2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

2D visual servoing for a long range navigation in a cluttered environment

A. Durand Petiteville, S. Hutchinson, V. Cadenat and M. Courdesses

Abstract— **In this paper, we consider the problem of realizing a vision-based long range navigation in a cluttered environment. To this aim, we couple a topological environment representation and a supervision algorithm. The latter organizes several elementary tasks which allow to perform image based visual servoing, obstacle avoidance and to estimate on line several required parameters. Well known and new methods based on the tasks function approach are introduced.The dynamical sequencing formalism is used to guarantee the global control law smoothness. Finally, simulation results validate our approach.**

I. INTRODUCTION

We address the problem of the navigation of a nonholonomic mobile robot based on visual informations. In the survey [Bonin-Font et al., 2008], there exists two main approaches: map-based navigation and mapless navigation. The first method is subdivided into: metric map-using navigation systems (MMUNS), metric map-building navigation systems (MMBNS) and topological map-based systems (TMBS). MMUNS require a complete map of the environment before the navigation starts. MMBNS build the map by themselves then use it during the navigation. Finally, TMBS build and/or use topological maps which consist of nodes linked by lines where each node represents the most characteristic places of the environment [Gaspar et al., 2000]. Mapless navigation mostly include reactive techniques that use visual clues derived from the segmentation of an image [Chaumette and Hutchinson, 2006] or optical flow [Santos-Victor and Sandini, 1995].

We aim to realize a long range navigation in a cluttered environment using the minimal *a priori* and *a posteriori* knowledges. The MMUNS and MMBNS need and/or provide a too rich information to be considered in our case. Several appropriate approaches based on reactive techniques coupled with a topological map have been proposed. In [Blanc et al., 2005], an indoor teleoperated pre-navigation is realized in order to record the image stream. Then, a visual route, comparable to a topological map, is built thanks to the key frames extracted from the stream. The key frames correspond to robot close configurations to ensure the local visual servoing stability. Finally, to reach the goal the robot has to navigate with the successive key frames. Thus, the path followed during the pre-navigation step is replayed thanks to the images. A similar approach is proposed in [Kralnìk and Přeučil, 2008] in an outdoor environment. Nevertheless, the obstacle avoidance is not taken into account in these methods. In [Cherubini and Chaumette, 2010], a controller allowing the robot to perform an obstacle avoidance is proposed. The previous methods involve two main drawbacks. First, numerous key frames compose the image data base to perform a long range navigation. Second, the navigation method requires to develop new necessary controllers, such as the obstacle avoidance in [Cherubini and Chaumette, 2010], to perform a safe navigation.

The image based visual servoing presented in [Chaumette and Hutchinson, 2006] is a well known and studied technique. It consists in controlling the robot to make the image features converge to the desired ones. These latter, which correspond to a robot configuration with respect to a landmark, can be obtained either during a pre-navigation work or estimated on-line [Durand Petiteville et al., 2010]. In addition to the controller realizing the navigation, several methods have been developed to avoid obstacles and to handle occlusions [Folio and Cadenat, 2008]. Nevertheless, image based visual servoing is usually used to perform short range navigation. Indeed, it requires to see the considered target to compute the control law during all the navigation and especially at the beginning. This condition can be unsatisfied because of the presence of occulting obstacles. If the target to reach cannot be seen at the inital robot position, a solution consists in dividing the global navigation mission into elementary navigation tasks with respect to a set of landmarks. Each intermediate target can be seen from the neighborhood of the previous one. So, in order to extend the navigation range, we propose to add a topological map whose each node corresponds to a landmark. Moreover, a supervision algorithm is developed to organize the tasks. Finally, each task is performed using the task function approach [Samson et al., 1991] and the transition between them are managed using the dynamical sequencing formalism proposed in [Souères and Cadenat, 2003].

The paper is organized as follows. Section 2 is dedicated to the topological map and supervision algorithm description. Section 3 details the several controllers computation. Finally, simulation results validate the proposed approach.

II. CONTROL STRATEGY

In this paper, we propose to realize a long range navigation based on a visual reactive controller and a topological map. The considered robot is a cart-like vehicle equipped with a

Durand Petiteville, Cadenat and Courdesses are with CNRS; LAAS : 7 avenue du colonel Roche, F-31077 Toulouse, France; Université de Toulouse ; UPS, INSA, INP, ISAE ; LAAS ; F-31077 Toulouse, France. [adurandp, cadenat, courdess]@laas.fr

S. Hutchinson is a professor of Electrical and Computer Engineering at the univerrsity of Illinois, USA. seth@illinois.edu

camera mounted on a pan-platform and a laser. The system is detailed in III-A.

