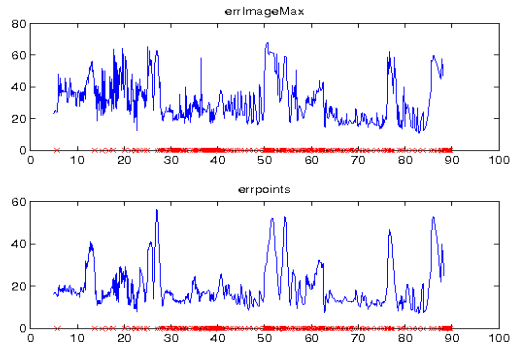


Fig. 10. Image errors: (errImageMax) and (errPoints) vs time (2nd experimentation)

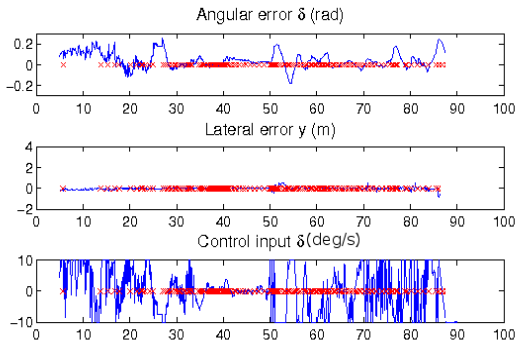


Fig. 11. Lateral y and angular θ errors and control input δ vs time (2nd experimentation)

changes in the environment. A man was going in the direction of the robot (at the left) during the manually driven step whereas a man is walking at the right of the pioneer during the autonomous navigation.

(a) (b)

Fig. 12. The image (a) corresponds to the reached key image (b) of the visual memory (2nd experimentation).

This experimentation shows that our method is not restricted to indoor environments. Here, the robot starts indoor and goes outdoor. Any parameter of the algorithm has been changed during the experimentation.

V. Conclusion and Future Works

In this paper a sensor-based navigation framework for a non-holonomic mobile robot has been presented. The framework is presented in the context of an indoor navigation and using as sensor a fisheye camera. We show that this autonomous navigation is possible using a single camera and natural landmarks. The navigation mission consists in following a visual route in a visual memory composed of omnidirectional key-images. The vision-based control law is realized in the sensor space, and it is adapted to the robot non-holonomy. Feature matching and tracking are well adapted to real-time application and have shown to be robust to occlusions and changes in the environment.

Acknowledgment

This work is supported by the EU-Project FP6 IST μDrones, FP6-2005-IST-6-045248.

References

[1] G. N. DeSouza and A. C. Kak, "Vision for mobile robot navigation: A survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 24, no. 2, pp. 237–267, february 2002.

[2] J. Hayet, F. Lerasle, and M. Devy, "A Visual Landmark Framework for Indoor Mobile Robot Navigation," in *Proc. Int. Conf. on Robotics and Automation (ICRA'02)*, Washington DC, USA, 2002, pp. 3942–3947.

[3] E. Royer, M. Lhuillier, M. Dhome, and J.-M. Lavest, "Monocular vision for mobile robot localization and autonomous navigation," *International Journal of Computer Vision, special joint issue on vision and robotics*, vol. 74, pp. 237–260, 2007.

[4] Y. Matsumoto, M. Inaba, and H. Inoue, "Visual navigation using view-sequenced route representation," in *IEEE International Conference on Robotics and Automation, ICRA'96*, vol. 1, Minneapolis, Minnesota, USA, April 1996, pp. 83–88.

[5] G. Blanc, Y. Mezouar, and P. Martinet, "Indoor navigation of a wheeled mobile robot along visual routes," in *IEEE International Conference on Robotics and Automation, ICRA'05*, Barcelona, Spain, 2005, pp. 3365–3370.

[6] J. Courbon, G. Blanc, Y. Mezouar, and P. Martinet, "Navigation of a non-holonomic mobile robot with a memory of omnidirectional images," in *ICRA 2007 Workshop on "Planning, perception and navigation for Intelligent Vehicles"*, Rome, Italy, 2007.

[7] C. Harris and M. Stephens, "A combined corner and edge detector," in *Alvey Conference*, 1988, pp. 189–192.

[8] Y. Matsumoto, K. Ikeda, M. Inaba, and H. Inoue, "Visual navigation using omnidirectional view sequence," in *International Conference on Intelligent Robots and Systems, IROS'99*, vol. 1, Korea, 1999, pp. 317–322.

[9] T. Goedemé, T. Tuytelaars, G. Vanacker, M. Nuttin, L. V. Gool, and L. V. Gool, "Feature based omnidirectional sparse visual path following," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Edmonton, Canada, August 2005, pp. 1806–1811.

[10] C. Samson, "Control of chained systems. application to path following and time-varying stabilization of mobile robots," *IEEE Transactions on Automatic Control*, vol. 40, no. 1, pp. 64–77, 1995.

[11] B. Thuilot, J. Bom, F. Marmoiton, and P. Martinet, "Accurate automatic guidance of an urban electric vehicle relying on a kinematic GPS sensor," in *5th IFAC Symposium on Intelligent Autonomous Vehicles, IAV'04*, Instituto Superior Técnico, Lisbon, Portugal, July 5-7th 2004.

[12] C. Geyer and K. Daniilidis, "Catadioptric projective geometry," in *International Journal of Computer Vision*, vol. 43, 2001, pp. 223–243.

[13] J. Courbon, Y. Mezouar, L. Eck, and P. Martinet, "A generic fisheye camera model for robotic applications," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'07*, San Diego, CA, USA, 29 October - 2 November 2007, pp. 1683–1688.

[14] T. Svoboda and T. Pajdla, "Epipolar geometry for central catadioptric cameras," *International Journal of Computer Vision*, vol. 49, no. 1, pp. 23–37, 2002.

[15] D. Nistér, "An efficient solution to the five-point relative pose problem," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 6, pp. 756–770, 2004.

[16] C. Mei, S. Benhimane, E. Malis, and P. Rives, "Constrained multiple planar template tracking for central catadioptric cameras," in *British Machine Vision Conference, BMVC'06*, Edinburgh, UK, 2006.