Proceedings of the 2006 IEEE
International Conference on Robotics and Biomimetics
December 17 - 20, 2006, Kunming, China

A redundancy-based scheme to perform safe vision-based tasks amidst obstacles

David FOLIO[†] and Viviane CADENAT
LAAS/CNRS
7, Avenue du Colonel Roche, 31077 Toulouse Cedex 4, France
{*dfolio, cadenat*}@laas.fr

***Abstract*— This paper presents a redundancy-based scheme allowing to avoid both occlusions and obstacles for a mobile robot performing a vision-based task in a cluttered environment. We consider the model of a cart-like robot equipped with ultrasonic sensors and a camera mounted on a pan-platform. The proposed method relies on the continuous switch between several controllers depending on the environment. Experimental results validating this approach are given at the end of the paper.**

***Index Terms*— Visual servoing, obstacle, occlusion avoidance.**

I. Introduction

Visual servoing techniques aim at controlling the robot motion using visual features provided by a camera mounted on the robot or fixed to the environment [1] [2]. Different approaches allow to design such control laws. Among them, the task function formalism [3] provides a general framework for designing sensor-based control laws. Indeed, this formalism can be applied to manipulators [4] as well as to nonholonomic mobile robots provided that, in this case, the camera is able to move independantly from the base [5]. However, the visual servoing techniques mentioned above require that the image features remain always in the camera field of view and that they are never occluded during the entire execution of the task. Most of the works which address this kind of problems are dedicated to manipulator arms. For example, in [6], the authors propose a method allowing to avoid self-occlusions and preserve visibility by path planning in the image for such robots. In [7], Marchand et al. benefit from manipulator arm redundancy to perform a vision-based task while avoiding occlusions, visual features loss and obstacles. In [8], the authors deal with the problem of robust 3D model-based tracking and presents an algorithm which is shown to be robust to occlusions, changes in illumination and miss-tracking. Finally, in [9], Wunsch et al. propose a model-based method allowing a robot to visually track 3D objects while occlusions are continuously predicted.

This paper focus on the sensor-based navigation of a nonholonomic mobile robot, equipped with an ultrasonic sensor belt and a camera mounted on a pan-platform, performing a nominal vision-based task in a cluttered environment. To realize such a task, it is necessary not only to preserve the image features visibility but also to prevent the mobile base from colliding with the obstacles. The issue is then quite different from the manipulator arms case. The proposed method is in the sequel of previous works [10] [11] [12] where different obstacle avoidance techniques and classical visual servoing control were merged at the control level to perform a vision-based navigation task in a cluttered environment. However, as these works were restricted to the case where occlusions could not occur, we have proposed in [13] a first extension allowing to take into account this phenomenon. In this last approach, we have chosen to split the global navigation task into two different subtasks to be sequenced : the nominal vision-based task and the obstacle and occlusion avoidance task. Therefore, in this work, obstacle bypassing and occlusion avoidance are realized *simultaneously* thanks to the robot redundancy. However, the main drawback of this modelling choice is that noncollision is only guaranteed *at best*, which means that, in some cases, the robot may move very close to the obstacles or even collide with them. As this solution is not completely satisfying, we propose in this article another method allowing to improve the robot safety.

Section II details the modelling step and the problem statement while section III is devoted to the design of the different controllers and the description of our control strategy. Finally, the last section shows the obtained experimental results.

II. Modelling and problem statement

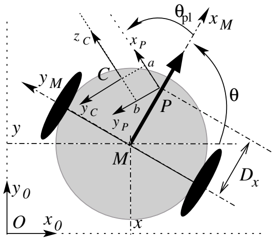

Fig. 1. The mobile robot with pan-platform

We consider the model of a cart-like robot with a CCD camera mounted on a pan-platform. The system kinematics is deduced from the whole hand-eye modelling given in [5] :

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\theta}_{pl} \end{pmatrix} = \begin{pmatrix} \cos\theta & 0 & 0 \\ \sin\theta & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} v \\ \omega \\ \varpi \end{pmatrix} \tag{1}$$

(x, y) are the coordinates of the robot reference point M with respect to the world frame $\mathcal{F}_O$. θ and θ_{pl} are

[†]PhD. thesis student supported by the European Social Fund.

