Visual servoing with safe interaction using Image Moments

Hamid Sadeghian*, Luigi Villani†, Zahra Kamranian*, and Abbas Karami‡
* Engineering Department, University of Isfahan, Isfahan, Iran
† Department of Electrical Engineering and Information Technology, University of Naples Federico II, Italy
‡ Mechanical Engineering Department, Isfahan University of Technology, Isfahan, Iran

***Abstract*—The problem of image based visual servoing for robots working in a dynamic environment is addressed in this paper. It is assumed that the environment is observed by depth sensors which allow to measure the distance between any moving obstacle and the robot. The main idea is to control suitable image moments during the interaction phase to relax a certain number of robot's degrees of freedom. If an obstacle approaches the robot, the main visual servoing task is attenuated or completely abandoned while the image features are kept in the camera field of view by controlling the image moments. Fuzzy rules are used to set the reference values for the controller. Beside that, the relaxed redundancy of the robot is exploited to avoid collisions as well. After removing the risk of collision, the main visual servoing task is resumed. The effectiveness of the algorithm is shown by several case studies on a KUKA LWR 4 robot arm.**

***Index Terms*—Visual Servoing, Interaction Control.**

I. Introduction

Classical image-based visual servoing (IBVS) is aimed at controlling the end effector of a robot carrying a camera in such a way that some measurable quantities extracted from the image captured by the camera, denoted as image features, attain desired values. This allows, for example, a robotic hand to be aligned with an object to grasp, like the handle of a drawer or a door whose position is uncertain or may change, especially in anthropic environments.

In dynamic environment, it is important for the robot controller to ensure suitable reaction capabilities beside main task performance. In this regard, two main strategy is usually adopted; avoiding undesired collisions, or handling the physical interaction. The latter case is implemented by increasing the compliance of the robot and using suitable observer to compensate the external forces exerted on the robot body, [1], [2]. However, it relies on fast control loop which in the case of visual task with limited camera sampling rate is not attainable.

The selection of the visual features is very important in 2D visual servoing and affects directly on interaction matrix structure. A good choice of the features allows to obtain a large full rank domain for the interaction matrix and linear relationship between image plan and 3D camera trajectory. A best way to ensure this condition is to associate each camera degree of freedom with only one visual feature. However, such a condition seems ideal and only partial decoupling can be obtained under some conditions using image moments and invariants [3], [4].

The problem of path planning for IBVS of robotic arms with the aim of extending the robustness of classical control techniques to include image constraints (field of view limits, occlusions) and physical constraints (joint limits, singularities in robot Jacobian, collisions with obstacles, self-collisions) is addressed, e.g., in [5] and references therein. The main idea is to plan and generate off-line feasible image feature trajectories while accounting for the constraints, and then to control the robot along the planned trajectories. Generally, this results in a more robust servoing process with respect to violation of image and physical constraints. However, these techniques are not effective in the presence of moving obstacles (like humans), whose position is not known a priori.

On the other hand, research efforts have been devoted to incorporate the above image and physical constraints into a reactive visual servoing loop. Earlier works include partitioned and switched strategies (see, e.g. [6]). In contrast with partitioned strategies, where certain degrees of freedom (DOFs) are controlled via IBVS, while others are controlled via position based visual servoing (PBVS), switched strategies consist of a set of visual servo controllers along with a switching rule. Partitioned techniques have advantage of both IBVS and PBVS techniques in avoiding some of the aforementioned constraints, while switched strategies enlarge the stability region of classical visual servoing techniques by switching between a set of unstable controllers to make the overall system stable.

An interesting approach to control the robot in the presence of constraints is the concept of task sequencing [7], [8], which exploits the functional kinematic redundancy of the robot. The key idea is to divide the task into several sub-tasks which in the presence of constraints are deactivated in sequence based on the priority level of the sub-tasks. Further works exploiting redundancy in visual servoing tasks are [9], [10].

Although reactive control techniques are more suitable than off-line planning techniques to ensure a safe coexistence of robots and humans, reactive approaches purposely designed to address this problem have not been developed so far. Only some heuristic techniques can be found in the literature [11].

In this respect, a promising direction of research is that of combining visual servoing approaches which guarantee that the visual features remain in the field of view of the camera

This work was partially funded by the European Community, within the FP7 ICT-287513 SAPHARI project. Corresponding author: H. Sadeghian (email: h.sadeghian@ eng.ui.ac.ir).

[12], [13], [14], [15] with collision avoidance techniques exploiting redundancy [16].

In this paper an image-based visual servoing algorithm with collision avoidance capabilities for dynamic obstacles is proposed. An eye-in-hand camera is used to extract image features which are used to control the end-effector motion. Moreover, the distance between humans and (any part of) the robot is measured using other sensors (e.g., Microsoft Kinect).

It is assumed that the a velocity controlled robot is executing a visual servoing task and the task can be partially relaxed when obstacle is too close or come in contact with the robot. To this aim, six combination of moments are selected as the visual features to control six degree of freedom of the camera such that the corresponding interaction matrix has maximum decoupling structure. During the interaction phase, only the visual features related to the centroid and the variance of the image is regulated to keep the features in the camera field of view (FOV). Therefore, the dimension of the servoing task decreases in a continuous manner and the released redundancy of the robot is further used to keep the robot at a safe distance from obstacle(s) through a repulsive action. Moreover, fuzzy rules are designed to set the references for the image moments, depending on some features of the expected collision.