A. The topological map

We propose to model the environment by a set $T_G = (T_1, T_2, ..., T_n)$ of n landmarks. For an indoor navigation different kinds of landmarks can be used : corridors, doors or wall edges [Vassalo et al., 2000]. Now, we define a graph G (Fig 1) where each node corresponds to a landmark T_i (for $i \in [1, ..., n]$). A node T_i is linked to a node T_{i+1} if T_{i+1} can be seen from the neighborhood of T_i. If the target to reach cannot be seen from the initial robot configuration, the graph G is used to provide a set T_P of successive targets allowing to perform the whole navigation. T_P only contains $m \leq n$ elements of T_G. To compute the set T_P, we propose to use the Dijkstra's algorithm. For example, in figure 1, the environment is depicted by the whole graph. The computed path allowing to reach T_7 from T_1 is represented in grey. In this case $T_P = [T_{P1}, T_{P2}, T_{P3}, T_{P4}, T_{P5}] = [T_1, T_2, T_4, T_6, T_7]$. Thanks to the topological map, the set of landmarks to reach in order to perform a long range navigation is known.

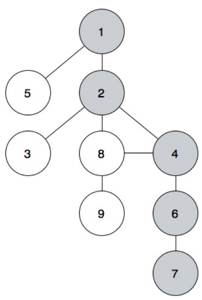


Fig. 1. The topological map

B. The global strategy

To perform a long range navigation in a cluttered environment guaranteeing the robot safety, several tools, such as visual servoing, obstacle avoidance and visual features estimation, are available. These tools are coupled to define a set of useful tasks, such as the visual servoing using estimated visual features to handle occlusion during the navigation. The tasks are placed in an supervision algorithm (Fig 2) in which the events involving a transition between two tasks are described. Thus the task to perform is defined using the robot perception of the environment and the knowledge of the previous task.

The different tasks have to be defined in order to allow the robot to reach successively the targets included in T_P while preserving the system and navigation integrity. The main task [B] is the visual servoing which is performed thanks to a reactive controller allowing the current image data, denoted by s_j, to converge towards the reference ones, noted s_j^* (for $j \in [1, ..., m]$). The latter corresponds to the sensor data values at the desired system configuration with respect to T_{Pj}. During the navigation, if an obstacle is detected in the robot neighbourhood, the algorithm switches to the

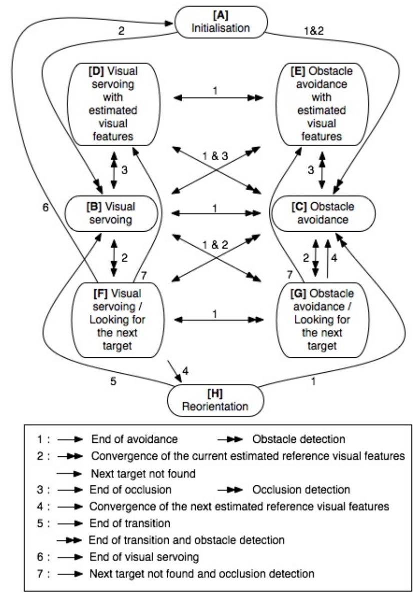


Fig. 2. Supervision algorithm

obstacle avoidance task [C]. This task is performed until the obstacle is no more considered as dangerous. In the same way, the supervisor switches to the tasks with estimated visual features ([D],[E]) if the target is no more perceived.

The desired visual features can be obtained during a pre-navigation step or computed on-line. If the second option is chosen, the estimation process is done during the tasks [A], [F] or [G]. For the first visual servoing s_1^* is estimated during the initialization task [A]. In this case the robot turns on itself to avoid collisions, and to obtain different visual informations used to estimate s_1^*. Once the value of s_1^* has converged, the visual servoing can be started.