respectively the direction of the vehicle and the pan-platform with respect to the x_M-axis. P is the pan-platform center of rotation, D_x the distance between M and P. We consider the successive frames: $\mathcal{F}_M\ (M, x_M, y_M, z_M)$ linked to the robot, $\mathcal{F}_P\ (P, x_P, y_P, z_P)$ attached to the pan-platform, and $\mathcal{F}_C\ (C, x_C\ , y_C\ , z_C)$ linked to the camera. The transformation between $\mathcal{F}_P$ and $\mathcal{F}_C$ is deduced from a hand-eye calibration method. The control input is defined by the vector $\dot{q} = (v, \omega, \varpi)^T$, where v and ω are the cart linear and angular velocities, and ϖ is the pan-platform angular velocity with respect to $\mathcal{F}_M$. Let $\mathcal{T}^c$ be the kinematic screw representing the translational and rotational velocity of $\mathcal{F}_C$ with respect to $\mathcal{F}_O$, expressed in $\mathcal{F}_C$. The kinematic screw is related to the joint velocity vector by the robot jacobian $\mathcal{J}$: $\mathcal{T}^c = \mathcal{J}\dot{q}$. As the camera is constrained to move horizontally it is sufficient to consider a reduced kinematic screw $\mathcal{T}^c_{\text{red}} = (V_{y_c}, V_{z_c}, \Omega_{x_c})^T$, and a reduced jacobian matrix $\mathcal{J}_{\text{red}}$ as follows:

$$\mathcal{T}^c_{\text{red}} = \begin{pmatrix} -\sin(\theta_{pl}) & D_x\cos(\theta_{pl}) + a & a \\ \cos(\theta_{pl}) & D_x\sin(\theta_{pl}) - b & -b \\ 0 & -1 & -1 \end{pmatrix} \begin{pmatrix} v \\ \omega \\ \varpi \end{pmatrix} = \mathcal{J}_{\text{red}} \cdot \dot{q} \quad (2)$$

III. Theorical aspects

The first three subsections present the controllers dedicated to visual servoing, occlusion avoidance and obstacle bypassing. The global control strategy is given in the last one.

Remark 1: For the problem to be well stated, we consider that no obstacle lies in a close neighborhood of the target.

A. Visual servoing

Here, we present the nominal vision-based controller in the case that occlusions and collisions do not occur. We consider the visual servoing technique given in [4]. This approach relies on the task function formalism, which consists in expressing the desired task as a task function e to be regulated to zero [3]. A sufficient condition that guarantees the control problem to be well conditioned is that e is $\rho-$admissible. Indeed, this property ensures the existence of a diffeomorphism between the task space and the state space, so that the ideal trajectory q_r corresponding to $e = 0$ is unique. This condition is fulfilled if $\frac{\partial e}{\partial q}$ is regular around q_r [3].

In our application, the target is made of 4 points, defining an 8-dimensional vector of visual signals s in the camera plane. The variation of the signals is related to the kinematic screw $\mathcal{T}^c_{\text{red}}$ by the interaction matrix $\mathcal{L}_{\text{red}}$ [4]:

$$\dot{s} = \mathcal{L}_{\text{red}}\mathcal{T}^c_{\text{red}} \quad (3)$$

Using the pinhole camera model, a point p of coordinates $(x, y, z)^T$ in $\mathcal{F}_C$ projected into a point $P(X, Y)$ in the image plane (see figure 2), $\mathcal{L}_{\text{red}}$ is directly deduced from the optic flow equations [4] and given by the following matrix :

$$\mathcal{L}_{\text{red}}(P) = \begin{bmatrix} 0 & \frac{X}{z} & XY \\ -\frac{1}{z} & \frac{Y}{z} & 1 + Y^2 \end{bmatrix} \quad (4)$$

$\mathcal{L}_{\text{red}}$ has a reduced number of columns to be compatible with the dimension of $\mathcal{T}^c_{\text{red}}$. Following the task function formalism, the visual servoing task is defined as the regulation of the following error fonction :

$$e_{\text{VS}}(q(t)) = \mathcal{C}(s(q(t)) - s^*) \quad (5)$$

where s^* is the desired visual signal and $q = [l,\, \theta,\, \theta_{pl}]^T$, l representing the curvilinear abscissa of the robot. $\mathcal{C}$ is a full-rank 3×8 combination matrix which allows to take into account more visual features than available degrees of freedom. A simple way to choose C is to consider the pseudo-inverse of the interaction matrix : $\mathcal{C} = (\mathcal{L}^T_{\text{red}}\mathcal{L}_{\text{red}})^{-1}\mathcal{L}^T_{\text{red}}$ as proposed in [4]. In this way, the positionning task jacobian $\frac{\partial e_{\text{VS}}}{\partial q} = \mathcal{C}\mathcal{L}_{\text{red}}\mathcal{J}_{\text{red}}$ can be simplified into $\mathcal{J}_{\text{red}}$, which is always invertible as $\det(\mathcal{J}_{\text{red}}) = D_x \neq 0$. The ρ-admissibility property is then insured. The control law design relies on this property. Indeed, classically, a kinematic controller can be determined by imposing an exponential convergence of e_{VS} to zero :

$$\dot{e}_{\text{VS}} = C\mathcal{L}_{\text{red}}\mathcal{J}_{\text{red}}\dot{q} = \mathcal{J}_{\text{red}}\dot{q} = -\lambda_{\text{VS}}e_{\text{VS}} \quad (6)$$

where λ_{VS} is a positive scalar or a positive definite matrix. ¿From this last relation together with equations (2), (3), (5) and thanks to the $\rho-$admissibility property, we can deduce :

$$\dot{q}_{\text{VS}} = \mathcal{J}^{-1}_{\text{red}}(-\lambda_{\text{VS}})e_{\text{VS}} \quad (7)$$

