A controller to avoid both occlusions and obstacles during a vision-based navigation task in a cluttered environment

David Folio and Viviane Cadenat

Abstract—This paper presents a sensor-based controller allowing to visually drive a mobile robot towards a target while avoiding visual features occlusions and obstacle collisions. We consider the model of a cart-like robot equipped with proximetric sensors and a camera mounted on a pan-platform. The proposed method relies on the continuous switch between three controllers realizing respectively the nominal vision-based task, the obstacle bypassing and the occlusion avoidance. Simulation results are given at the end of the paper.

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. For example, Martinet et al. [3] and Bellot et al. [4] use respectively an $H_\infty$ controller and LMI techniques to perform vision-based tasks. The task function formalism [5] also provides a general framework for designing sensor-based control laws. Indeed, this formalism can be applied to manipulators [6] as well as to nonholonomic mobile robots provided that, in this case, the camera is able to move independently from the base [7]. The visual servoing techniques mentioned above require that the image features remain always in the field of view of the camera, and that they are never occluded during the whole execution of the task. Most of the works which address this kind of problems are dedicated to manipulator arms. For example, in [8], the authors propose a method allowing to avoid self-occlusions and preserve visibility by path planning in the image for such robots. In [9], Marchand et al. benefit from manipulator arm redundancy to perform a vision-based task while avoiding occlusions, visual features loss and obstacles. In [10], 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 [11], Wunsch et al. propose a model-based method allowing a robot to visually track 3D objects while occlusions are continuously predicted.

In this paper, we address the problem of avoiding occlusions and collisions during the execution of a given vision-based task in a cluttered environment. We consider a nonholonomic mobile robot equipped with proximetric sensors and a camera mounted on a pan-platform. The proposed method is in the sequel of previous works where the idea was to merge classical vision-based control to obstacle avoidance techniques based on nonlinear path following design [12], potential field approach [13] or on the task function formalism [14] to perform a visually guided navigation task in a cluttered environment. However, as these works were a first attempt to answer this kind of problem, they were restricted to the case where occlusions could not occur. Therefore, in this article, we aim at extending these techniques to improve the robot abilities and avoid both collisions and occlusions. The proposed strategy consists in designing three controllers, the first one performing the desired vision-based task in the free space, the second one guaranteeing occlusion avoidance whenever a risk of occlusion occurs and the last one insuring non collision in the vicinity of the obstacles. Then, we switch from one controller to the other depending on the risk of occlusion and of collision.

The paper is organized as follows: System modelling and problem statement are given in section II. The different controllers and the control strategy are presented in section III. Finally, simulation results are described in section IV.

II. MODELLING AND PROBLEM STATEMENT

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 [7]:

$$
\begin{pmatrix}
\dot{x} \\
\dot{y} \\
\dot{\theta} \\
\dot{\theta}_{pl}
\end{pmatrix} =
\begin{pmatrix}
\cos \theta & 0 & 0 \\
\sin \theta & 0 & 0 \\
0 & 1 & 0 \\
0 & 1 & 1
\end{pmatrix}
\begin{pmatrix}
v \\
w_1 \\
w_2
\end{pmatrix}
$$

(1)

($x, y$) are the coordinates of the robot reference point $M$ with respect to the world frame $\mathcal{F}_D$. $\theta$ and $\theta_{pl}$ are respectively the direction of the vehicle and the direction of the pan-platform with respect to the $x$-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. It consists of an horizontal translation of vector \((a, b, 0)^T\) and a rotation of angle \(\frac{\pi}{2}\) about the \(y_P\)-axis. The control input is defined by the vector \(\hat{q} = (v, \omega, \varpi)^T\), where \(v\) and \(\omega\) are the cart linear and angular velocities, and \(\varpi\) is the pan-platform angular velocity.