II. Preliminaries

A. Camera Model and Image Jacobian

Consider a camera with coordinate frame $\mathcal{F}_c$. The camera views a collection of N visual features

$$s(t) = [s_1^T, s_2^T, ..., s_N^T]^T. \tag{1}$$

If the features are static in an inertial frame, the features' velocity in the image plane is given as a function of the camera velocity v_c by the relationship

$$\dot{s} = Lv_c, \tag{2}$$

where $L(t)$ is the interaction matrix and is constructed by concatenating a set of N sub-matrices $L_k(s_k)$. For any point $(x_k(t), y_k(t))$ in the image plane, associated with 3D point with depth Z_k in $\mathcal{F}_c$, the interaction matrix is given by

$$L_k = \begin{bmatrix} -1/Z_k & 0 & x_k/Z_k & -x_k y_k & -1-x_k^2 & y_k \\ 0 & -1/Z_k & y_k/Z_k & 1+y_k^2 & -x_k y_k & -x_k \end{bmatrix}. \tag{3}$$

The interaction matrix related to almost any geometrical primitives can be obtained as well.

To follow a desired trajectory $s_d(t)$ in the image plane, the control scheme usually is designed to ensure an exponential decoupled decrease of the visual features. For instance for eye-in-hand system observing an static object, the command camera velocity is given as

$$v_{c-com} = \widehat{L}^{+}(\dot{s}_d + \gamma(s_d - s)), \tag{4}$$

where $\widehat{L}^{+}(t)$ denotes the left pseudo inverse of the *estimated* matrix $\widehat{L}(t)$ and γ is a positive gain. Note that the depth of object points which is generally unknown in 2D visual servoing is necessary for calculation of $L(t)$. However, this value can be estimated by suitable observer during camera motion (for instance, [17]) or simply set as the desired depth value. The non-linearity in interaction matrix explains the unpredictable behavior of the robot in 3D space even for a small displacement in image plan.

B. Image Moments and Moment Invariants

The image moments is very useful for providing an intuitive and meaningful representation of the object. The moments $m_{ij}(t)$ of an object with a projected image area $\mathcal{R}(t)$ in the image plane are defined by

$$m_{ij}(t) = \iint_{\mathcal{R}(t)} I(x,y)\, x^i y^j dxdy, \tag{5}$$

where $I(x, y)$ is the intensity level of image point with coordinates x and y. Usually in computer vision dealing with image motion analysis the intensity level is assumed to be constant. the centered moments μ_{ij} are computed with respect to the object centroid (x_m, y_m) and is given by

$$\mu_{ij} = \iint_{R} (x - x_m)^i (y - y_m)^j dxdy, \tag{6}$$

Similarly for a discrete set of N image points, the moments m_{ij} and centered moments μ_{ij} are defined by

$$m_{ij} = \sum_{k=1}^{N} (x_k)^i (y_k)^j \tag{7}$$

and

$$\mu_{ij} = \sum_{k=1}^{N} (x_k - x_m)^i (y_k - y_m)^j \tag{8}$$

respectively. The centered moments defined either from (6) or (8) are known to be invariant to 2D translational motion if the image plan is parallel to the object. In the literature, many methods have been presented to derive moment invariants to scale and rotations. These invariants can be found in [4] and the references therein.

As for the classical geometrical features, a linear link can be expressed in the form of

$$\dot{m}_{ij} = L_{m_{ij}} v_c, \tag{9}$$

between the time variation of moments and the relative camera- object velocity components. $L_{m_{ij}}$ is the related interaction matrix in the form of

$$L_{m_{ij}} = [m_{vx}\, m_{vy}\, m_{vz}\, m_{\omega x}\, m_{\omega y}\, m_{\omega z}]. \tag{10}$$

The analytical form of interaction matrix related to any image moments of order $i + j$ can be given under some conditions [3]. The moments based on the above definition and their combinations can be effectively used in visual servoing as the features of the image, provided that it is well-defined and differentiable. For instance by the selection $s = (x_m, y_m, m_{00}, \mu_{02}, \mu_{20}, \alpha)$ with $\alpha = \frac{1}{2}\tan^{-1}(\frac{2\mu_{11}}{\mu_{20}-\mu_{02}})$ as the object orientation, we may hope that the six DOF of the camera can be controlled effectively. However, the associated interaction matrix is not full rank for a symmetric object appearing in the image plan. For non-symmetric objects the interaction matrix is of full rank 6 but with high condition number!

III. Control Development

Let us assume that a (redundant) robot is performing a visual servoing task using an eye in hand camera. Depth sensors (e.g. Kinect) are employed to measure the distance of robot body and end effector from any obstacle or humans. The visual servoing task is that to follow a desired trajectory $s_d(t)$ in the image plane (IBVS). To this aim, the control law (4) can be adopted for the robot end effector.

A. Keeping the Image Features in the FOV

When an obstacle comes near the robot, collisions must be avoided. In some particular cases, the redundant degrees of freedom of the robot allows to avoid the collision by reconfiguring the robot through internal motions while keeping the servoing task. However, this is not always possible and, to avoid the obstacle, the task must be abandoned and cannot be easily resumed.

The solution proposed here for obstacle avoidance is that of relaxing the servoing task partially, while, keeping the image features in the camera field of view by using a reduced number of degrees of freedom. This allows to have more degrees of freedom at disposal for obstacle avoidance and to recover the servoing task once the collision is avoided. To this end, the moments based features of order up to 2, including the centroid of the object's image and the centered moments μ_{02} and μ_{20} are regulated during the interaction phase. Note that in comparison with the visual servoing task that is usually a 6 DOF task, this task involves 3 or 4 DOF and thus relaxes more degrees of freedom which may be exploited for obstacle avoidance.

