HAL
open science

Visual servoing based mobile robot navigation able to deal with complete target loss

Wenhao Fu, Hicham Hadj-Abdelkader, Etienne Colle

To cite this version:

Wenhao Fu, Hicham Hadj-Abdelkader, Etienne Colle. Visual servoing based mobile robot navigation able to deal with complete target loss. 18th International Conference on Methods and Models in Automation and Robotics (MMAR 2013), Aug 2013, Międzyzdroje, Poland. (to appear). hal-00845541

HAL Id: hal-00845541

https://hal.science/hal-00845541v1

Submitted on 19 Jul 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Visual servoing based mobile robot navigation able to deal with complete target loss

Wenhao Fu, Hicham Hadj-Abdelkader and Etienne Colle
IBISC Lab, IRA2 Team
University of Evry-Val-d'Essonne
40, rue du Pelvoux, 91020 Courcouronnes - France
Email: FirstName.LastName@ibisc.univ-evry.fr

***Abstract*— This paper combines the reactive collision avoidance methods with image-based visual servoing control for mobile robot navigation in an indoor environment. The proposed strategy allows the mobile robot to reach a desired position, described by a natural visual target, among unknown obstacles. While the robot avoids the obstacles, the camera could lose its target, which makes visual servoing fail. We propose in this paper a strategy to deal with the loss of visual features by taking advantage of the odometric data sensing. Obstacles are detected by the laser range finder and their boundaries are modeled using B-spline curves. We validate our strategy in a real experiment for an indoor mobile robot navigation in presence of obstacles.**

I. INTRODUCTION

Most of recent mobile robot navigation approaches are based on visual servoing techniques. Visual servoing consists in directly using vision in the robot control loop [1] [2]. Vision based mobile robot navigation benefits from the efficacy of visual servoing in unknown and dynamic environments. Compared with model-based mobile robot navigation, image-based visual servoing operates in the image space without any model of the environment. Moreover, it can accomplish various robot tasks due to the vast information acquired from the camera. This paper uses a visual servoing based control to navigate a mobile robot in an indoor environment.

Visual servoing approaches were initially applied to 6 DOF manipulators. They have been extended to control mobile robots since 1990's. In the context of mobile robot navigation, in particular nonholonomic mobile robot navigation, three critical issues should be taken into account: (1) positioning tasks with respect to specific geometric features, while taking nonholonomic properties into account; (2) combining image-based visual servoing with other tasks beyond image space, notably collision avoidance; (3) visibility of visual feature during servoing, particularly when considering a conventional perspective camera system fixed on a mobile robot.

For the first issue, the nonholonomic property that the DOF of the control input is less than that of the robot configuration causes that a pure state feedback stabilization, namely image-based visual servoing, around a given reference configuration is impossible. To overcome this problem, the pan angle of camera is added to the input with more control DOFs, and teaching and playback is realized with a sequence of reference images [3]. This method can converge nonholonomic vehicles to a desired state. For our application, we aim to use visual servoing with a topological map to solve the large scale navigation problem. We do not aim to dedicate to positioning for this problem. Instead, we focus on navigating the robot towards a given reference image key while avoiding collision in an unknown environment using a perspective camera fixed on a mobile robot. When the robot avoids obstacles, the target can be lost in the field of view of the camera, which is related to the second and the third issues.

For the second issue, a hybrid method was introduced to combine visual servoing control and collision avoidance control [4]. We use the hybrid framework and adopt a path following method to deal with collision avoidance while we extend the obstacle presentation with B-spline, which can smooth the reference path during obstacle avoidance.