At each configuration of the robot, the variation of an 8-dimensional vector of visual signals is directly deduced from the optic flow equations [6] and a reduced jacobian matrix \(T_{red}\) of columns to be compatible with the dimension of \(T_{red}^c\):

\[
L_{red} = \begin{bmatrix}
0 & \frac{X}{2} & XY \\
-\frac{1}{2} & \frac{Y}{2} & 1 + Y^2
\end{bmatrix}
\]

Following the task function formalism, the task is defined as the regulation of an error function \(e_{vs}(q(t))\) to zero:

\[
e_{vs}(q(t)) = C (s(q(t)) - s^*)
\]

where \(s^*\) is the desired value of the visual signal and \(q = [l, \theta, \theta_{pl}]^T\), \(l\) representing the curvilinear abscissa of the robot. \(C\) is a full-rank \(3 \times 8\) 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, that is \(C = (L_{red}^T L_{red})^{-1} L_{red}^T\) as proposed in [6]. In this way, the positioning task jacobian \(\frac{\partial e_{vs}}{\partial q} = C L_{red} J_{red}\) can be simplified into \(J_{red}\), which is always invertible as \(\det(J_{red}) = D_x \neq 0\). The \(\rho\)-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:

\[
\dot{e}_{vs} = CL_{red} J_{red} \hat{q} = J_{red} \dot{q} = -\lambda_{vs} e_{vs}
\]

where \(\lambda_{vs}\) 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:

\[
\dot{q} = \dot{q}_{vs} = -\lambda_{vs} J_{red}^{-1} e_{vs}
\]

### B. The occlusion avoidance control

Now, we suppose that an occluding object \(O\) is present in the image plane as shown on figure 2 and we denote by \(Y_{obs}\) and \(Y_{obs}^+\), the ordinates of its left and right borders. \(X_{im}\) 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 \(O\). As the camera is constrained to move in the horizontal plane, there is no loss of generality in stating the reasoning on \(Y_{obs}\) and \(Y_{obs}^+\).

Our goal is to define a task function allowing to preserve the visual features visibility in the image. To this aim, we have chosen to use the redundant task function formalism [5]. This formalism has been already used to perform a vision-based task while following a trajectory [6] or avoiding joint limits, singularities [15] and occlusions [9] for manipulators. It has also been used to avoid obstacles in visually guided navigation tasks for mobile robots [14]. Let \(e_1\) be a redundant task, that is a low-dimensional task which does not constraint...
all degrees of freedom of the robot. Therefore, \( e_1 \) is not \( \rho \)-admissible and an infinity of ideal trajectories \( q_t \) 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 [5]:

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

where \( W^+ = W^T(WW^T)^{-1} \) is the pseudo-inverse of \( W \), \( g = \frac{\partial h}{\partial q} \) and \( \beta \) is a positive scalar, sufficiently small in the sense defined in [5]. Under some assumptions (which are verified if \( W = \frac{\partial h}{\partial q} \)), the task jacobian \( \frac{\partial h}{\partial q} \) is positive-definite around \( q_t \), insuring that \( e \) is \( \rho \)-admissible [5].

Our objective is to apply these theoretical results to avoid occlusions while keeping the target in the image. We have chosen to define the occlusion avoidance as the primary task. The target tracking will then be considered as the secondary objective and will be modelled as a criterion \( h_s \) to be minimized. We propose the following occlusion avoidance task function \( e_{oa} \):

\[
e_{oa}(q(t)) = W_{occ}^+ e_{occ} + \beta_{oa}(I - W_{occ}^+ W_{occ}) g_s
\]

\( e_{occ} \) is the redundant task function allowing to avoid the occlusions, \( W_{occ} = \frac{\partial e_{occ}}{\partial q} \), \( g_s = \frac{\partial h_s}{\partial q} \) and \( \beta_{oa} \) is a positive scalar as explained above. We propose the following criterion to track the target and keep it in the camera line of sight:

\[
h_s = \frac{1}{2} (s - s^*)^T (s - s^*) \Rightarrow g_s = ((s - s^*)^T L_{red} J_{red})^T
\]

Fig. 3. Definition of the relevant distances for occlusion avoidance

Now, let us define the priority task function \( e_{occ} \) to avoid occlusions. Considering figure 3, we denote by \((X_{s+},Y_{s+})\) the coordinates of each point \( P_j \) of the target in the image frame, \( Y_{\text{min}} \) and \( Y_{\text{max}} \) 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 \( O \). It can be defined as:

\[
d_{occ} = \min \left( \min_j |Y_j - Y_{\text{obs}^+}|, \min_j |Y_j - Y_{\text{obs}^-}| \right) = |Y_s - Y_{\text{occ}}|
\]

where \( Y_s \) is the ordinate of the closest point \( P_j \) to object \( O \), while \( Y_{\text{occ}} \) represents the closest border of \( O \) to the visual features (in the case of figure 3, \( Y_{\text{occ}} = Y_{\text{obs}}^- \)).