The coordinate of the centroid point of a set of visual features in the image plane is given by

$$x_m = m_{10}/m_{00}, y_m = m_{01}/m_{00}. \tag{11}$$

For the case of N point features, the time derivative of s_m is given by

$$\dot{s}_m = J_m \dot{s}, \tag{12}$$

where J_m is a $(2 \times 2N)$ matrix computed as

$$J_m = \frac{1}{N}[I_2, ..., I_2]. \tag{13}$$

where I is identity matrix. In this case also the moments μ_{02} and μ_{20} represent the *variance* of the feature points in the current image;

$$s_v = \frac{1}{N}\sum_{k=1}^{N}\begin{bmatrix}(x_k - x_m)^2\\ (y_k - y_m)^2\end{bmatrix} = \frac{1}{m_{00}}\begin{bmatrix}\mu_{20}\\ \mu_{02}\end{bmatrix} \tag{14}$$

The Jacobian of the variance task function, $J_v(t)$ is a $(2\times 2N)$ matrix given by

$$J_v = \frac{2}{N}\begin{bmatrix} x_1 - x_m & 0 & ... & ... & x_N - x_m & 0\\ 0 & y_1 - y_m & ... & ... & 0 & y_N - y_m\end{bmatrix} \tag{15}$$

and relates the time derivative of s_m to $\dot{s}$ through

$$\dot{s}_v = J_v(t)\dot{s}. \tag{16}$$

Note that $J_v(t)$ is singular when all the feature points are collinear. Current study assumes that J_v remains full rank. It is clear that the associated interaction matrix of these moments are obtained through multiplication of the above Jacobian matrices by the interaction matrix of feature points. It can also be shown that the these task Jacobians are completely independent and thus the augmentation

$$J_{mv} = \begin{bmatrix}J_m\\ J_v\end{bmatrix}, \tag{17}$$

does not introduce any singularities [14].

B. Selection of the Visual Features

The selection of feature points in 2D visual servoing is an important issues for both the discrete and continues features. Indeed, the unpredictable behavior of the robot trajectory in 3D space is because of the high nonlinear terms in interaction matrix. Thus, it is important to determine visual features to minimize this non-linearity. Moreover, the well-conditioning of the interaction matrix is also important for the convergence of servoing scheme. The best way to ensure that is to associate each camera DOF with only one visual feature to obtain a decoupled behavior. On the other hand, since the depth information is necessary in interaction matrix and usually an estimation of interaction matrix is used in schemes, it is usually appealing to choose feature points in such a way that the interaction matrix to be almost constant around the desire pose, especially when the object and image planes are parallel at the desired position. The coordinates x_m and y_m of the object image center, the orientation of the object α and the area m_{00} of the object image is of particular interest since they have direct and intuitive link with 3D motion. Indeed the coordinate x_m and y_m are closely related to v_x and ω_y, and v_y and ω_x respectively. Furthermore, The area and the orientation of the object in the image scene is affected mainly by v_z and ω_z. On the other hand, based on the results of previous section, the μ_{20} and μ_{02} affects on the variance of the features and can be interesting for the aim of keeping the features in FOV during any interaction.

In order to decouple ω_y from v_x, and ω_x from v_y two combination of moments of order 3 has been proposed in [3]. These combinations are obtained based on the following invariants to translational motions and rotational motion around the optical axis, since the visual features to control these motions is already available.

$$\begin{aligned} I_1 &= (\mu_{20} - \mu_{02})^2 + (2\mu_{11})^2,\\ I_2 &= (\mu_{03} - 3\mu_{21})^2 + (\mu_{30} - 3\mu_{12})^2,\\ I_3 &= \mu_{20} - \mu_{02}. \end{aligned} \tag{18}$$

The combinations are different for the case of symmetric and non-symmetric observed objects. Here, we recall these supplementary features for non-symmetric case which is given by

$$\begin{aligned} p_x &= I_1/I_3^2,\\ p_y &= m_{00}I_2/I_3^3. \end{aligned} \tag{19}$$

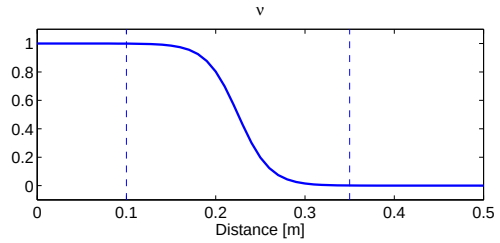

Fig. 1: The parameter ν for numerical values $\rho_1 = 0.25$, $\rho_2 = 0.1$, $\alpha = 7$)

For symmetric object this combination is more complex and is shown by s_x and s_y. The related equations can be found in [3]. Indeed, when the object is parallel to the image plane, the interaction matrix related to these features has only non zero value for the columns associated with ω_x and ω_y. However, as we will see in the experimental simulations, in other configurations the values of the other columns are not much and the results is satisfactory even when the object is not parallel to the image plane during servoing task.