For the following visual servoing, the next target T_{Pj+1} to reach needs to be found by the camera during the navigation wrt T_{Pj}. To this aim, during the tasks [F] and [G], the mobile base is controlled to perform a visual servoing or an obstacle avoidance, based on the estimated visual data as in [D] and [E], whereas the camera is looking for the next target by scanning the environment from one of its mechanical bound to the another one. If the target T_{Pj+1} is not found at the end of the scan, the visual servoing wrt the current target T_{Pj} or the obstacle avoidance restarts. Otherwise the camera is controlled to keep the target T_{Pj+1} in its field on view. Then, the obtained visual data are used to estimate s_{j+1}^*. If the target is not found at the end of the visual servoing task, the map is updated and the planning task is restarted.

Once the convergence of s_{j+1}^* is detected, T_{Pj+1} becomes the task to realized and task [H] is launched. This task is defined in section III and allows to guarantee the continuity of the control law when switching between two visual servoing with respect to two different targets.

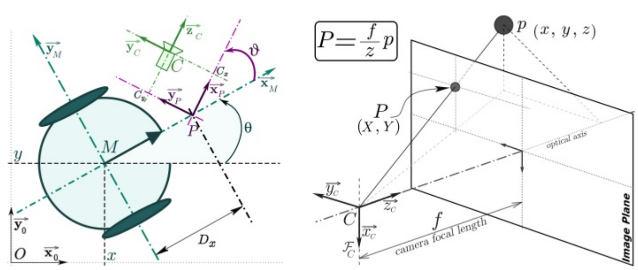
By applying this algorithm for the m targets included in T_P, the robot is able to reach the final target which cannot be seen by the camera at the initial position. Moreover, the robot safety and the navigation success are guaranteed thanks to the use of several tasks allowing the system to overcome the problems of occlusion and collision. However, the continuity problems between each task have to be addressed during the controllers definition.

III. CONTROL ASPECTS

In this section, a set of controllers allowing to perform each task of the supervision algorithm are presented. We choose to compute them thanks to the task function approach in order to use the dynamical sequencing formalism. Thus, the continuity between two tasks is guaranteed. Moreover the different estimation processes are introduced.

A. System modelling

The considered system is a cart-like robot equipped with a camera mounted on a pan-platform. Moreover a laser has been added. Fig 3(a) introduces the robot model, which requires to define the successive frames : F_O attached to the world, F_M linked to the robot, F_P attached to the platform, and F_C linked to the camera. Let θ be the direction of the robot wrt. $\vec{x}_O$, ϑ the direction of the pan-platform wrt. $\vec{x}_M$, P the pan-platform centre of rotation, C_x and C_y the coordinates of C in F_P, and D_x the distance between the robot reference point M and P. Defining vector $q = (l, \theta, \vartheta)^T$ where l is the robot curvilinear abscissa, the control input is given by $\dot{q} = (\upsilon, \omega, \varpi)^T$, where υ and ω are the cart linear and angular velocities, and ϖ is the pan-platform angular velocity wrt. F_M.



(a) The robot model (b) The camera pinhole model

Fig. 3. The robotic system

The pinhole model is used to represent the camera (Fig 3(b)). In this case, a point P, with coordinates (X,Y) in the image plane, is the projection of a point p, with coordinates $(\mathbf{x}, \mathbf{y}, \mathbf{z})$ in F_O. Moreover we define z as the depth in F_C of the projected point p and f as the camera focal.

A landmark, which can be characterized by a number k of points P_i (for $i \in [1, ..., k]$) in the world frame, is represented by a $2k$-dimensional vector s made of the coordinates (X_i, Y_i) in the image plane. For such a robot and visual features, the relation between $\dot{s}$ and the control inputs $\dot{q}$ is given by :

$$\dot{s} = L_{(s,z)} J_r \dot{q} \tag{1}$$

where J_r is the robot jacobian [Folio and Cadenat, 2008]

$$J_r = \begin{pmatrix} -\sin(\vartheta(t)) & D_x\cos(\vartheta(t)) + C_x & C_x \\ \cos(\vartheta(t)) & D_x\sin(\vartheta(t)) - C_y & -C_y \\ 0 & -1 & -1 \end{pmatrix} \tag{2}$$

and $L_{(s,z)} = [L^T_{(P_1)}, ..., L^T_{(P_k)}]^T$ the interaction matrix. $L_{(P_i)}$ is classically given by [Espiau et al., 1992]:

$$L_{(P_i)} = \begin{pmatrix} L_x(s_i, z_i) \\ L_y(s_i, z_i) \end{pmatrix} = \begin{pmatrix} 0 & \frac{X_i}{z_i} & \frac{X_i Y_i}{f} \\ -\frac{f}{z_i} & \frac{Y_i}{z_i} & f + \frac{Y_i^2}{f} \end{pmatrix} \tag{3}$$