B. Occlusion avoidance

Let us suppose now that an occluding object $\mathcal{O}$ is present in the image. Its projection appears in the image plane as shown on figure 2. X_{im} and Y_{im} correspond to the axes of the frame attached to the image plane. The proposed strategy only relies on the detection of the two borders of $\mathcal{O}$, defined by $Y^-_{\mathcal{O}}$ and $Y^+_{\mathcal{O}}$. As the camera is constrained to move in the horizontal plane, there is no loss of generality in stating the reasoning on $Y^-_{\mathcal{O}}$ and $Y^+_{\mathcal{O}}$.

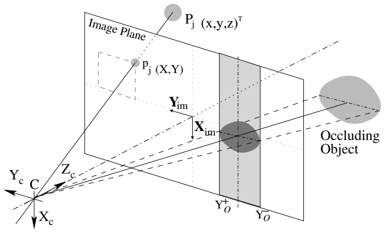

Fig. 2. Projection of the occluding object in the image plane

To define the occlusion avoidance task, we have chosen to use the redundant task function formalism [3]. This formalism has been already used to perform a vision-based task while following a trajectory [4] or avoiding joint limits, singularities [14] and occlusions for manipulators [7]. It has been also used to avoid obstacles in visually guided navigation tasks for mobile robots [12]. Let e_1 be a redundant task, that is a low-dimensioned task which does not constraint all degrees of freedom of the robot. Therefore, e_1 is not ρ-admissible and an infinity of ideal trajectories q_r corresponds to the regulation of e_1 to zero. The basic idea of the formalism is to benefit from this redundancy to perform an additional objective. This latter can be modelled as a cost function h to be minimized under the constraint that e_1 is perfectly performed. The resolution of this optimization problem leads one to define e as follows [3]:

$$e = W^+ e_1 + \beta(I - W^+ W)g$$

where $W^+ = W^T(WW^T)^{-1}$, $g = \frac{\partial h}{\partial q}$ and β is a positive scalar. Under some assumptions, which are verified if $W = \frac{\partial e_1}{\partial q}$, the task jacobian $\frac{\partial e}{\partial q}$ is positive-definite around q_r, insuring that e is ρ-admissible [3].

Now, let us apply these theoretical results to preserve the visual features visibility in the image. We propose to define the occlusion task as the priority task, which leads to the following task function e_{OA}:

$$e_{\text{OA}}(q(t)) = W^+_{\text{occ}} e_{\text{occ}} + \beta_{\text{OA}}(I - W^+_{\text{occ}} W_{\text{occ}})g \qquad (8)$$

where e_{occ} is the redundant task function allowing to avoid the occlusions, $W_{\text{occ}} = \frac{\partial e_{\text{occ}}}{\partial q}$, β_{OA} is a positive scalar, and $g = \frac{\partial h}{\partial q}$. Instead of defining the criterium to avoid collisions as in [13], we choose h to keep on tracking the target while avoiding occlusions. We get :

$$h = \frac{1}{2}(s - s^*)^T(s - s^*) \quad \Rightarrow g = \left((s - s^*)^T \mathcal{L}_{\text{red}} \mathcal{J}_{\text{red}}\right)^T \qquad (9)$$

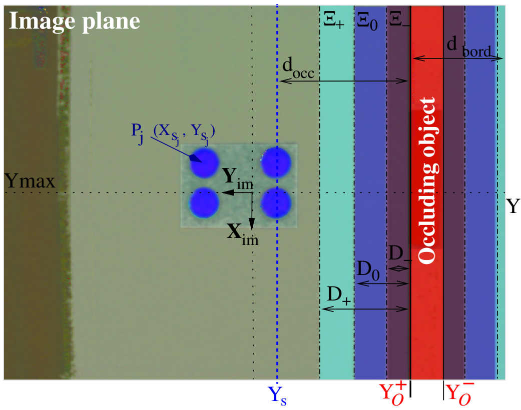

Fig. 3. Definition of the relevant distances for occlusion avoidance on an acquired RGB-normalized image

Now, let us define the prioritary task function e_{occ} to avoid occlusions. Considering figure 3, we denote by (X_{s_j}, Y_{s_j}) the coordinates of each point P_j of the target in the image frame, Y_{min} and Y_{max} representing the ordinates of the two image sides. We introduce the following distances:

- d_{occ} characterizes the distance before occlusion, that is the shortest distance between the visual features s and the occluding object $\mathcal{O}$. If there is no occluding object in the picture, then d_{occ} is computed from the closest image side.
- d_{bord} denotes the distance separating the occluding object $\mathcal{O}$ and the opposite image side to the visual features.
- D_+ defines an envelope Ξ_+ delimiting the region inside which the risk of occlusion is detected.
- D_0 and D_- correspond to two additional envelopes Ξ_0 and Ξ_-. They respectively surround the critical zone inside which it is necessary to start avoiding occlusion and the region where the occlusion danger is the highest. They will be used in the sequel to determine the global controller.