C. Obstacle Avoidance

The distance vector between the obstacle $\mathcal{O}$ and any point $\mathcal{P}$ on the robot $D(P,O)$ is assumed to be known by depth sensor or laser sensor. As soon as the distance $||D(P,O)||$ is found to be less than $||D(P,O)||_{\min} = \rho_1$, a repulsive vector is produced and used to modify on-line the current trajectory. The measure of proximity of the obstacle to the robot is defined by a smooth function,

$$\nu(P,O) = \frac{1}{1 + e^{((||D(P,O)|| - \rho_2)(2/\rho_1) - 1)\alpha}}, \quad 0 < \nu < 1 \tag{20}$$

which is equal to one when $||D(P,O)||_{\min} < \rho_2$ and is zero for $||D(P,O)||_{\min} > \rho_1$ for appropriate shape factor α. An example of this profile is depicted in Fig. 1 for some numerical values.

A simple and effective repulsive vector is given by

$$V_{rep} = \nu_1 V_{\max} \frac{D(P,O)}{||D(P,O)||}, \tag{21}$$

where, ν_1 is given by (20) with appropriate parameters. This vector have the same direction as $D(P,O)$ with $V_{\max}$ as the maximum admissible speed in task space. Note that this repulsive action is given only based on the distance between obstacle $\mathcal{O}$ and control point $\mathcal{P}$ on the robot body. The factor ν_1 allows to further modulate the repulsive action depending on the distance of the obstacle from the robot body. It is more wise to include also the direction and velocity of the obstacle which can be extracted by observing the time variation of repulsive vector, i.e. $\dot{V}_{rep}$. These information can be used, for instance, to escape the collision by moving perpendicular to the direction of obstacle trajectory. The algorithm is called Pivot algorithm [16] and modifies the direction of the repulsive vector according to its variation. The repulsive vector velocity is then reflected in the joint space through the transpose of the Jacobian J_P associated with the point $\mathcal{P}$, i.e.

$$\dot{q}_{rep} = J_P^T V_{rep}. \tag{22}$$

D. Combined Action

Now, assume that a robot is doing a visual task and the command velocity for the end effector is calculated by (4). As soon as an obstacle approaches the robot, the repulsive vector $\dot{q}_{rep}$ is used to update the trajectory. However, applying this repulsive action definitely affects the assigned visual servoing task for the end effector and may cause the robot to lose the feature points during obstacle avoidance.

In order to keep visual features in the field of view during the repulsive action, the combination $s = (x_m, y_m, \mu_{02}, \mu_{20}, p_x, p_y)^T$ is selected as the features vector. During the interaction phase, the first four features which regulate the centroid and the variance of the image is kept and the last *two* features are released. This increase the dimension of the null-space which is exploited to handle the interaction. However, for symmetric object it can be shown that the the interaction matrix associated with the first four feature is singular when the camera plane and the objects are parallel. That's why it is proposed to use $s' = (x_m, y_m, \mu_{02} + \mu_{20}, p_x, p_y, \alpha)^T$ for symmetric objects. In this case, the last *three* features are released during the interaction and provides three dimensional redundant space. Based of above selections the feature vector and the interaction matrix is partitioned as

$$s = \begin{bmatrix} s_1 \\ s_2 \end{bmatrix}, \widehat{L} = \begin{bmatrix} \widehat{L}_1 \\ \widehat{L}_2 \end{bmatrix}. \tag{23}$$

The above choices of s or s' present maximum decoupling of interaction matrix as mentioned in section III.B. Finally, for a robot with low level controller in the joint space, the command velocity is given by

$$\dot{q} = (\widehat{L}J)^{\dagger} \begin{bmatrix} \dot{s}_{com-1} \\ (1-\nu_2)\dot{s}_{com-2} + \nu_2(\widehat{L}_2 J)(\widehat{L}_1 J)^{\dagger}\dot{s}_{com-1} \end{bmatrix} + \nu_2 N \dot{q}_{rep} \tag{24}$$

where, J is the robot's Jacobian matrix, ν_2 is given by (20) with appropriate parameters. $\dot{s}_{com-1}$ and $\dot{s}_{com-2}$ are command feature velocities related to the first and second partition of s or s'. These commands can be given by

$$\begin{aligned} \dot{s}_{com-1} &= \dot{s}_{des-1} + \gamma_1(s_{des-1} - s_1), \\ \dot{s}_{com-2} &= \dot{s}_{des-2} + \gamma_2(s_{des-2} - s_2), \end{aligned} \tag{25}$$

with γ_1 and γ_2 as positive gains. In (24), N is the projection matrix given by

$$N = I - (\widehat{L}_1 J)^{\dagger}(\widehat{L}_1 J). \tag{26}$$

Note that, in the absence of interaction, i.e., $\nu_2 = 0$, from (24) the command joint space velocity is reduced to

$$\dot{q} = (\widehat{L}J)^{\dagger} \begin{bmatrix} \dot{s}_{com-1} \\ \dot{s}_{com-2} \end{bmatrix} \tag{27}$$

When an obstacle comes near the robot, ν_2 increase to 1 and (24) decrease to

$$\dot{q} = (\widehat{L}_1 J)^{\dagger}\dot{s}_{com-1} + N\dot{q}_{rep}, \tag{28}$$

and control of centroid and variance with a continuous transition from full visual servoing is ensured. For more information about the continuous transition among multiple tasks and during multi-priority control [2], [18] are referred.

The interaction matrix $\widehat{L}$ can be chosen as constant matrix at the desired pose, $L_{|s=s_{des}}$, which presents excellent decoupling properties. However the following choice seems more exact;

$$\widehat{L}_s = \frac{1}{2}(L_{|s=s_{des}} + L_{|s(t)}) \tag{29}$$

In the case that the feature vectors are different from the above selection s or $s^{'}$, a similar algorithm can be used. In that case the visual servoing task is attenuated by a factor of proximity and the centroid and variance features are selected and controlled to keep the scene in FOV.