- \( d_{bord} \) is defined by the distance separating the occluding object \( O \) and the opposite image side to the visual features:

\[
d_{bord} = \min \left( |Y_{\text{obs}^+} - Y_{\text{max}}|, |Y_{\text{obs}^-} - Y_{\text{min}}| \right) = |Y_{\text{occ}} - Y_{\text{bord}}|
\]

where \( Y_{\text{bord}} \) corresponds to the image border towards which the occluding object must move to leave the image without occluding the target (see figure 3).

- \( D_+ \) defines an envelope \( \Xi_+ \) delimiting the region inside which the risk of occlusion is detected.

- \( D_0 \) and \( D_- \) define two additional envelopes \( \Xi_0 \) and \( \Xi_- \). They respectively surround the critical zone inside which it is necessary to start avoiding occlusion and the region where the danger of occlusion 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 \( e_{occ} \):

\[
e_{occ} = \left( \frac{\tan \left( \frac{\pi}{2} - \frac{\pi}{2} \frac{d_{bord}}{D_+} \right)}{d_{bord}} \right)
\]

The first component allows to avoid target occlusions: indeed, it increases when the occluding object is getting closer to the visual features and becomes infinite when \( d_{occ} \) tends to zero. On the contrary, it decreases when the occluding object is moving far from the visual features and vanishes when \( d_{occ} \) equals \( D_+ \). Note that, \( \forall d_{occ} \geq D_+ \), \( e_{occ} \) 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 tasks must be compatible (that is, they can be realized simultaneously) 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, let us determine \( W_{occ} = \frac{\partial e_{occ}}{\partial q} \). We get:

\[
W_{occ} = \left( -\frac{1}{\pi} \frac{n}{\pi} e_{occ} \left( 1 + \tan^2 \left( \frac{\pi}{2} - \frac{\pi}{2} \frac{d_{occ}}{D_+} \right) \right) \frac{\partial Y_{occ}}{\partial q} - \frac{\partial Y_{bord}}{\partial q} \right)
\]

where \( e_{occ} = \text{sign}(Y_s - Y_{\text{occ}}) \) and \( e_{bord} = \text{sign}(Y_{\text{occ}} - Y_{\text{bord}}) \) while \( \frac{\partial Y_{occ}}{\partial q} \) and \( \frac{\partial Y_{bord}}{\partial q} \) are deduced from the optic flow equations as follows:

\[
\begin{align*}
\frac{\partial Y_{occ}}{\partial q} &= \left( -\frac{1}{z_{occ}} \frac{Y_s}{z_{occ}} - 1 + Y_s^2 \right) J_{red} \\
\frac{\partial Y_{bord}}{\partial q} &= \left( -\frac{1}{z_{bord}} \frac{Y_{occ}}{z_{bord}} - 1 + Y_{occ}^2 \right) J_{red}
\end{align*}
\]

where \( z_{occ} \) and \( z_{bord} \) are the depth of the target and of the occluding object expressed in frame \( F_C \).

At this step, the task function \( e_{oa} \) guaranteeing the occlusion avoidance while keeping the target in the image is completely determined (see relation (8)). Now, it remains to design a controller allowing to regulate it to zero. As \( W_{oa} \) and \( \beta_{oa} \) are chosen to fulfill the assumptions of the redundant task formalism [5], the task jacobian \( \frac{\partial h_{oa}}{\partial q} \) is positive definite around the ideal trajectory and \( e_{oa} \) is \( \rho \)-admissible. This result also allows to simplify the control synthesis as it can be shown that a controller making \( e_{oa} \) vanish is given by [6]:

\[
\dot{q} = \dot{q}_{oa} = -\lambda_{oa} e_{oa}
\]

where \( \lambda_{oa} \) is a positive scalar or a positive definite matrix.
C. The obstacle avoidance control

The avoidance strategy is based on the proximetric data. From these data, we compute a set of values characterizing locally any obstacle located at a distance inferior to $d_+$ (see figure 4). We obtain a couple $(\delta_{av}, \alpha)$, where $\delta_{av}$ is the signed distance between $M$ and the closest point $Q$ on the obstacle, and $\alpha$ is the angle between the tangent to the obstacle at $Q$ and the robot direction. Note that there exists two angles $\alpha$ 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 $\alpha$ so that the robot moves around the obstacle in the direction given by the pan-platform.