B. Controller sequencing using the task function approach

In this section we propose to define the various tasks necessary for the long range navigation, using the task function approach [Samson et al., 1991]. With this method, each task is defined by a vector e, called task function, to vanish. To ensure the continuity between e_i the current task and e_{i+1} the next one, we suggest to use the dynamical sequencing formalism [Souères and Cadenat, 2003]. To ensure the continuity, the tasks have to be admissible at the switching time t_s. A task is admissible if its jacobian can be invertible. For a transition, the control law expresses as follows:

$$\dot{q}(t) = \left[\begin{array}{ll} \dot{q}_i(t) & \forall t \leq t_s \\ \dot{q}_{i+1}(t) & \forall t \geq t_s \end{array} \right. \tag{4}$$

where $\dot{q}_i(t)$ and $\dot{q}_{i+1}(t)$ are respectively the control inputs allowing to make e_i and e_{i+1} vanish. To guarantee the control law smoothness between these controllers, [Souères and Cadenat, 2003] propose to use the dynamical sequencing formalism. The control law is defined as:

$$\dot{q}_{i+1}(t) = J^{-1}_{i+1} \dot{e}^*_{i+1}(t) \tag{5}$$

where $\dot{e}^*_{i+1}(t)$ is computed to ensure the global control input continuity at the first order. We have chosen the dynamic presented in [Mansard and Chaumette, 2007], which allows to set independently the error decreasing speed and the transition time duration. So we obtain :

$$\dot{e}^*_{i+1}(t) = -\lambda e_{i+1}(t) + \rho(t) \tag{6}$$

where

$$\rho(t) = [J_{i+1} J_i^{-1} \dot{e}_i(t_s) + \lambda e_{i+1}(t_s)] \exp^{-\tau(t - t_s)} \tag{7}$$

with $\lambda > 0$ and $\tau > 0$. Choosing $\dot{e}_i(t_s) = J_i \dot{q}_i(t_s)$ ensures the control law smoothness at $t = t_s$, as the dynamics of the task function e_{i+1} at $t = t_s$ depends on the control law $\dot{q}_i(t)$.

C. The controllers

In this section, the necessary controllers to perform the different previously introduced tasks are presented. The visual servoing and obstacle avoidance controllers come from former works, whereas the reorientation one is developed for the first time.

1) Visual servoing: The vision-based navigation task consists in positioning the camera with respect to a given static landmark using visual data. The value of the visual features s depends on the relative camera position with respect to the landmark. So the goal of the image based visual servoing is to make the current visual signals s converge to their reference values s^*. s^* then corresponds to the value of s obtained at the desired camera pose with respect to the target.

To perform the desired vision-based task, we apply the visual servoing technique given in [Espiau et al., 1992] to mobile robots as in [Pissard-Gibollet and Rives, 1995]. The proposed approach consists in expressing the visual servoing task by the following task function to be regulated to zero:

$$e_{vs} = C(s - s^*) \tag{8}$$

where matrix C, called combination matrix, allows to take into account more visual features than available degrees of freedom. For $C = L^+_{(s^*,z^*)}$, the task jacobian is equal to $J_{VS} = CL_{(s^*,z^*)}J_r = J_r$. The task is admissible as $det(J_r) = D_x \neq 0$.

By imposing an exponential decrease on e, a controller making e vanish is obtained in [Folio and Cadenat, 2008] :

$$\dot{q}_{vs} = -(CL_{(s,z)}J_r)^{-1}\lambda_{vs}C(s - s^*) \tag{9}$$

where λ_{vs} is a positive scalar or a positive definite matrix.

2) Obstacles avoidance: To perform a vision based navigation task in a cluttered environment, we have to deal with the problems of collision. In this section we first present the mobile base controller allowing to avoid obstacles, then the pan-platform one to perform the target tracking.