During the interaction phase, the desired trajectory for the centroid and the variance features can be intact or may be set as these value at the last capture of the visual servoing task. However, these values can be chosen wisely based on the location of the nearby obstacle as explained in the next Section.

The proposed formulation can be used for interaction on the robot body as well as on the end effector of the robot using the Jacobian matrix of the interaction point $\mathcal{P}$ which is assumed to be known by monitoring the robot scene with depth sensors. Note that the above analysis was performed assuming no occlusion.

IV. Experimental Simulations

The proposed algorithm is verified in a simulation environment consist of 7DOF KUKA LWR arm, a camera mounted on the end effector and a spherical moving obstacle which can resemble part of a human body. The observed object is considered as a rectangle and the combination $s^{'} = (x_m, y_m, \mu_{02} + \mu_{20}, p_x, p_y, \alpha)^T$ is selected as the feature vector. The desired camera pose is such that image plane is parallel to the scene, at a distance of about 25 cm from the camera optical center. At this configuration s_{des} is given by $s_{des} = (0, 0, 0.607, 0, 0, 0)$ and the corresponding interaction matrix is a square matrix obtained as

$$L_{s|s=s_{des}} = \begin{bmatrix} -4.13 & 0 & 0 & 0 & -1.11 & 0 \\ 0 & -4.13 & 0 & 1.04 & 0 & 0 \\ 0 & 0 & 5.01 & 0 & 0 & 0 \\ 0 & 0 & 0 & -0.18 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.18 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix} \tag{30}$$

which has a good decoupling structure. The initial configuration of the robot is chosen different from the desired pose of the end-effector. It is assumed that the distance between the control point $\mathcal{P}$ on the robot and the human is estimated using a depth sensor. The procedure that is used to obtain this distance is not discussed here. As mentioned above, the desired values of the centroid position and the image variances during the interaction phase are the important parameters that mainly affect the behavior of the robot. In other words, by the former the centroid of the object in the image plane is controlled and the latter affects on the scale in image plane. The desired value for the centroid is usually set to zero to keep the centroid of feature points in the center of image plane. The choice of the desired value of the variance can be more elaborating. To increase the autonomy of the algorithm during interaction a *Mamdani*-type fuzzy planning algorithm is proposed to set this desired value wisely.

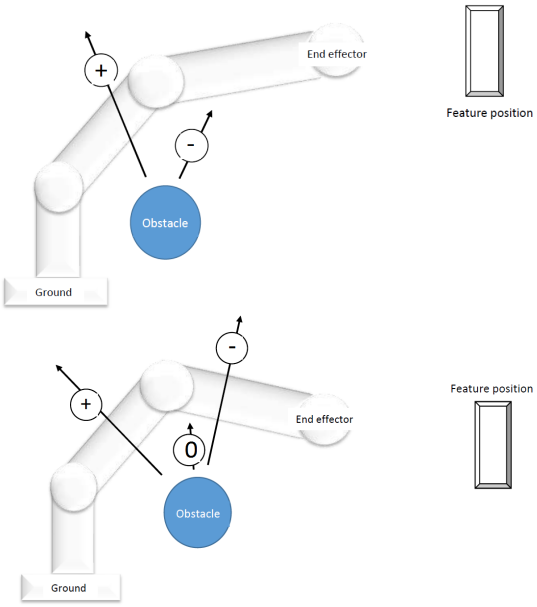

Fig. 2: Examples of typical decisions made by fuzzy planner in different situations for KUKA LWR4 manipulator during a visual servoing task. Black arrows show obstacle trajectory which is passing above or below of the link. "+" and "-" shows typical situations in which increasing or decreasing the desired variance value is helpful. "0" shows a typical condition in which changing the desired variance value is not helpful.

A. Fuzzy Based Planning of the Desired Variance

The obstacle data which are available through the depth sensors are used as the input in the fuzzy planner. Fuzzy system's rules are defined such that the robot body always has a safe distance from obstacle trajectory. Some robot-obstacle situations and the corresponding variation of the variance desired value are shown in Fig. 2 for a typical manipulation task. The Fuzzy planner changes the desired variance value based on the situations and the defined rules. When the robot is in the situation that changing the desired variance value is not useful to avoid the obstacle, it would not change the desired variance. For example, when the obstacle passes above the robot arm (Fig. 2), fuzzy planner will decrease the desired variance value. Decreasing the desired value of the variance would increases the camera distance with respect to the scene and move the robot away from the obstacle trajectory.

The fuzzy subsystem uses four different inputs in order to decide how to change the variance. The obstacle distance from the manipulator, the obstacle trajectory relative to manipulator configuration, the link exposed to the interaction and the actual robot configuration are the information required to

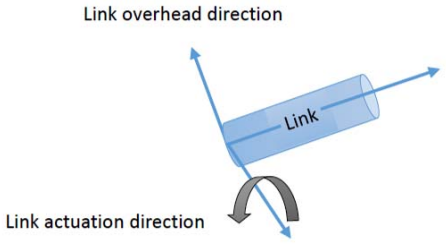

Fig. 3: link overhead direction.