For the third issue, ensuring visibility is still challenging. The visibility requires that a minimum number of visual features must remain in the camera field of view (FOV) during the servoing [5]. If these features are lost, visual servoing will fail. Feature loss can be caused by occlusion or by the control generated from other tasks such as obstacle avoidance. Earlier methods to deal with this problem are dedicated to avoiding occlusions and loss by keeping all visual features or the whole object visible during visual servoing [6], [7], [8], [9]. These methods are dedicated to manipulator arms with 6 DOFs, and they benefit from redundancy. In case of mobile robot control, Chirubini et al. [10] propose a redundancy-based approach by adding an extra DOF for the camera. This end should be able to keep the target in the field of view during obstacle avoidance. Folio et al. propose in [11] a method to take into account both visibility and collision. Contrary to preserving visibility of all the visual features, a first approach [5] allows the changes in visibility of the visual features during the servoing. However, this method is limited to partial changes of visibility. And if the appearing visual features are less than a required minimum number or totally loss, the method will also fail. In order to handle the total loss of visual features, a method is introduced in [12]. This method benefits from a visual data estimation algorithm. It uses the previous visual measurements and the control inputs to estimate the current visual features when they are lost. However, the proposed method is subjected to estimation errors which may increase quickly. In this paper, we show how the target can be estimated using internal odometry.

The paper is organized as follows. Section II presents the system modeling. Section III describes a collision avoidance method based on a B-spline representation. Section IV introduces an image-based control. Section V proposes a strategy considering target loss in the image-based control. Section VI presents experimental results.

II. System modeling

We consider a two-wheel differential drive mobile robot equipped with a fixed pinhole camera, a 2-D laser range finder and an odometry sensor.

We consider the motion of the mobile robot in an inertial frame $\mathcal{F}_W$ fixed in the world. Let $\mathcal{F}_R$ be the frame attached to the mobile robot. The origin R will be termed reference point of the mobile robot. The x-axis is aligned with the heading of the robot. The z-axis is perpendicular to the motion plane.

We suppose that the mobile robot can only move in a planar workspace. The configuration of the mobile robot is described by $[x\ y\ \theta]^\top$, where x and y are the position coordinates of R with respect to X and Y axes, of $\mathcal{F}_W$, respectively. The angular deviation θ is defined by the positive counterclockwise angle between X axes of $\mathcal{F}_W$ and $\mathcal{F}_R$ (see Figure 1(a)). The robot motion is controlled by the linear velocity υ and the angular velocity ω. The non-holonomic kinematic model of the robot is given as:

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \upsilon \\ \omega \end{pmatrix} \tag{1}$$

We consider a perspective camera fixed on the mobile robot. Let $\mathcal{F}_C$ be the frame attached to the camera. The origin C will be termed optical center, and the z-axis is the optical axis. A pinhole camera model is considered as shown in Figure 1(b). Let $\mathcal{P}$ be a 3-D point with homogeneous coordinates $\mathbf{X} = [X\ Y\ Z\ 1]^\top$ with respect to $\mathcal{F}_C$. The point $\mathcal{P}$ is projected in the normalized image plane to a point of homogeneous coordinates $\mathbf{x} = [x\ y\ 1]^\top$. The image point $\mathbf{m} = [u, v, 1]$ is the homogeneous coordinates given in pixel as $\mathbf{m} = \mathbf{K}\mathbf{x}$, where $\mathbf{K}$ is the camera intrinsic matrix, obtained after calibration.

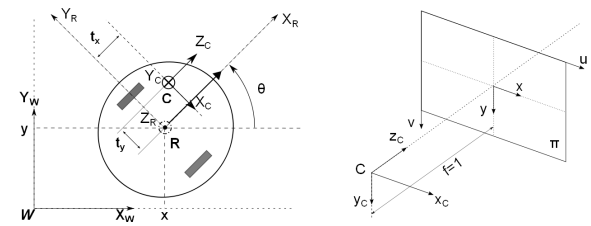
Let $\mathbf{v}^c = [\upsilon_x\ \upsilon_y\ \upsilon_z\ \omega_x\ \omega_y\ \omega_z]^\top$ be the camera kinematic screw, and $\dot{\mathbf{q}} = [\upsilon\ \omega]^\top$ be the control input of the mobile robot. They are related by the jacobian matrix $\mathbf{J}$: $\mathbf{v}^c = \mathbf{J}\,\dot{\mathbf{q}}$ where:

$$\mathbf{J} = \begin{bmatrix} \mathbf{J}_\upsilon & \mathbf{J}_\omega \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ -t_x & 0 & -t_y & 0 & -1 & 0 \end{bmatrix}^\top \tag{2}$$

with $\mathbf{J}_\upsilon$ and $\mathbf{J}_\omega$ are the first and the second columns of $\mathbf{J}$ respectively.