¿From these definitions, we propose the following redundant task function e_{occ} :

$$e_{\text{occ}} = \begin{pmatrix} \tan\left(\frac{\pi}{2} - \frac{\pi}{2} \cdot \frac{d_{\text{occ}}}{D_+}\right) \\ d_{\text{bord}} \end{pmatrix} \qquad (10)$$

The first component allows to avoid target occlusions or loss : indeed, it increases when the occluding object or image side is getting closer to the visual features. Note that, $\forall d_{\text{occ}} \geq D_+$, e_{occ} is maintained to zero. The second component makes the occluding object go out of the image, which is realized when d_{bord} vanishes. Let us remark that these two subtasks must be compatible (that is, they can be simultaneously realized) in order to guarantee the control problem to be well stated. This condition is fulfilled by construction thanks to the choice of d_{occ} and d_{bord} (see figure 3). Now, it remains to design a controller allowing to regulate it to zero. As W_{occ} and β_{OA} are chosen to fulfill the assumptions of the redundant task formalism [3], the task jacobian $\frac{\partial e_{\text{OA}}}{\partial q}$ is positive definite around the ideal trajectory and e_{OA} is ρ-admissible. This result also allows to simplify the control design as it can be shown that a controller making e_{OA} vanish is given by [4]:

$$\dot{q}_{\text{OA}} = -\lambda_{\text{OA}} e_{\text{OA}} \qquad (11)$$

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

C. Obstacle avoidance

The avoidance strategy is based on the ultrasonic data from which we compute a set of values characterizing locally any obstacle located at a distance inferior to d_+ (see figure 4). We obtain a couple $(d_{\text{coll}}, \alpha)$, where d_{coll} is the signed distance between M and the closest point Q on the obstacle, and α is the angle between the tangent to the obstacle at Q and the robot direction. Note that there exists two angles α corresponding to the two possible directions for the avoidance motion. As the obstacle can also be an occluding object, we propose to maintain the target visibility by defining α so that the robot moves around the obstacle in the direction given by the pan-platform.

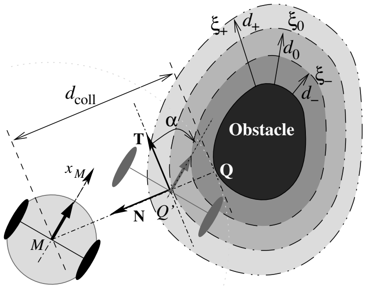

Fig. 4. Obstacle avoidance

Remark 2: For the problem to be well stated, we assume that $d_{\text{coll}} > 2d_+$ to prevent the robot from considering several obstacles simultaneously.

Around each obstacle, three envelopes are defined (see figure 4). The first one ξ_+ located at a distance d_+ surrounds the zone inside which the obstacle is detected by the robot.

The second one ξ_0, located at a lower distance d_0 constitutes the virtual path along which the reference point M will move around the obstacle. The last one ξ_- defines the region inside which the risk of collision is maximal (this envelope will be used in the sequel to define the global controller). Using the path-following formalism introduced in [15], we define a mobile frame on ξ_0 whose origin Q' is the orthogonal projection of M. During obstacle avoidance, the robot linear velocity is supposed to be kept constant. Let $\delta = d_{\text{coll}} - d_0$ be the signed distance between M and Q'. With respect to the moving frame, the dynamics of the error terms (δ, α) is described by the following system :

$$\left\{ \begin{array}{l} \dot{\delta} = v \sin \alpha \\ \dot{\alpha} = \omega - v\chi \cos \alpha \end{array} \right. \quad \text{with } \chi = \frac{\frac{\sigma}{R}}{1 + \frac{\sigma}{R}\delta} \qquad (12)$$

where $\sigma = \{-1, 0, +1\}$ depending on the sense of the robot motion around the obstacle and R is the curvature radius of the obstacle. The path following problem is classically defined as the search for a controller ω allowing to steer the pair (δ, α) to $(0, 0)$ under the assumption that v never vanishes to preserve the system controllability. Here, our goal is to solve this problem using the task function formalism. To this aim, we have to find a task function whose regulation to zero will make δ and α vanish while insuring $v \neq 0$. We propose the following redundant task function e_{coll} :