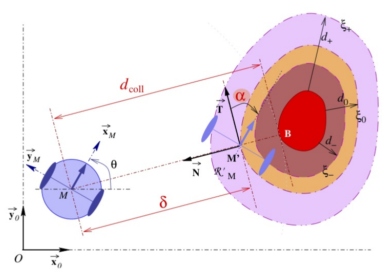


Fig. 4. Collision detection

The chosen strategy consists in making the vehicle avoid the obstacle by following the security envelope ξ_0 as shown in Fig. 4. With this method, only the mobile base (i.e υ and ω) is taken into account. To design the desired controller, the following task function has been defined [Folio and Cadenat, 2005]:

$$e_{mb} = (l - v_r t \quad \delta + k\alpha)^T \tag{10}$$

where l is the curvilinear abscissa of point M (Fig 4), k a positive gain to be fixed and v_r the desired linear velocity. The first component of this task function allows to regulate the linear velocity to v_r. The second component can be seen as a sliding variable whose regulation to zero makes both δ and α vanish (Fig 4).

Thanks to the task (10), only the mobile base is controlled. To keep the target in the camera field of view during the avoidance phase, we have to define a specific task function for the pan-platform :

$$e_{pp} = Y_0 \tag{11}$$

where Y_0 is the abscissa of the visual pattern gravity center. We can now obtain a task function $e_{oa} = (e_{mb} \; e_{pp})^T$ for the obstacle avoidance. Using classical visual servoing techniques, a controller making e_{oa} vanish is obtained :

$$\dot{q}_{oa} = J_{oa}^{-1}(-\lambda_{oa}e_{oa} - A_{oa}) \tag{12}$$

where λ_{oa} is a positive scalar, $A_{oa} = (\; -v_r \quad 0 \quad 0 \;)^T$ and:

$$J_{oa} = \begin{pmatrix} 1 & 0 & 0 \\ \sin(\alpha) - k\chi\cos(\alpha) & k & 0 \\ & L_{Y_0}J_r & \end{pmatrix} \tag{13}$$

We define $\chi = \frac{\frac{1}{R}}{1+\frac{\sigma}{R}\delta}$, with R the curvature radius of the obstacle and $\sigma = \{-1, 0, 1\}$ depending on the sense of the robot motion around the obstacle. Moreover, $L_{Y_0} = (-\frac{f}{z_0}, \frac{Y_0}{z_0}, f + \frac{Y_0^2}{f})$, and z_0 is the depth of the visual pattern gravity center. It should be noticed that the obstacle avoidance task is admissible as the task function jacobian J_{oa} is invertible:

$$det(J_{oa}) = -\kappa\left(\frac{C_x}{z_0} + \frac{C_y Y_0}{z_0} + 1 + Y_0^2\right) \neq 0 \tag{14}$$

3) The reorientation phase: The classical image based visual servoing presented in III-C.1 allows to compute the camera kinematic screw. Some of the approaches proposed in [Chaumette and Hutchinson, 2006] allow to decouple the camera velocities. Thus, if the camera trajectory appears to be correct, it is not necessary the case of the one of the mobile base. For example, as shown in figure 5, where $q_{init} = [\; 0 \quad \frac{-2\pi}{3} \quad \frac{5\pi}{6} \;]^T$, the system performs the task whereas this global behaviour is incongruous.

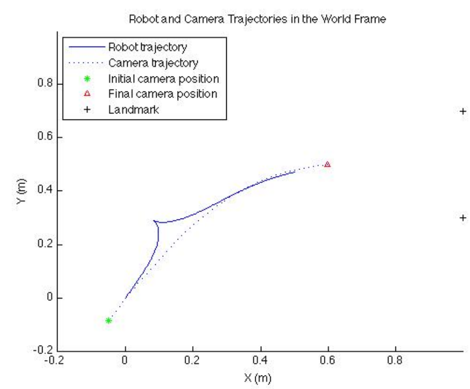


Fig. 5. Example of incongruous behaviour during a visual servoing

To overcome this problem we propose to define a controller allowing to turn toward the mobile base toward the next target without performing a reverse movement. Moreover, the camera has to be controlled to keep the target in its field of view. To this aim, we define the following task function :

$$e_{tr} = (l_{tr} - v_{tr}t \quad \theta_{tr} - \omega_{tr}t \quad Y_0)^T \tag{15}$$

where v_{tr} and ω_{tr} are the reference linear and angular velocities, l_{tr} is the curvilinear abscissa of point M and θ_{tr} the robot heading. v_{tr} is chosen > 0 to preserve the robot of a reverse movement while ω_{tr} is the same sign as $\vartheta(t_s)$, the pan-platform orientation in the robot frame at the switching time. Indeed, before the transition phase, the camera is controlled to preserve the target visibility. So, the pan-platform orientation at t_s informs us about the suitable mobile base heading. Controlling the mobile base to regulate to zero the camera orientation means directing the former towards the new target.