III. Reactive collision avoidance

Reactive collision avoidance algorithms compute collision-free motion control from the sensing data within each control sampling interval. They are suitable for unknown and dynamic environments which are unpredictable. A reactive collision avoidance method is basically composed of three steps: (1) obstacle detection; (2) obstacle modeling; (3) robot motion generation to avoid obstacles.



(a) Model of non-holonomic wheeled mobile robot with camera

(b) Pinhole camera model

Fig. 1. System modeling

In [13], a framework based on a path following method combined with vision-based tracking task to deal with obstacle avoidance was proposed. Inspired by this framework, we extend the method by generating a smooth curve representing obstacle boundary. Then path following method is used to drive the robot avoid the obstacle by keeping a safe distance with respect to the interpolated curve.

A. Obstacle detection

In order to select the obstacle to be avoided, the raw data provided by the laser range finder should be first handled by creating a set of segments. Each segment represents a full or a part of obstacle. We then select the segments which can potentially block the motion of the robot towards to the goal. The selection is based on a predictor which simulates a short term of motion from the goal-directedness controller.

We use a simple and efficient approach for processing the laser data. We segment laser scan using Successive Edge Following (SEF) algorithms, according to the distance between each two adjacent data of the scan. The groups are discarded if they consist of too few points, which are noises. Because of the uncertainty of the sensor, noises may break one obstacle into to two or more segments. We associate thus the adjacent segments into one. A sample result from the real experiment of clustering the laser data to segments which can approximately represent the obstacles nearby the robot, is shown in Figure 2. Different segments are marked in different colors.

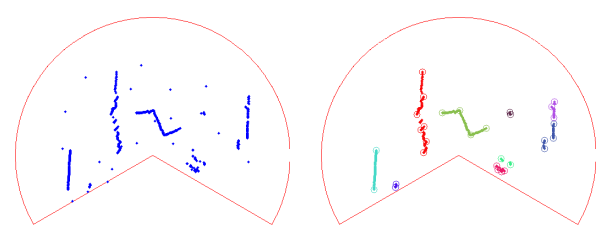


Fig. 2. Obstacles segmentation: before and after segmentation in left and right respectively.

After segmentation, we select the obstacle by predicting a small motion step of the mobile robot using the controller provided by the goal-directedness task. Figure 4 shows a real situation of a selected obstacle (shown in red color) in the laser range finder space.

B. Obstacle representation

Since our avoidance strategy is based on following the obstacle boundary, a model of the boundary is required. Usually, obstacle are modeled by line segments, circles or boxes. However, these simple models need a global knowledge of the obstacle. Whereas, in the case of using ultrasonic or 2D laser range finder, usually a part of the obstacle is sensed. Consequently, box or circle models are not adapted for such situation. Line based models have no continuity at the connections which is not suitable for the robot motion. In this paper, we use a cubic B-spline to approximate the convexity of the obstacle boundary. The main advantage is that it can have arbitrary order of continuity, which is suitable for mobile robot control.

The basic idea behind B-spline based obstacle boundary representation includes two steps. Firstly, we extract a convex polygonal chain of the laser data belong to the obstacle. Then, we interpolate the vertices of the extracted convex polygonal chain by a cubic B-spline. As in [14], we use the distance with symbol in the Iterative-End-Point-Fit algorithm to select the the convex points. The process is illustrated in Figure 3. A line is constructed connecting the first and the last points of the segment. Then we calculate the distances of other points to this line. Each distance has a sign according to the location of the point with respect to the line segment. The point with a positive maximum distance is selected as a convex point if its value is bigger than a given threshold. Then the segment is divided into two subsets at the selected point. The process is recursively repeated to find the convex points in the subsets until no convex point can be found. The process outputs a list of convex points as the vertices of the convex polygonal chain. The vertices of the convex polygonal chain are interpolated by a cubic B-spline curve as can be shown with green color in Figure 4.