$$e_{\text{coll}} = \left(\begin{array}{c} l - v_r t \\ \delta + k\alpha \end{array} \right) \qquad (13)$$

where l is the curvilinear abscissa of point M and k a positive scalar. The first component of e_{coll} allows to regulate the linear velocity of the mobile base to a nonzero constant value[1] v_r. The second component can be seen as a sliding variable whose regulation to zero makes both (δ, α) vanish (see [16] for a detailed proof). Therefore, the regulation to zero of e_{coll} guarantees that the robot follows the security envelope ξ_0 with a nonzero linear velocity, insuring non collision. As the chosen task function does not constraint the whole degrees of freedom of the robot, we use the redundant task function formalism to avoid target loss and occlusions at best and we define the corresponding cost function by $h_{\text{occ}} = \frac{1}{d_{\text{occ}}}$. The global task function e_{CA} is then given by :

$$e_{\text{CA}} = W^+_{\text{coll}} e_{\text{coll}} + \beta_{\text{CA}}(I - W^+_{\text{coll}} W_{\text{coll}}) g_{\text{occ}} \qquad (14)$$

Following the redundant task function formalism, a controller making e_{CA} vanish is given by:

$$\dot{q} = \dot{q}_{\text{CA}} = -\lambda_{\text{CA}} e_{\text{CA}} \qquad (15)$$

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

D. The global controller

There exist two approaches for sequencing tasks. In the first one, the switch between two successive tasks is dynamically performed using the definition of a differential structure on the robot state space [17], or benefiting from the redundant task function formalism to stack elementary tasks and design control laws guaranteeing smooth transitions [18]. The second class of tasks sequencing techniques relies on convex combinations between the successive task functions [5] [12] or the successive controllers [10] [11]. In that case, applications can be more easily carried out, but it is usually harder to guarantee the task feasibility. The control proposed here relies on the second approach. Our idea is to combine the three previously defined controllers (7), (11) and (15) to drive the robot the best way depending on the environment. To this aim, we introduce two parameters μ_{occ} and $\mu_{\text{coll}} \in [0, 1]$ depending on the risk of occlusion and of collision as follows:

• If the occluding object $\mathcal{O}$ lies outside the region defined by Ξ_0 or is not in the image and if there is no obstacle in the robot vicinity, μ_{occ} and μ_{coll} are fixed to 0. Only $\dot{q}_{\text{VS}}$ must be sent to the robot in this case.

• If the visual features enter the zone delimited by Ξ_0, μ_{occ} progressively increases to reach 1 when Ξ_- is crossed. At this time, μ_{occ} is fixed to 1 until the object $\mathcal{O}$ leaves the image or at least goes out the critical zone. When one of these two events occurs, μ_{occ} decreases and vanishes once s crosses Ξ_+. Let OCCLU be the flag indicating that μ_{occ} has reached its maximal value and D_{leave} the value of the distance d_{occ} when $\text{OCCLU} = 1$. We propose the following expression:

$$\left\{ \begin{array}{ll} \mu_{\text{occ}} = 0 & \text{if } d_{\text{occ}} > D_0 \text{and OCCLU} = 0 \\ \mu_{\text{occ}} = \frac{d_{\text{occ}} - D_0}{D_- - D_0} & \text{if } d_{\text{occ}} \in [D_-, D_0] \text{ and OCCLU} = 0 \\ \mu_{\text{occ}} = \frac{d_{\text{occ}} - D_+}{D_{\text{leave}} - D_+} & \text{if } d_{\text{occ}} \in [D_{\text{leave}}, D_+] \\ & \text{and } (d_{\text{bord}} = 0 \text{ or } d_{\text{occ}} \geq D_0) \\ \mu_{\text{occ}} = 1 & \text{otherwise} \end{array} \right.$$

• If the mobile base enters the zone surrounded by ξ_0, μ_{coll} is continuously increased to 1 when $d_{\text{coll}} \leq d_-$. If ξ_- is never crossed, μ_{coll} is brought back to 0, once $d \geq d_0$. If μ_{coll} reaches 1, the collision risk is maximum and a flag AVOID is enabled. As the robot safety is considered to be the most important objective, the global controller must be designed so that only $\dot{q}_{\text{CA}}$ is applied to the vehicle once μ_{coll} has reached 1. In this way, it is possible to guarantee non collision while occlusions are avoided at best. The robot is then brought back on the security envelope ξ_0 and follows it until the condition to leave is fulfilled. This event occurs when the camera and the mobile base have the same direction. A flag LEAVE is then positioned to 1 and μ_{coll} is decreased to vanish on ξ_+. Let d_s be the value of the distance d_{coll} when $\text{LEAVE} = 1$. Therefore, μ_{coll} depends on d_{coll} as follows:

$$\left\{ \begin{array}{ll} \mu_{\text{coll}} = 0 & \text{if } d_{\text{coll}} > d_0 \text{ and AVOID} = 0 \\ & \text{and LEAVE} = 0 \\ \mu_{\text{coll}} = \frac{d_{\text{coll}} - d_0}{d_- - d_0} & \text{if } d_{\text{coll}} \in [d_-, d_0] \text{ and AVOID} = 0 \\ \mu_{\text{coll}} = \frac{d_{\text{coll}} - d_+}{d_s - d_+} & \text{if } d_{\text{coll}} \in [d_s, d_+] \text{ and LEAVE} = 1 \\ \mu_{\text{coll}} = 1 & \text{otherwise} \end{array} \right.$$