Making e_{tr} vanish implies that the robot linear and angular velocities are the right ones, while the camera is looking to the landmark. We propose to impose an exponential decrease:

$$\dot{e}_{tr} = -\lambda_{tr} e_{tr} \quad (16)$$

The expression of $\dot{e}$ is now required:

$$\dot{e}_{tr} = (v - v_{tr} \quad \omega - \omega_{tr} \quad \dot{Y}_0)^T = J_{tr}\dot{q}_{tr} + A_{tr} \quad (17)$$

where

$$J_{tr} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ & L_{Y_0}J_r & \end{pmatrix} \quad A_{tr} = \begin{pmatrix} -v_{tr} \\ -\omega_{tr} \\ 0 \end{pmatrix} \quad (18)$$

where L_{Y_0} is given in III-C.2. It shoud be noted that:

$$det(J_{tr}) = \frac{-f(D_x\cos(\vartheta) - C_x) + Y_0(D_x\sin(\vartheta) - C_y)}{z_0} - f - \frac{Y_0^2}{f} \neq 0 \quad (19)$$

Now, thanks to (15), (16) and (17), we obtain the controller to realize a transition:

$$\dot{q}_{tr} = J_{tr}^{-1}(-\lambda_{tr}e_{tr} - A_{tr}) \quad (20)$$

The transition phase is considered to be finished when $\vartheta = 0$. At this moment the mobile base is oriented toward the target.

D. Parameters estimation for visual servoing

Several parameters are needed to be able to compute the control inputs (9) : the current visual features s, the desired visual features s^* and the depth z_i of each point p_i which characterizes the landmark. In the following sections, the estimation process for each parameter is presented.

1) The current visual features: The controller (9) can only be used if the visual data are available. If they are not the task cannot be realized anymore. To remedy this critical situation, [Folio and Cadenat, 2008] has recently proposed to compute the visual data evolution for any $t \in [t_{k-1}, t_k]$, with $t_k = t_{k-1} + T_s$ and T_s the sampling period. To this aim, the values of $X_i(k), Y_i(k)$ and $z_i(k)$ are computed using $X_i(k-1), Y_i(k-1), z_i(k-1)$ and $\dot{q}(k-1)$. This method requires the initial values of $X_i(t_{occ}), Y_i(t_{occ})$ and $z_i(t_{occ})$, where t_{occ} correspond to the occlusion time. The two first ones can be easily obtained thanks to the last image before occlusion, while the depth has to be estimated.

This method is used to realize the visual servoing [D] or the obstacle avoidance [E] tasks using the estimated visual features.

2) The visual feature depth: To provide the necessary depths for III-D.1, the estimator proposed in [Durand Petiteville et al., 2010] is used. The estimation process is based on a predictor/corrector pair using a number m of previous images. In the proposed algorithm, a high number of data is used to improve the depth estimation. Thus, the estimation process is robust to the errors due to the image process and odometry.

3) The desired visual features: In [Durand Petiteville et al., 2010], a method is proposed to automatically compute the desired visual features. Thanks to the estimated depth of the visual data and the images, the landmark dimensions are computed. Then a desired camera position with respect to the landmark is defined. Thus, using the camera pinhole model, the desired visual features can be obtained.

IV. SIMULATIONS

We have simulated a long range navigation in a cluttered environment using the software Matlab^{TM}. The scene is composed of four occulting obstacles, a non-occulting one and a set of nine landmarks (see figure 6). The landmarks are made of four circles, whose centers C_k are extracted by an image processing. So, for a target T_j, $s_j = [X_1, Y_1, X_2, Y_2, X_3, Y_3, X_4, Y_4]^T$ where X_k and Y_k are the centers C_k coordinates. The topological map described in figure 1 is used to model the navigation environment. In this simulation the robot has to reach the target T_7 from its initial configuration $q_{init} = [1, 1, 1.81, 0]^T$. As T_7 cannot be seen from q_{init} because of an occulting obstacle, the set $T_P = [T_1, T_2, T_4, T_6, T_7]$ is computed using the Dijkstra's algorithm, as shown in figure 1. In order to be closer of the real conditions, a one pixel noise has been added to the visual data whereas a 3% error has been introduced in the odometry process. Finally the sampling time is $T_s = 0.1s$

As shown in figure 6, the long range navigation (about 20 meters) has been successfully performed. The robot safety is guaranteed and the different targets are sequenced until the goal is reached. It should be noticed that each task shown in the supervision algorithm has been realized at least one time. Indeed, in addition to the classical [B] and [C] tasks, the reorientation task [H] is executed when the considered target changes, unless an obstacle avoidance is performed. Thus, the robot behaviour presented in III-C.3 is not found during the navigation.