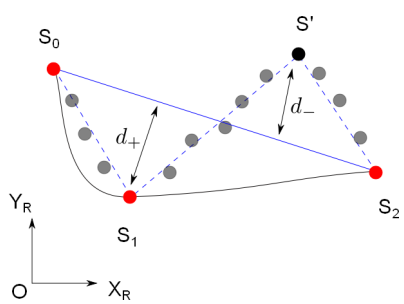


Fig. 3. Convex point selection and B-spline fitting

C. Motion control

Our obstacle avoidance is based on the approach proposed in [13]. The idea is to use the path following approach [15] to control the robot moving around the obstacle. The path following aims to minimize an error related to the current and desired configuration of the mobile robot.

To define the error vector, two envelops ξ_0 and ξ_+ are defined. They surround the obstacle at distance $d_0 < d_+$

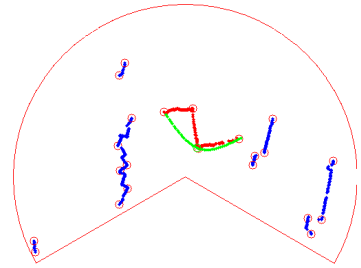


Fig. 4. Obstacle selecting and boundary fitting

representing the risk of collision. ξ_0 is the reference path to follow. ξ_+ defines a zone, inside which the obstacle is considered. We suppose that the distance between each two obstacles is bigger than $2d_+$ which ensures that the robot deals with one obstacle at each time. A new reference frame $\mathcal{F}_{R_0}$ of origin R_0 is generated on the boundary ξ_0. The origin point R_0 is the orthogonal projection of the origin R of the robot frame on ξ_0. The point R_0 is the closest point between R and ξ_0, and the closest distance is noted as δ. The closest distance between R and the obstacle boundary is noted as d_{coll}.

Let the error vector $\mathbf{e} = (\delta, \alpha, \chi)$ where $\delta = d_{coll} - d_0$ is the lateral deviation of the mobile robot with respect to the reference path ξ_0, $\alpha \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ is the angular deviation between the robot direction and the tangent of the path at R_0, and χ is the curvature of the path ξ_0 at R_0.

By using the task function formalism [16], the control law can be designed by setting a non-zero linear velocity and an angular velocity as:

$$\omega_{coll} = -v(k\delta + \alpha + 2k\sin\alpha - \chi\cos\alpha) \qquad k > 0 \qquad (3)$$

This control law aims to regulate the error vector $\mathbf{e}$ to zero, which control the mobile robot to move around obstacle.

IV. Image-based visual servoing for mobile robot control

In order to move the mobile robot from an initial to a desired configurations in an indoor environment, visual servoing can be used to design the control law which is able to handle this aim. In this work, an image-based visual servoing is used to design the control law. To deal with this, the visual target corresponding to the desired position of the mobile robot, should be detected at the initial position and tracked during servoing. We use a natural planar target such as a poster or a painting. We suppose that the target is initially visible by the camera. If that condition is not satisfied, a first task like looking for the target by moving the mobile robot (turning around) can be defined. When the target is in the camera field of view, its can be recognized by matching detected features. In this work, we use SIFT [17] detector and descriptor to extract and match local features from the reference image and the current image. Furthermore, the robot motion constraint is used to improve matching by rejecting outliers through RANSAC. When the current image is matched with the reference one, a homography relationship can be computed. This homography

is then used to initialize an efficient template-based visual tracking, namely, efficient second-order minimization method (ESM) [18]. This tracking method computes the parameterized homography using a minimization method, which efficiently approximates the second-order of the cost function without computing Hessian matrices.

Four corner points of the tracked planar target are selected as the visual features for the image-based visual servoing control scheme. We note the visual features of the current image as $\mathbf{s} = [\mathbf{s}_0\ \mathbf{s}_1\ \mathbf{s}_2\ \mathbf{s}_3]^\top$, where $\mathbf{s}_i = [x_i\ y_i]^\top$ is the coordinate vector of each point expressed in the normalized image plane. If we note the visual features vector of the desired image as $\mathbf{s}^*$, there is a single homography matrix $\mathbf{H}$ such that $\mathbf{s} \propto \mathbf{H}\mathbf{s}^*$.