Following this reasoning and recalling that $\dot{q}_{\text{VS}}$, $\dot{q}_{\text{OA}}$ and $\dot{q}_{\text{CA}}$ are given by equations (7), (11) and (15), we propose the following global controller :

$$\dot{q} = (1 - \mu_{\text{occ}})(1 - \mu_{\text{coll}})\dot{q}_{\text{VS}} + \mu_{\text{occ}}(1 - \mu_{\text{coll}})\dot{q}_{\text{OA}} + \mu_{\text{coll}}\dot{q}_{\text{CA}} \quad (16)$$

[1] v_r must be chosen small enough to let the robot sufficiently slow down to avoid collision when entering the critical zone.

Remark 3: The presence of an occluding object in the image does not necessarily mean that a collision may occur. Indeed, an obstacle may be detected by the camera before it becomes dangerous for the mobile base. This is the reason why this approach, simply by considering separately obstacle bypassing and occlusion avoidance, leads to better results for the robot safety than the one described in [13] where these two tasks are treated simultaneously.

Remark 4: The different envelops are chosen close enough to reduce the transition phase duration. Recalling that μ_{occ} and μ_{coll} are maintained to 1 once they have reached this value, the control strategy is built to insure that the robot will be rapidly controlled by the most relevant controller. In this way, the risks of instability, target loss or collisions during the switch are significantly reduced and the task feasibility can be considered to be guaranteed.

IV. Experimental aspects

A. *Description of the robotic system*

We have implemented the above control laws on the mobile robot SuperScout II [2] (see figure 5). This robot is a cylindric cart-like vehicle, specific to indoor navigation. It is equipped with sixteen ultrasonic sensors. Incremental encoders mounted on the wheels provide the approximative pose of the mobile base. A DFW-VL500 Sony color digital IEEE1394 camera, mounted on a pan-platform, captures high-quality pictures in YUV 4:2:2 format with 640 × 480 resolution. The robot is controlled by an on-board laptop computer running under Linux on which is installed a specific architecture called G$^{\text{en}}$oM (Generator of Module). G$^{\text{en}}$oM is a LAAS-CNRS tool dedicated to real-time software architectures design for complex embedded systems [19].

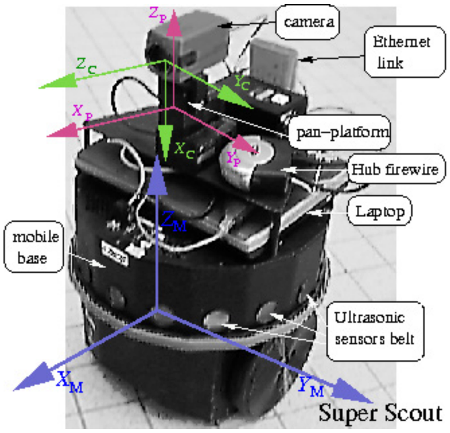

Fig. 5. Nomadic SuperScout II

The implementation of our control strategy requires different basic services. They are provided by two already existing modules respectively dedicated to the pan-platform control and the robot kinematic control together with data sensory processing. First, we have designed a specific module to manage the camera features. It allows to initialize, configure the camera, acquire pictures and process them to extract the visual signal s and the occluding object position from which we compute d_{occ} and d_{bord}. The image processing pipeline is shown on figure 6, which presents an example of extraction of the target together with the occluding object. The proposed image processing relies on the following hypotheses to reduce the computation time : all the occluding objects have the same color, the occluding object and the target colors are different, and finally, an occluding object detected in the image can never be positioned *behind* the target in the 3D scene. We have then embedded our control strategy in a second module which offers the required services to realize vision-based navigation tasks amidst obstacles: robotic system initialization, software emergency stop, parameters settings (choice of the different control gains, of the desired visual features $s^*\dots$), execution of a vision-based task with or without taking into account the occlusion and/or collision phenomena. We present below the obtained experimentation results and describe the different encountered problems.