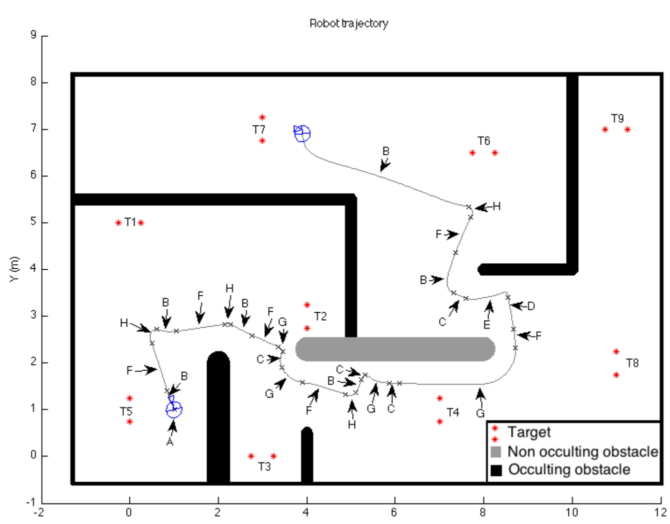


Fig. 6. The robot trajectory

The depth estimation allows to compute the reference visual features s_j^* (Fig. 7) during the tasks [A], [B], [C] and [H], and to estimate the visual features when no image is available (Fig. 8). Thus during tasks [D], [E], [F] and [G] the robot can perform the navigation wrt the current target without using the visual data obtained by the camera.

From a global point of view, the long range navigation is successfully and automatically performed using the unique topological map knowledge. Indeed at the end of the navigation, the camera has reached the desired final position $q_{final} = [3.9, 7, \pi]^T$. However, the trajectory made by the robot is not the shortest one. As the path is realized using only local and reactive techniques, the global trajectory is not improved. Moreover, it should be noticed that the system does not switch to the next landmark when it can be seen but only when it has been detected by the camera. The search task cannot be improved because the topological map does not contain any informations about the landmarks positions. Thus, it is impossible to privilege any direction of research for the next target.

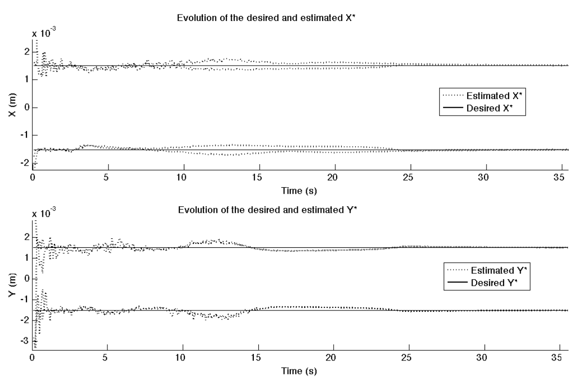


Fig. 7. Estimated and real reference visual data with respect to T_7

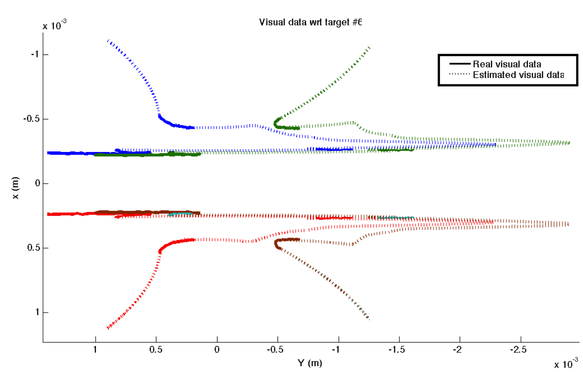


Fig. 8. The visual data with respect to T_6

V. CONCLUSION AND FUTURE WORKS

In this paper we have proposed a method allowing to perform a vision based long range navigation. The latter relies on a supervision algorithm organizing a set of elementary tasks coupled to a topological map. This map is made of a small number of landmarks, which is the only knowledge available about the environment. The environment is then represented using a reduced database contrary to the other approaches of the literature. The supervision algorithm allows to select the right elementary task at the right instant. Each of these tasks is performed thanks to a local controller, based either on real or estimated sensory data. In this way, it becomes possible to perform long range navigation mission using well-known reactive techniques.