Consider figure 4. Around each obstacle, three envelopes are defined. The first one $\xi_+$ located at a distance $d_+ > 0$ surrounds the zone inside which the obstacle is detected by the robot. For the problem to be well stated, the distance between two obstacles is assumed to be greater than $2d_+$ to prevent the robot from considering several obstacles simultaneously. The second one $\xi_0$, located at a lower distance $d_0 > 0$ constitutes the virtual path along which the reference point $M$ will move around the obstacle. The last one $\xi_-$ 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 [16], we define a mobile frame on $\xi_0$ whose origin $Q'$ is the orthogonal projection of $M$. During obstacle avoidance, the robot linear velocity is supposed to be kept constant. Let $\delta = \delta_{av} - d_0$ be the signed distance between $M$ and $Q'$. With respect to the moving frame, the dynamics of the error terms $(\delta, \alpha)$ is described by the following system:

$$\begin{cases} \dot{\delta} = v \sin \alpha \\ \dot{\alpha} = \omega - \nu \chi \cos \alpha \end{cases} \quad \text{with} \quad \chi = \frac{\pi}{1 + \frac{R}{\delta} \delta} \quad (14)$$

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 $\omega$ allowing to steer the pair $(\delta, \alpha)$ 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 $\delta$ and $\alpha$ vanish while insuring $v \neq 0$. We propose the following redundant task function $e_{av}$:

$$e_{av} = \left( \frac{l - \nu \chi t}{\delta + k\alpha} \right) \quad (15)$$

where $l$ is the curvilinear abscissa of point $M$ and $k$ a positive gain to be fixed. The first component of $e_{av}$ allows to regulate the linear velocity of the mobile base to a nonzero constant value $v_r$. In this way, the linear velocity never vanishes, guaranteeing that the control problem is well stated and that the robot will not remain stuck on the security envelope $\xi_0$ during the obstacle avoidance. The second component of $e_{av}$ can be seen as a sliding variable whose regulation to zero makes both $\delta$ and $\alpha$ vanish (see [17] for a detailed proof). Therefore, the regulation to zero of $e_{av}$ guarantees that the robot follows the security envelope $\xi_0$, 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 perform a secondary objective while the obstacle avoidance is realized. We propose to model this objective to avoid target loss and occlusions at best and we define the corresponding cost function by $h_{occ} = \frac{1}{d_{occ}}$. This criterion can be seen as a potential function allowing to avoid both occlusions and target loss as $d_{occ}$ is defined with respect to the image borders when no occluding object lies in the image plane. The global task function $e_{ob}$ is then given by:

$$e_{ob} = W_{av}^+ e_{av} + \beta_{ob} (I - W_{av}^+ W_{av}) g_{occ} \quad (16)$$

where $\beta_{ob}$ is a positive scalar. We deduce $g_{occ}$ and $W_{av}$ by differentiating equations (10) and (15):

$$g_{occ} = \frac{\partial h_{occ}}{\partial q} = -\frac{1}{d_{occ}^2} \frac{\partial d_{occ}}{\partial q} = -\frac{e_{occ}}{d_{occ}^2} \left( \frac{\partial Y_\alpha}{\partial q} - \frac{\partial Y_\beta}{\partial q} \right) \quad (17)$$

$$W_{av} = \frac{\partial e_{av}}{\partial q} = \begin{pmatrix} \sin \alpha - k\chi \cos \alpha & -k \\ 0 & 0 \end{pmatrix}$$

where $\frac{\partial Y_\alpha}{\partial q}$ and $\frac{\partial Y_\beta}{\partial q}$ are defined by relation (12). Following the redundant task function formalism, a controller making $e_{ob}$ vanish is given by:

$$\dot{q} = \dot{\beta}_{ob} = -\lambda_{ob} e_{ob} \quad (17)$$

where $\lambda_{ob}$ 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 [18], or benefiting from the redundant task function formalism to stack elementary tasks and design control laws guaranteeing smooth transitions [19]. The second class of tasks sequencing techniques relies on convex combinations between the successive task functions [7][14] or the successive controllers [12][13]. 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 to drive the robot the best way depending on the environment. To this aim, we introduce two parameters $\mu_{ob}$ and $\mu_{ob} \in [0, 1]$ depending on the risk of occlusion and of collision as follows:

$$\dot{q} = \dot{\beta}_{ob} = -\lambda_{ob} e_{ob} \quad (17)$$

$^1$ $v_r$ must be chosen small enough to let the robot sufficiently slow down to avoid collision when entering the critical zone.
If the occluding object $O$ lies outside the region defined by $\Xi_0$ or is not in the image and if there is no obstacle in the robot vicinity, $\mu_{oa}$ and $\mu_{ob}$ are fixed to 0. Only $\dot{q}_{oa}$ must be sent to the robot in this case.

If the visual features enter the zone delimited by $\Xi_0$, the danger of occlusion becomes higher and $\mu_{oa}$ progressively increases to reach 1 when they cross $\Xi_\mu$. If the visual features naturally leave the critical zone defined by $\Xi_0$, $\mu_{oa}$ goes back to 0 without having reached 1. On the contrary, if $\Xi_\mu$ is crossed, $\mu_{oa}$ is fixed and maintained to 1 until the object $O$ leaves the image ($d_{bord} = 0$) or at least goes out the critical zone ($d_{occ} \geq D_0$). When one of these two events occurs, $\mu_{oa}$ is progressively reduced from 1 to 0 and vanishes once the visual features cross $\Xi_+$. Following this reasoning, $\mu_{oa}$ depends on the distance between the occluding object and the visual features in the image, namely $d_{occ}$. Let OCLUD be the flag indicating that $\mu_{oa}$ has reached its maximal value and $D_{leave}$ the value of the distance $d_{occ}$ for which the leaving condition has been fulfilled. We propose the following expression:

$$
\begin{align*}
\mu_{oa} &= 0 \quad \text{if } d_{occ} > D_0 \text{ and OCLUD } = 0 \\
\mu_{oa} &= \frac{d_{occ} - D_0}{D_0 - D_0} \quad \text{if } d_{occ} \in [D_-, D_0] \text{ and OCLUD } = 0 \\
\mu_{oa} &= \frac{D_{leave} - D_+}{d_{bord} - D_{leave}} \quad \text{if } d_{occ} \in [D_{leave}, D_+] \\
\mu_{oa} &= 1 \quad \text{and } (d_{bord} = 0 \text{ or } d_{occ} \geq D_0)
\end{align*}
$$

If the mobile base enters the zone surrounded by $\xi_0$ ($d_{av} < d_0$), the danger of collision rises and $\mu_{ob}$ is continuously increased from 0 to reach 1 when $d_{av} \leq d_-$. If $\xi_\mu$ is never crossed, $\mu_{ob}$ is brought back to 0, once $d \geq d_0$. If $\mu_{ob}$ 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 $\dot{q}_{ob}$ is applied to the vehicle once $\mu_{ob}$ 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 $\xi_0$ and follows it until the condition to leave is fulfilled. This event occurs when the camera and the mobile base have the same direction ($\theta = \theta_{pd}$). A flag LEAVE is then positioned to 1 and $\mu_{ob}$ is decreased to vanish on $\xi_\mu$. Therefore, $\mu_{ob}$ depends on distance $d_{av}$ as follows:

$$
\begin{align*}
\mu_{ob} &= 0 \quad \text{if } d_{av} > d_0 \text{ and AVOID } = 0 \\
\mu_{ob} &= \frac{d_{av} - d_0}{d_0 - d_0} \quad \text{if } d_{av} \in [d_- - d_0] \text{ and AVOID } = 0 \\
\mu_{ob} &= \frac{d_{av} - d_0}{d_0 - d_0} \quad \text{if } d_{av} \in [d_0, d_+] \text{ and LEAVE } = 1 \\
\mu_{ob} &= 1 \quad \text{otherwise}
\end{align*}
$$

where $d_\mu$ is defined by the distance $d_{av}$ when $\text{LEAVE} = 1$.