The relationship between the velocity of visual feature $\dot{\mathbf{s}}$ and the kinematic screw of the camera $\mathbf{v}^c$ is given by:

$$\dot{\mathbf{s}} = \mathbf{L}_\mathbf{s}\mathbf{v}^c \tag{4}$$

where $\mathbf{L}_\mathbf{s}$ is the interaction matrix related to $\mathbf{s}$. The expression of $\mathbf{L}_{\mathbf{s}_i}$ for a single normalized image point $\mathbf{s}_i$ is:

$$\mathbf{L}_{\mathbf{s}_i} = \begin{bmatrix} -/Z_i & 0 & x_i/Z_i & x_i y_i & -(1+x_i^2) & y_i \\ 0 & -/Z_i & y_i/Z_i & 1+y_i^2 & -x_i y_i & -x_i \end{bmatrix} \tag{5}$$

where the value Z_i is the depth of the 3D point with respect to the camera frame.

By considering the jacobian matrix (2) linking the camera velocity vector to the mobile robot one, we obtain:

$$\dot{\mathbf{s}} = \mathbf{L}_{\mathbf{s},v}\, v + \mathbf{L}_{\mathbf{s},\omega}\, \omega \tag{6}$$

where $\mathbf{L}_{\mathbf{s},v} = \mathbf{L}_\mathbf{s}\mathbf{J}_v$ and $\mathbf{L}_{\mathbf{s},\omega} = \mathbf{L}_\mathbf{s}\mathbf{J}_\omega$.

Visual servoing control consists in minimizing the error $\mathbf{e} = \mathbf{s} - \mathbf{s}^*$. By considering the desired features constants, an exponential decoupled decrease of the error can be obtained using the following control law:

$$\omega_{nav} = -\mathbf{L}^+_{\mathbf{s},w}(\lambda\mathbf{e} + \mathbf{L}_{\mathbf{s},v}\, v) \tag{7}$$

where $\mathbf{L}^+_{\mathbf{s},w}$ is the Moore-Penrose pseudo-inverse of $\mathbf{L}_{\mathbf{s},w}$. Note that the linear velocity v in (7) can be assigned to a constant value or computed from a designed linear velocity profile.

V. Image-based mobile robot navigation in presence of obstacles

During mobile robot navigation to the goal thanks to the visual servoing control, obstacles present in the trajectory of the robot should be detected and avoided as described in the previous section. However, visual target can be lost during collision avoidance task and thus visual servoing will fail. We propose in this work a strategy to bring visual servoing task feasible when the obstacle avoidance is done. Indeed, the system should be able to decide when the obstacle is avoided in order to switch from the collision avoidance task to the facing the visual target task. Then, the image-based navigation task can be successfully start again. Otherwise, the system will be reset as at the beginning of the image-based navigation task.

A. Collision avoidance deactivation

When the mobile robot detect an obstacle, the image-based control is stopped to stand aside the collision avoidance control as given in (3). However, the presented strategy for collision avoidance which is based on moving around the obstacle boundary, should be deactivated when the mobile robot is able to reach the desired position without collision. With this aim in mind, the mobile robot retains the direction of the target when the obstacle avoidance is activated. As shown in Figure 5, a linear path can be defined using the retained direction. A basic control law $\omega_{virtual}$ (using [15]) for linear path following can thus be designed during collision avoidance task since the motion is estimated between the current and collision avoidance activation times, thanks to the odometric data. This control law aims to correct the lateral and the angular deviations (d_l, α_l) with respect to the linear path. In our case, this control law is virtual since it is never applied to control the mobile robot, and only used as a condition to decide when the collision avoidance task can be deactivated. Indeed, the condition is based on a comparison between the angular velocities of both controllers (virtual and collision avoidance): if $|\omega_{coll}| > |\omega_{virtual}|$ and $sign(\omega_{coll}) = sign(\omega_{virtual})$, then deactivate collision avoidance; otherwise, continue collision avoidance.

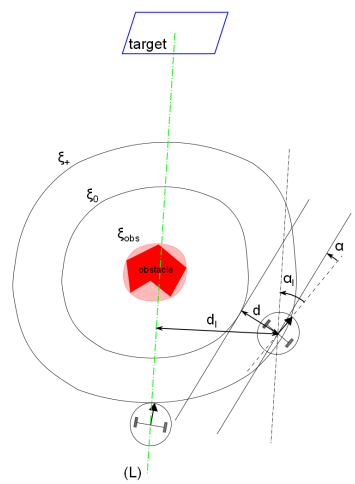


Fig. 5. Collision avoidance

B. Image target estimation during collision avoidance

The idea is to estimate the homography relationship between the image at the beginning of obstacle avoidance and the current image. Since the target is no more visible in the current image, the homography is computed using the recovered camera motion thanks to the proprioceptive sensor (odometry) of the mobile robot.