We will validate the proposed approach using artificial landmarks on our robot Pekee II, developed by Wany Robotics. Finally, it will be also necessary to consider the topological map construction. The use of advanced pattern recognition and image processing techniques will allow to take into account a wider variety of landmarks. In this way, it will become possible to perform a vision based long range navigation in a natural indoor environment.

REFERENCES

[Blanc et al., 2005] Blanc, G., Mezouar, Y., and Martinet, P. (2005). Indoor navigation of a wheeled mobile robot along visual routes. In *International Conference on Robotics and Automation*, Barcelona, Spain.

[Bonin-Font et al., 2008] Bonin-Font, F., Ortiz, F., and Oliver, G. (2008). Visual navigation for mobile robots : a survey. *Journal of intelligent and robotic systems*, 53(3):263.

[Chaumette and Hutchinson, 2006] Chaumette and Hutchinson (2006). Visual servo control, part 1 : Basic approaches. *IEEE Robotics and Automation Magazine*, 13(4).

[Cherubini and Chaumette, 2010] Cherubini, A. and Chaumette, F. (2010). A redundancy-based approach to obstacle avoidance applied to mobile robot navigation. In *Proc. of IEEE Int. Conf. on Intelligent Robots and Systems*, Taipei, Taiwan.

[Durand Petiteville et al., 2010] Durand Petiteville, A., Courdesses, M., Cadenat, V., and Baillion, P. (2010). On-line estimation of the reference visual features. application to a vision based long range navigation task. In *IEEE/RSJ 2010 International Conference on Intelligent Robots and Systems*, Taipei, Taiwan.

[Espiau et al., 1992] Espiau, Chaumette, and Rives (1992). A new approach to visual servoing in robotics. *IEEE Trans. Robot. Automat.*, 8:313–326.

[Folio and Cadenat, 2008] Folio and Cadenat (2008). *Computer Vision*, chapter 4. Xiong Zhihui; IN-TECH.

[Folio and Cadenat, 2005] Folio, D. and Cadenat, V. (2005). A controller to avoid both occlusions and obstacles during a vision-based navigation task in a cluttered environment. In *European Control Conference (ECC05)*, page 38983903, Seville, Espagne.

[Gaspar et al., 2000] Gaspar, J., Winters, N., and Santos-Victor, J. (2000). Vision-based navigation and environmental representations with an omnidirectional camera. *IEEE transactions on robotics and automation*, 6(6):890–898.

[Kralnìk and Přeučil, 2008] Kralnìk, T. and Přeučil, L. (2008). *A simple visual navigation system with convergence property*. H. Bruyninckx et al. (Eds.).

[Mansard and Chaumette, 2007] Mansard, N. and Chaumette, F. (2007). Task sequencing for sensor-based control. *IEEE Trans. on Robotics*, 23(1):60–72.

[Pissard-Gibollet and Rives, 1995] Pissard-Gibollet and Rives (1995). Applying visual servoing techniques to control a mobile handeye system. In *IEEE Int., Conf. on Robotics and Automation*, Nagoya, Japan.

[Samson et al., 1991] Samson, Borgne, and Espiau (1991). Robot control : The task function approach. *Oxford science publications*.

[Santos-Victor and Sandini, 1995] Santos-Victor, J. and Sandini, G. (1995). Visual-based obstacle detection: a purposive approach using the normal flow. In *International Conference on Intelligent Autonomous Systems*, Karlsruhe, Germany.

[Souères and Cadenat, 2003] Souères, P. and Cadenat, V. (2003). Dynamical sequence of multi-sensor based tasks for mobile robots navigation. In *SYROCO*, Wroclaw, Poland.

[Vassalo et al., 2000] Vassalo, R., Schneebeli, H., and Santos-Victor, J. (2000). Visual servoing and appearance for navigation. *Robotics and autonomous systems*.