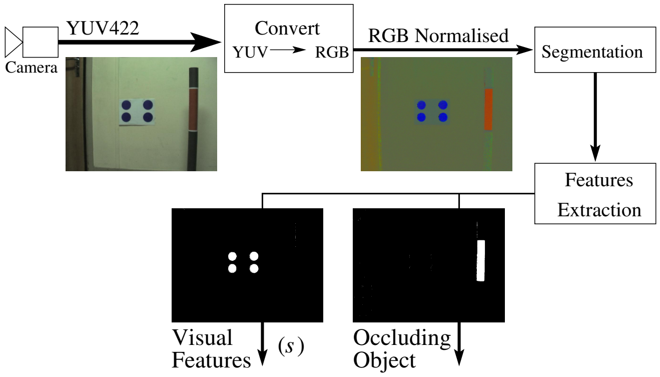

Fig. 6. Visual features and occluding object extraction

B. *Results*

Our method has been implemented to realize a mission whose objective is to position the camera in front of a given target. To validate our approach, the environment has been cluttered with two cylindric obstacles which may occlude the camera or represent a danger for the mobile base. For this test, D_-, D_0 and D_+ have been respectively fixed to 40, 90 and 120 pixels, and d_+, d_0, d_- to 0.7m, 0.55m, and 0.4m. The robot initial configuration and the positions of the two obstacles have been chosen to induce occlusions or collisions. The obtained results are presented on figures 7, 8 and 9.

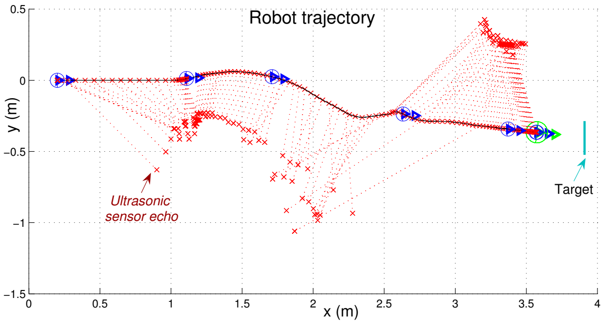

Fig. 7. Robot trajectory

As shown on figures 7 and 8, the task is correctly performed: target occlusions and obstacle collisions never occur during the whole mission. At the beginning of the task,

[2] The mobile robot SuperScout II is provided by the AIP-PRIMECA.

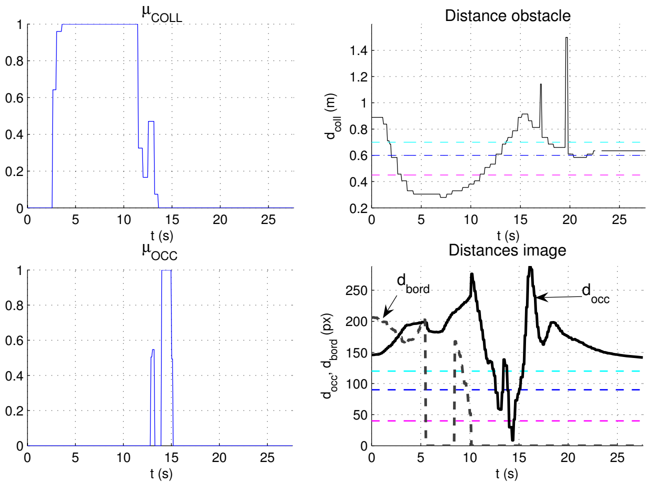

Fig. 8. Evolution of $\mu_{\rm occ}$, $\mu_{\rm coll}$, and relevant distances

there is no risk of occlusion nor collision, the robot is only controlled by $\dot{q}_{\rm VS}$ and starts converging towards the target. When the vehicle enters the vicinity of the first encountered obstacle, $\mu_{\rm coll}$ is progressively increased to 1 (see figure 8). Then the robot starts avoiding the first obstacle (modelled by the first set of ultrasonic sensors echoes on figure 7), while minimizing occlusions at best. During this phase, in spite of occlusion minimization, $\mu_{\rm occ}$ keeps on rising. Therefore, at this step, the robot has to avoid both collision and occlusion. When the robot leaves the vicinity of the first obstacle, $\mu_{\rm coll}$ vanishes. However, as the occlusion risk still increases to become maximum, the sole controller $\dot{q}_{\rm OA}$ is used to guarantee a safe motion. When there is no more collision or occlusion risk, the robot converges towards the target using the sole visual servoing controller. The navigation task is finally successfully realized.

Remark 5: As the second obstacle is a very small diameter cylinder, it may occur that the ultrasonic sensors cannot detect it efficiently. This phenomenon happens twice during the task, inducing the two jumps which appear in the evolution of $d_{\rm coll}$.