The homography matrix $\mathbf{H}$ can be decomposed as:

$$\mathbf{H} = \mathbf{R} + \frac{\mathbf{t}}{d^*}\mathbf{n}^{*\top} \tag{8}$$

where $\mathbf{R}$ and $\mathbf{t}$ are respectively the rotation matrix and the translation vector between the current and desired camera

frames, $\mathbf{n}^*$ and d^* are respectively the normal vector to the planar target and its distance with respect to the camera frame when the obstacle avoidance is activated.

A straightforward way to compute $\mathbf{H}$ is to estimate all the elements in (8). By supposing that the obstacle avoidance will be done in a short path, the motion estimation, based on the odometry provided by the proprioceptive sensor of the mobile robot, has a low drift and can thus be exploited to compute the homography relationship with an adequate precision. The odometry based motion estimation is started at the beginning of the obstacle avoidance task. At this time, the normal vector $\mathbf{n}^*$ and the distance d^* should be estimated to recover the homography relationship between the images at the avoidance beginning and end. The estimation problem is compared to a stereo camera system where the displacement between two images is provided by the odometry and then the 3D structure can be built using for example [19].

The idea behind homography estimation is illustrated in figure 6. The 3D structure is estimated at ${}^wC_{lost}$ using the current image and the farthest image having the target in the field of view (before lose the target), during the obstacle avoidance. After the obstacle avoidance task, the recovered homography is used to represent the target at ${}^wC_{find}$ where it will probably be outside the image plane. Whatever, the deviation $\Delta\theta$ to bring the target inside the image plane can be estimated. A proportional control $\omega = \lambda\Delta\theta$ can thus be used to take the target back in the field of view. Then, the target tracking can be initialized and the visual servoing based goal-directedness task can be done successfully.

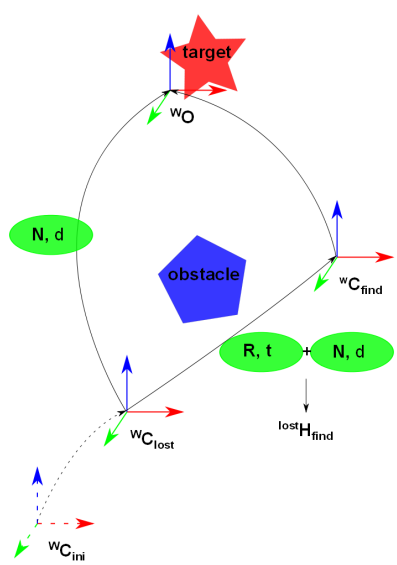


Fig. 6. Homography estimation

VI. Experimental results

We have accomplished the above methods on our mobile robot platform named Lina (see Figure 7). Lina mobile robot is equipped with various exteroceptive and proprioceptive sensors such as odometry, laser range finder, two ACTi IP cameras of resolution of 640×480 where only one camera (left camera) is used in this experiment. All data processing is done on a deported PC. Controls and data sensing are respectively sent and received to the robot platform through a local network (wired or wireless). The mobile robot is driven towards a goal represented in the image space by a natural image (a poster of size 55.8×73.5 cm). The four corner points of the target image are selected as visual features to build the visual servoing control. We use box-like objects as obstacles. They are placed between the initial and desired positions of the mobile robot.