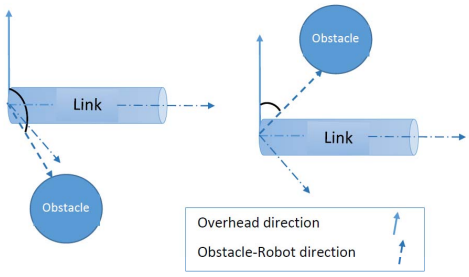

Fig. 4: Obstacle trajectory crossing below (left) and over (right) the manipulator link. Link direction and link axis are also shown.

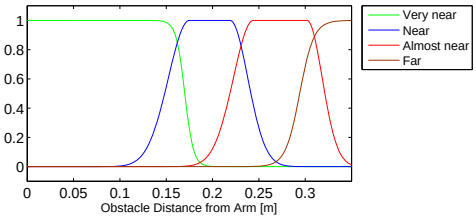

Fig. 5: Membership functions for obstacle distance from manipulator.

make a proper decision. These data are prepared using the vector connecting a moving control point $\mathcal{P}$ on the robot to the obstacle. The control point moves along the robot body repeatedly, to find the the possible position of collision.

In order to find the relative trajectory of the obstacle with respect to the links, a so called *link overhead direction* is calculated based on the cross product of the link direction and its axis (Fig. 3). This vector shows the normal vector of the plane that includes link direction and link axis. The angle between this vector and the vector connecting the control point to the obstacle is used to find whether the obstacle is passing above or below the link (Fig. 4). Finally, based on a the current robot configurations the desired variance is updated. The membership functions are shown in Fig. 5, Fig. 6 and Fig. 7. The output membership functions are illustrated in Fig. 8 as well.

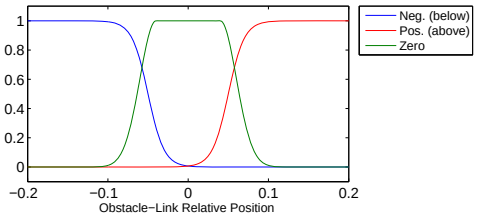

Fig. 6: Membership functions in order to find the relative obstacle-link position; It describes if the obstacle is passing below or above the link.

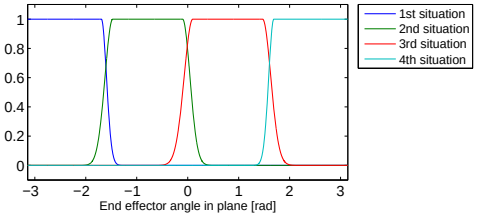

Fig. 7: Membership functions for manipulator configurations, 1st and 4th situations are corresponding to the two typical situations shown in Fig. 2 from top to the bottom, respectively. Other situations are omitted for brevity.

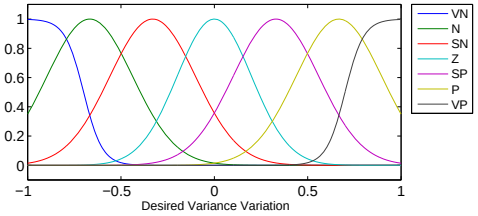

Fig. 8: Output membership functions. "VN" and "VP" stand for variance negative and positive variation, respectively. Other situations are between these two extremes.

B. Case Studies

The simulations are performed for two cases. In the first case the obstacle is passing close to the robot elbow during visual servoing task. The snapshots of this experiment are illustrated in Fig. 9. The parameters in the reference velocity (24) are chosen as follows. For the visual servoing task the parameters γ_1 and γ_2 in (25) are chosen as $\gamma_1 = \gamma_2 = 3I_3$. Moreover, $V_{\max}$ in (21) is set $V_{\max} = 5$. The distance between the risky control point $\mathcal{P}$ on the robot body and the obstacle as well as the parameters ν_1 and ν_2 are shown in Fig. 10. As soon as the obstacle moves near to the robot, the parameter ν_2 increases rapidly to 1 and the visual servoing task is partially released. The desired centroid values are set to its last values before the interaction phase, which in this experiment are zero. The desired variance is devised by the fuzzy planner as shown in Fig. 11. In fact, as it was mentioned before, the desired centroid values control the offset of the image center in camera frame and the desired variance value affects mainly on the distance of the camera frame with respect to image scene. Thus, for higher values of the desired variance, higher retraction of the camera is achieved during the interaction. The time history of these values are illustrated in Fig. 11. Furthermore, the three dimensional position of the end-effector

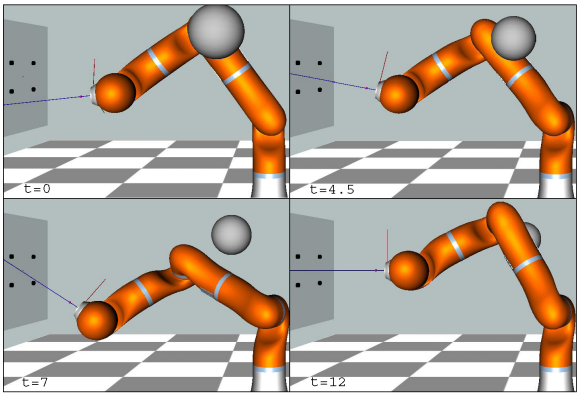

Fig. 9: Case 1: Snapshot of the system during interaction on the robot body.