This reasoning is summarized on table I. Using this table and recalling that $\dot{q}_{vs}$, $\dot{q}_{oa}$ and $\dot{q}_{ob}$ are given by equations (7), (13) and (17), we propose the following global controller:

$$
\dot{q} = (1 - \mu_{oa})(1 - \mu_{ob})\dot{q}_{vs} + (1 - \mu_{ob})\mu_{oa}\dot{q}_{oa} + \mu_{ob}\dot{q}_{ob}
$$

### TABLE I

<table>
<thead>
<tr>
<th>$\mu_{oa}$</th>
<th>$\mu_{ob}$</th>
<th>$d_{av}$</th>
<th>$\dot{q}_{vs}$</th>
<th>$\dot{q}_{oa}$</th>
<th>$\dot{q}_{ob}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mu_{oa} = 0$</td>
<td>$\mu_{ob} = 0$</td>
<td>$d_{av} &lt; d_0$</td>
<td>$\dot{q}_{vs}$</td>
<td>$\dot{q}_{oa}$</td>
<td>$\dot{q}_{ob}$</td>
</tr>
<tr>
<td>$\mu_{oa} \in [0, 1]$</td>
<td>$\mu_{ob} \in [0, 1]$</td>
<td>$d_{av} \in [d_-, d_0]$</td>
<td>$\dot{q}_{oa}$</td>
<td>$\dot{q}_{oa}$</td>
<td>$\dot{q}_{ob}$</td>
</tr>
<tr>
<td>$\mu_{oa} = 1$</td>
<td>$\mu_{ob} = 1$</td>
<td>$d_{av} \in [d_0, d_+]$</td>
<td>$\dot{q}_{oa}$</td>
<td>$\dot{q}_{ob}$</td>
<td>$\dot{q}_{ob}$</td>
</tr>
</tbody>
</table>

### Remark 1:

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 we consider two different controllers $\dot{q}_{oa}$ and $\dot{q}_{ob}$ depending on the occlusion occurs far from the obstacle or close to it.

### IV. Simulation Results

We have simulated a mission whose objective is to position the camera in front of a given target. 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, 60 and 75 pixels, and $d_\mu$, $d_0$ and $d_\mu$ to 0.7m, 0.55m, and 0.45m. The sampling period is the same as on our real robot, that is $T_s = 150$ms. The obtained results are presented on figures 5, 6 and 7. As shown on these figures, the task is correctly performed as occlusions and collisions never occur. At the beginning of the task, there is no risk of occlusion or collision, the robot is only controlled by $\dot{q}_{vs}$ and starts converging towards the target. When the vehicle enters the vicinity of the first encountered obstacle, $\dot{q}_{ob}$ is progressively increased (see figure 6), and the robot first follows $\xi_0$ before being brought back on $\xi_\mu$ once the leaving condition has been fulfilled. During this phase, $\mu_{oa}$ remains equal to 0 as there is no risk of occlusion. When $\xi_\mu$ is crossed, $\dot{q}_{ob}$ vanishes and the robot executes once again the nominal vision-based task. However, the second obstacle induces a risk of both collision and occlusion. $\mu_{oa}$ and $\mu_{ob}$ are then continuously increased to reach 1 and the sole controller $\dot{q}_{ob}$ is used to guarantee a safe motion for the robot. Therefore, the vehicle is controlled to follow $\xi_0$ while occlusions are avoided at best. When the leaving condition is obtained, $\dot{q}_{ob}$ is rapidly decreased while $\mu_{oa}$ has already vanished because the avoidance motion has made the obstacle leave the image plane. Once again, the robot starts converging towards the target. However, the target and the obstacle positions have been chosen to insure that this motion brings the vehicle back towards the obstacle. As a consequence, instead of vanishing, $\dot{q}_{ob}$ rises again, making the robot avoid the obstacle. Thanks to the avoidance motion, the vehicle leaves progressively the obstacle vicinity and $\dot{q}_{ob}$ vanishes. The sole visual servoing controller is then applied to the robot and the camera finally reaches its desired position, realizing perfectly the task.

### V. Conclusion

The proposed sensor-based controller allows a mobile robot to perform safely a vision-based task in a cluttered environment.
environment. The method relies on the switch between different controllers depending on the risk of collision and occlusion. The obtained results are quite satisfying and these control laws are currently being experimented on our mobile robots. 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. A dynamical sequence of the controllers could also be interesting to provide better theoric feasibility conditions.

REFERENCES