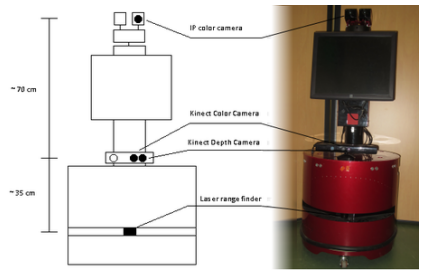


Fig. 7. Lina mobile robot

Figure 8 shows the the navigation steps where the different positions are given in red circles and numbers (1 to 5). The trajectory of the mobile robot is drawn in blue color. Obstacles detected by the laser range finder are given in black color. We suppose that the target is visible by the camera at the starting position. First, the target is detected by matching the current image with the reference one (acquired at the desired position). Then, the template based tracking using the ESM tracking technique, is initialized (at position 1). The mobile robot starts to reach the goal trough the visual servoing control scheme. During servoing, collision avoidance is considered in the mobile robot control when obstacles are detected in the path towards the goal (at position 2). During the obstacle avoidance (from position 2 to 3), the mobile robot position is recovered using the odometric data, and the reference plane parameters (the normal vector of the reference plane and its distance with respect to the camera frame) are estimated. When the obstacle is totally avoided (at position 3), the mobile robot is oriented towards the target (position 4) and then the visual servoing is restarted to reach the goal (position 5).

Figure 9 shows images and laser data at different steps (1 to 5) as described in figure 8. A video of the real experiment is available at `http://aramis.iup.univ-evry.fr:8080/~hadj-abdelkader/Videos/Wenhao/video_MMAR2013.wmv`.

VII. Conclusions

A solution for mobile robot navigation in indoor environments has been presented in this paper. Vision based navigation and laser range finder based obstacle avoidance have been used to reach a desired position of the mobile robot safely (without obstacle collision). A B-spline based model of obstacles and their segmentation have been proposed to

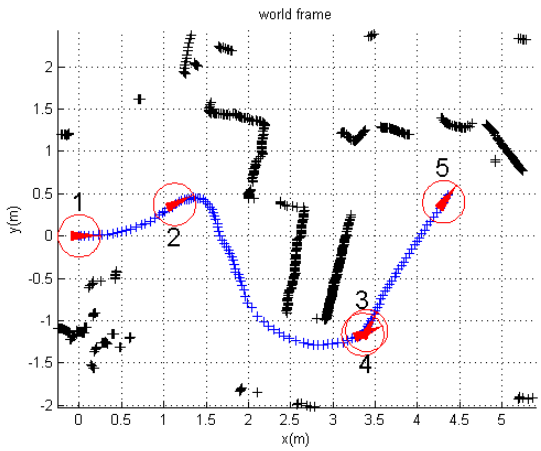


Fig. 8. Vision-based navigation

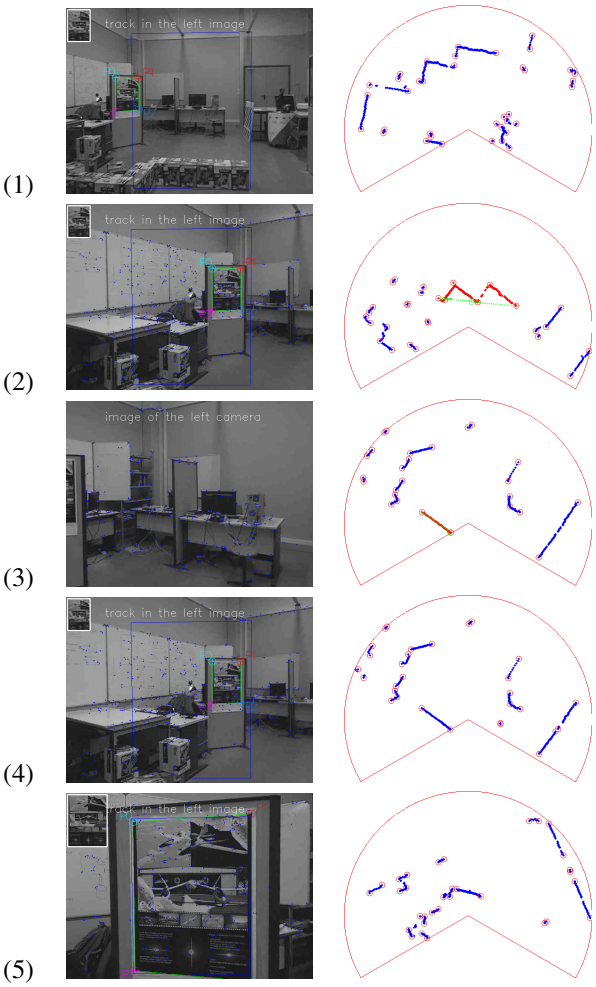


Fig. 9. Sensor data

ensure a smooths motion of the mobile robot during obstacle avoidance. Visual target loss while obstacle avoidance, has been solved by exploiting the odometry based pose estimation and the homography relationship. In the future works, stereo vision will be exploited to recover the target loss and complex obstacles detection such as tables and chairs.