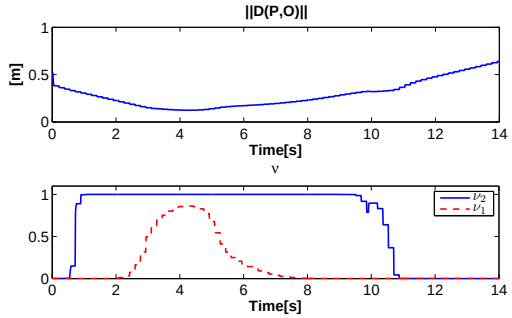

Fig. 10: Case 1: the distance between the risky control point on the robot body and the obstacle (Top), the parameter ν_1 ($\rho_1 = 0.3$, $\rho_2 = 0.05$, $\alpha = 7$) (Bottom).

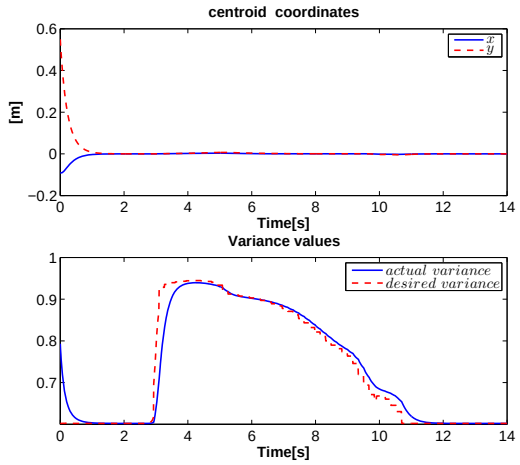

Fig. 11: Case 1: the centroid (Top) and the variance (Bottom) values of the visual features in the image plane.

as well as the image plane trajectory are shown in Fig. 12.

In the second case study the interaction happens on the end effector. Here, the visual servoing task is the same as in the first case study but the obstacle passes near to the end effector assuming no occlusion. The main parameters of the velocity controller have not been changed to show the effectiveness of the algorithm for different interaction points. The results are depicted in Figs. 13-16. It can be seen that the safety distance between the end effector and the obstacle is always respected (Fig. 14), and can be controlled by ν_1, ν_2 and $V_{\max}$. The retraction of the end-effector during the interaction phase as a

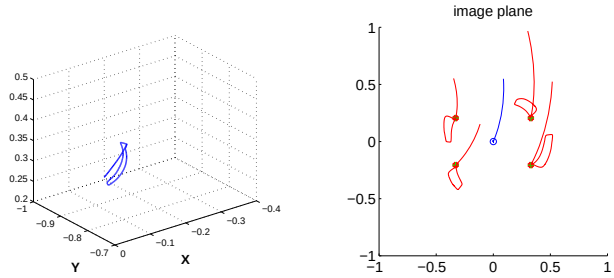

Fig. 12: Case 1: the end effector position trajectory (left), and the image plane trajectory (right).

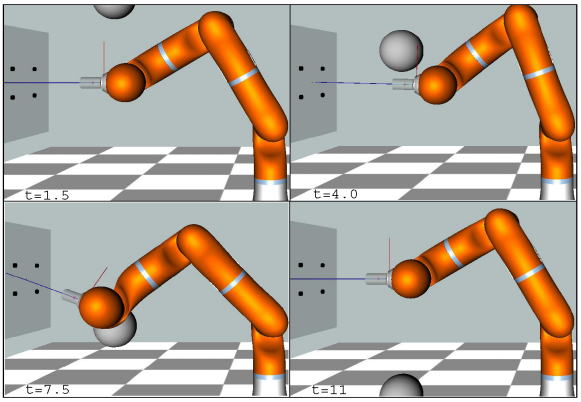

Fig. 13: Case 2: snapshot of the system during interaction on the end-effector.

result of teh variation of the variance desired value is clearly seen by comparison of the robot configurations at the times $t = 4$ and $t = 7$ in Fig. 13. The effect of V_{rep} can be seen by comparison of these snapshots with the snapshot related to $t = 7$.

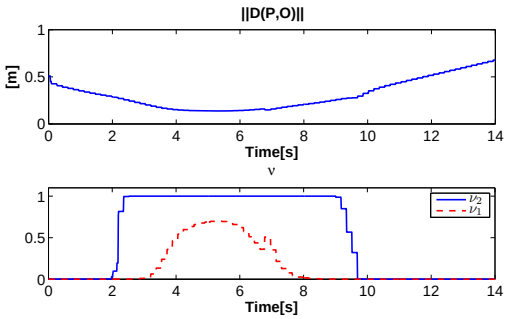

Fig. 14: Case 2: the distance between the risky control point on the end-effector and the obstacle (Top), the parameter ν_1 ($\rho_1 = 0.3$, $\rho_2 = 0.05$, $\alpha = 7$) (Bottom).

V. Conclusion

Control of the interaction during an image based visual servoing for a robot working in dynamic (human) environment was considered in the paper. The main concerns in this scenario are the performance of the main visual servoing task, keeping the visual feature in the field of view as well as a safe obstacle-robot distance. To cope with these three objectives, a special combination of the visual features extracted from the image moments including the centroid and the variance

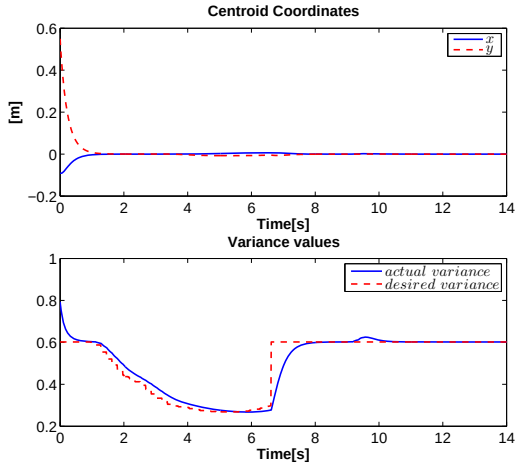

Fig. 15: Case 2: the centroid (Top) and the variance (Bottom) values of the visual features in the image plane.