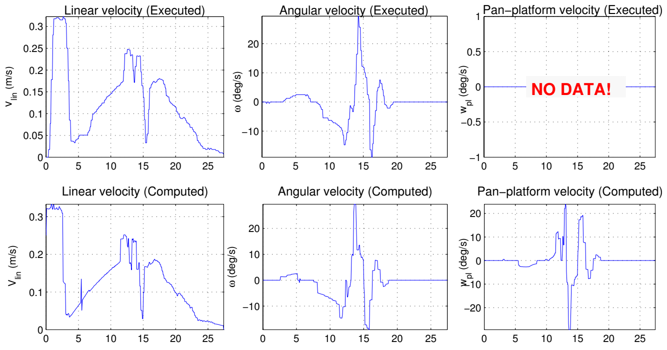

Fig. 9. Control inputs ($\dot{q} = (v, \omega, \varpi)^T$)

The different relevant velocities are presented on figure 9. The first and last lines respectively show the evolution of the executed velocities and the computed control inputs $\dot{q} = (v, \omega, \varpi)^T$. As we can see, the computed control law given by (16) is correctly executed without delay nor saturation problem. This result demonstrates the adequation of our control strategy with the low-level feedback loops which control the robot actuators. The obtained inputs are also consistent with the trajectory performed by the robot.

V. Conclusion

The proposed sensor-based controller allows a mobile robot to perform safely a vision-based task in a cluttered environment. The method relies on the switch between different controllers depending on the risks of collision and occlusion. The obtained experimental results are quite satisfactory, validating our approach. However, this work is restricted to missions where occlusions can be *effectively* avoided, which is not the case of all robotic tasks. Therefore, further extensions will have to accept that occlusions may occur rather than to avoid them.

References

[1] P. Corke, *Visual control of robots : High performance visual servoing*. Research Studies Press LTD, 1996.

[2] S. Hutchinson, G. Hager, and P. Corke, “A tutorial on visual servo control,” *IEEE Trans. Robot. Automat.*, Oct. 1996.

[3] C. Samson, M. L. Borgne, and B. Espiau, *Robot Control : The task function approach*, Clarendon Press ed. Oxford, England: Oxford science publications, 1991.

[4] B. Espiau, F. Chaumette, and P. Rives, “A new approach to visual servoing in robotics,” *IEEE Trans. Robot. Automat.*, June 1992.

[5] R. Pissard-Gibollet and P. Rives, “Applying visual servoing techniques to control a mobile hand-eye system,” in *ICRA’95*, Nagoya, Japan, May 1995.

[6] Y. Mezouar and F. Chaumette, “Avoiding self-occlusions and preserving visibility by path planning in the image,” *Robotics and Autonomous Systems*, Nov. 2002.

[7] E. Marchand and G. Hager, “Dynamic sensor planning in visual servoing,” in *ICRA’98*, Leuven, Belgium, May 1998.

[8] A. I. Comport, E. Marchand, and F. Chaumette, “Robust model-based tracking for robot vision,” in *IROS’04*, Sendai, Japan, Oct. 2004.

[9] P. Wunsch and G. Hirzinger, “Real-time visual tracking of 3D objects with dynamic handling of occlusion,” in *ICRA’97*, Albuberque, Mexico, Apr. 1997.

[10] V. Cadenat, P. Souères, and M. Courdesses, “An hybrid control for avoiding obstacles during a vision-based tracking task,” in *ECC’99*, Karlsruhe, Germany, Sept. 1999.

[11] V. Cadenat, P. Souères, R. Swain, and M. Devy, “A controller to perform a visually guided tracking task in a cluttered environment,” in *IROS’99*, Korea, Oct. 1999.

[12] V. Cadenat, P. Souères, and M. Courdesses, “Using system redundancy to perform a sensor-based navigation task amidst obstacles,” *Int. J. Robot. Automat.*, 2001.

[13] D. Folio and V. Cadenat, “Using redundancy to avoid simultaneously occlusions and collisions while performing a vision-based task amidst obstacles,” in *ECMR’05*, Ancona, Italy, Sept. 2005.

[14] F. Chaumette and E. Marchand, “A new redundancy-based iterative scheme for avoiding joint limits: application to visual servoing,” in *ICRA’00*, San Francisco, CA, USA, May 2000.

[15] C. Samson, “Path following and time-varing feedback statbilization of a wheeled mobile robot,” in *ICARCV’92*, Singapore, Sept. 1992.

[16] P. Souères, T. Hamel, and V. Cadenat, “A path following controller for wheeled robots which allows to avoid obstacles during the transition phase,” in *ICRA’98*, Leuven, Belgium, May 1998.

[17] P. Souères and V. Cadenat, “Dynamical sequence of multi-sensor based tasks for mobile robots navigation,” in *SYROCO’03*, Wroclaw, Poland, Sept. 2003.

[18] N. Mansard and F. Chaumette, “Tasks sequencing for visual servoing,” in *IROS’04*, Sendai, Japan, Sept. 2004.

[19] S. Fleury and M. Herrb, *G^{en}oM : User Manual*, LAAS-CNRS, May 2001. [Online]. Available: http://www.laas.fr/ sara/RIA/RIA-genom.html