References

[1] S. Hutchinson, G. Hager, and P. Corke, "A tutorial on visual servo control," *Robotics and Automation, IEEE Transactions on*, vol. 12, no. 5, pp. 651–670, 1996.

[2] F. Chaumette and S. Hutchinson, "Visual servo control, part ii: Advanced approaches," *IEEE Robotics and Automation Magazine*, vol. 14, no. 1, pp. 109–118, March 2007.

[3] Y. Masutani, M. Mikawa, N. Maru, and F. Miyazaki, "Visual servoing for non-holonomic mobile robots," in *Intelligent Robots and Systems '94. 'Advanced Robotic Systems and the Real World', IROS '94. Proceedings of the IEEE/RSJ/GI International Conference on*, vol. 2, Sep, pp. 1133–1140 vol.2.

[4] V. C. Ricardo, R. Swain, and M. Devy, "A controller to perform a visually guided tracking task in a cluttered environment," in *Department of Computer Science at Yale University. Since*, 1999, pp. 775–780.

[5] N. Garcia-Aracil, E. Malis, R. Aracil-Santonja, and C. Perez-Vidal, "Continuous visual servoing despite the changes of visibility in image features," *Robotics, IEEE Transactions on*, vol. 21, no. 6, pp. 1214–1220, 2005.

[6] S. Benhimane and E. Malis, "Vision-based control with respect to planar and non-planar objects using a zooming camera," in *IEEE International Conference on Advanced Robotics*, 2003, pp. 991–996.

[7] G. Chesi, K. Hashimoto, D. Prattichizzo, and A. Vicino, "Keeping features in the camera's field of view: a visual servoing strategy," in *15th Int. Symp. on Mathematical Theory of Networks and Systems*, August 2002.

[8] Y. Mezouar and F. Chaumette, "Optimal camera trajectory with image-based control," *I. J. Robotic Res.*, vol. 22, no. 10-11, pp. 781–804, 2003.

[9] ——, "Path planning for robust image-based control," *Robotics and Automation, IEEE Transactions on*, vol. 18, no. 4, pp. 534–549, 2002.

[10] A. Cherubini, F. Chaumette, and G. Oriolo, "A position-based visual servoing scheme for following paths with nonholonomic mobile robots," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS 2008*, Nice, France, September 2008, pp. 1648–1654.

[11] D. Folio and V. Cadenat, "A controller to avoid both occlusions and obstacles during a vision-based navigation task in a cluttered environment," in *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC '05. 44th IEEE Conference on*, Dec., pp. 3898–3903.

[12] ——, "A sensor-based controller able to treat total image loss and to guarantee non-collision during a vision-based navigation task," in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, Sept., pp. 3052–3057.

[13] V. CADENAT, P. SOUERES, and M. COURDESSES, "An hybrid control for avoiding abstacles during a vision-based tracking task," in *European Control Conference (ECC)*, 1999.

[14] J. Han, "Detection of convex and concave discontinuous points in a plane curve," in *Computer Vision, 1990. Proceedings, Third International Conference on*, 1990, pp. 71–74.

[15] C. Samson, "Path following and time-varying feedback stabilization of a wheeled mobile robot," in *Int. Conf. ICARCV'92*, 1992.

[16] C. Samson, M. Leborgne, and B. Espiau, "Robot control. the task-function approach," in *Oxford Engineering Science Series, vol. 22*. Oxford University Press, 1991.

[17] D. Lowe, "Object recognition from local scale-invariant features," in *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, vol. 2, pp. 1150–1157.

[18] S. Benhimane and E. Malis, "Homography-based 2d visual tracking and servoing," *I. J. Robotic Res.*, vol. 26, no. 7, pp. 661–676, 2007.

[19] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2004.