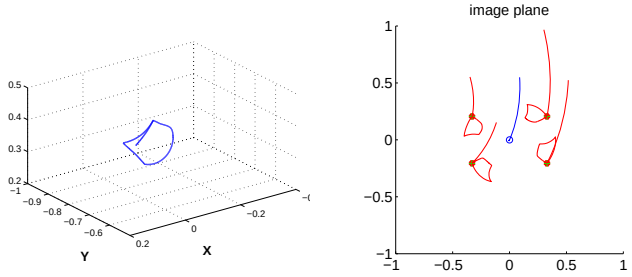

Fig. 16: Case 2: the end effector position trajectory (left), and the image plane trajectory (right).

values of the visual features is selected. During the interaction, these values are controlled to keep the visual features in the field of view and the rest of the visual features are released smoothly. Consequently, the null-space dimension increases and is exploited to effectively keep a safe distance from obstacle. In addition, a fuzzy planner was used in order to change the desired variance value such that the available degrees of freedom can be better exploited. The proposed approach was tested in several simulations on a KUKA LWR robot arm in the presence of a dynamic moving obstacle. Future work will be also devoted to study of the occlusion and physical interaction within the above algorithm.

REFERENCES

[1] H. Sadeghian, L. Villani, M. Keshmiri, and B. Siciliano, "Task-space control of robot manipulators with null-space compliance ," *IEEE Transactions on Robotics*, vol. 30, no. 2, pp. 493–506, 2014.

[2] H. Sadeghian, L. Villani, M. Keshmiri, and B. Siciliano, "Dynamic multi-priority control in redundant robotic systems," *Robatica*, vol. 13, no. 7, pp. 1155–1167, 2013.

[3] F. Chaumette, "Image Moments: A General and Useful Set of Features for Visual Servoing," *IEEE Transaction on Robotics*, vol. 20, no. 4, pp. 713–723, 2004.

[4] O. Tahri, F. Chaumette, "Point-based and Region-based Image Moments for Visual Servoing of Planar Objects," *IEEE Transaction on Robotics*, vol. 21, no. 6, pp. 1116–1127, 2005.

[5] M. Kazemi, K. Gupta, and M. Mehrandezh, "Randomized kinodynamic planning for robust visual servoing," *IEEE Transactions on Robotics*, vol. 29, no. 5, pp. 1197–1211, 2013.

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

[7] N. Mansard and F. Chaumette, "Visual Servoing Sequencing Able to Avoid Obstacles," *IEEE International Conference on Robotics and Automation*, pp. 3154–3159, 2005.

[8] N. Mansard and F. Chaumette, "Task sequencing for high-level sensor-based control," *IEEE Transactions on Robotics*, vol. 23, no. 1, pp. 60–72, 2007.

[9] A. De Luca, G. Oriolo, and P. Giordano, "Image-based visual servoing schemes for nonholonomic mobile manipulators," *Robotica*, vol. 25, no. 2, pp. 131–145, 2007.

[10] A. De Luca, M. Ferri, G. Oriolo, and P. Giordano, "Visual servoing with exploitation of redundancy: An experimental study," in *IEEE International Conference on Robotics and Automation*, pp. 3231–3237, 2008.

[11] G. Garcia, J. Corrales, J. Pomares, F. Candelas, and F. Torres, "Visual servoing path tracking for safe human-robot interaction," in *IEEE International Conference on Mechatronics*, 2009.

[12] A. Remazeilles, N. Mansard, and F. Chaumette, "A qualitative visual servoing to ensure the visibility constraint," in *IEEE International Conference on Intelligent Robots and Systems*, pp. 4297–4303, 2006.

[13] I.-J. Ha, D.-H. Park, and J.-H. Kwon, "A novel position-based visual servoing approach for robust global stability with feature points kept within the field-of-view," in *11th International Conference on Control, Automation, Robotics and Vision*, pp. 1458–1465, 2010.

[14] N. Gans, G. Hu, K. Nagarajan, and W. Dixon, "Keeping multiple moving targets in the field of view of a mobile camera," *IEEE Transactions on Robotics*, vol. 27, no. 4, pp. 822–828, 2011.

[15] O. Kermorgant and F. Chaumette, "Combining IBVS and PBVS to ensure the visibility constraint," in *IEEE International Conference on Intelligent Robots and Systems*, pp. 2849–2854, 2011.

[16] F. Flacco, T. Kröger, A. De Luca, and O. Khatib, "A depth space approach to human-robot collision avoidance," in *IEEE International Conference on Robotics and Automation*, pp. 338–345, 2012.

[17] A. De Luca, G. Oriolo, and P. Giordano, "On-line estimation of feature depth for image-based visual servoing schemes," in *IEEE International Conference on Robotics and Automation*, pp. 2823-2828, 2007.

[18] J. Lee, N. Mansard and J. Park, "Intermediate Desired Value Approach for Continuous Transition among Multiple Tasks of Robots," in *IEEE International Conference on Intelligent Robots and Systems*, pp. 1276–1282, 2011.