Visual Path Following using a Sequence of Target Images and Smooth Robot Velocities for Humanoid Navigation

Josafat Delfin[1], Hector M. Becerra[2] and Gustavo Arechavaleta[1]

***Abstract*— In this paper, we propose an approach of following a visual path for humanoid navigation. The problem consists in computing appropriate robot velocities for the humanoid walking task from the visual data shared between the current robot view and a set of target images. Two types of visual controllers are evaluated: a position-based scheme and an image-based scheme. Both of them rely on the estimation of the homography model even for non-planar scenes. We assume that the sequence of target images is given and we focus on the controllers performance. Because classical visual path following controllers generate discontinuous robot velocities, we propose a generic controller (applicable for different types of visual feedback) to alleviate this issue, which is a main contribution of the paper. The stability of such controller is addressed theoretically and verified through experiments with a NAO humanoid robot.**

I. INTRODUCTION

Important efforts have been done to integrate the visual servoing approach [1] to the problem of robot long-distance navigation. This problem has been mainly addressed for wheeled mobile robots [2]–[5]. Since the visual servoing approach is a local solution, a natural extension is the use of a sequence of target images and the formulation of a visual path following problem for navigation. This idea was introduced in [2] for wheeled mobile robots and called view-sequenced route representation. This navigation approach can be treated through any of the two classical approaches of visual servoing: a position-based scheme (PBVS), where the estimation of some 3D pose parameters is needed, or an image-based scheme (IBVS), where direct feedback of image features is used [1]. The capture of a sequence of target images is carried out from a supervised teaching phase, which generates the also called visual memory [4], [5]. An example of position-based scheme that uses the visual memory approach and 3D reconstruction is [3]. A complete map building is avoided in [4] by relaxing to a local Euclidean reconstruction. An image-based scheme is presented in [5], where direct feedback from a visual geometric constraint is used.

Work has also been done for vision-based indoor navigation of humanoid robots. In [6], a sequence of target images and a technique based on template correlation is used to decide the actions that a humanoid has to perform for navigation in corridors. The strategy proposed in [7] copes with different corridor configurations by taking advantage of vanishing points computed in that particular environment. In [8], a vision-based control scheme is proposed to track a desired trajectory in the Cartesian space. Vision and odometry data are fused to estimate the robot motion and a controller is designed on the basis of the unicycle model. Unlike some of these approaches, our strategy might be applied for less restrictive indoor environments where the sequence of target images had been previously acquired: corridors, uncluttered or cluttered environments. Similar to [6], we assume that a sequence of target images are known and for each target a visual servoing task has to be accomplished. This means that the humanoid locomotion process is directed by several visual tasks.

Humanoids are able to perform several tasks simultaneously, however, in this work we focus on the locomotion guided directly from vision. In [9], a grasping task commanded by visual servoing is performed while the humanoid is walking. Recently, this strategy has been incorporated within a visually-guided locomotion planner in [10]. In these cases, the locomotion task cannot be modified by an error function in terms of visual data. A reactive walking controller introduced in [11] overcomes this limitation. Using that scheme, the foot placements are automatically generated from a desired sequence of linear and angular velocities of the robot's center of mass (CoM). This sequence can be computed from a visual task as it is demonstrated in [12].

In this work, humanoid navigation is formulated as the problem of following a visual path, which is defined by a given sequence of target images. In particular, two types of visual controllers are evaluated: PBVS and IBVS. The visual path following consists in calculating the appropriate robot velocities for the walking task. The humanoid robot is treated as a holonomic system at the locomotion level, without imposing motion constraints (as in the unicycle model), which allows the robot to move appropriately even in visual paths that naturally requires pure lateral motion. In this context, we propose a generic control law capable to alleviate the discontinuous behavior of the robot velocities that appears when a new target image is given [3]–[5]. In summary, the contribution of the paper is the proposal of a pure vision-based navigation strategy that exploits the holonomic nature of the humanoid locomotion, mitigating discontinuities of the robot velocities.

The paper is structured as follows. Sect. II gives an overview of the navigation strategy. Sect. III describes the visual path following scheme tailored to humanoids. We

[1] J. Delfin and G. Arechavaleta are with Robotics and Advanced Manufacturing Group, Centro de Investigaci6n y de Estudios Avanzados del IPN, Saltillo, Coah. Mexico. {josafat.delfin,garechav}@cinvestav.edu.mx

[2] H. M. Becerra is with Centro de Investigación en Matemáticas (CIMAT), C.P. 36240, Guanajuato, Gto., Mexico. hector.becerra@cimat.mx

present the proposed smooth transition control in Sect. IV. Experiments are described in Sect. V and some final remarks are given in Sect. VI.

II. OVERVIEW OF THE NAVIGATION STRATEGY

A visual path consists of an ordered set of n target images acquired on a supervised teaching phase of the navigation method. In this work, we assume that the visual path is given previously, so that, we focus on the development of an appropriate control law that allows the humanoid robot replaying autonomously the memorized path.

Fig. 1 presents an overview of the proposed framework for visual path following for humanoids. At the beginning of the autonomous navigation, the robot is close to the starting position of the teaching phase, in such a way that $\mathcal{I}_1^*$ is the first target image to reach. The task of following the visual path is divided into n subtasks, each one consisting of driving to zero the visual error between the currently acquired image $\mathcal{I}$ and the next target image $\mathcal{I}_i^*$. Point features are matched between the current image $\mathcal{I}$ and the corresponding target image $\mathcal{I}_i^*$ along the robot motion. According to Fig. 1, the matched image features are used to compute the subtask error that has to be driven to zero through an adequate control law. Additionally, the mean squared error between the set of corresponding point features is used to define the switching condition to the next target image. When the error is small enough, a new target image is requested to be reached and the same whole cycle is repeated for each target image until the final one $\mathcal{I}_n^*$ is reached.

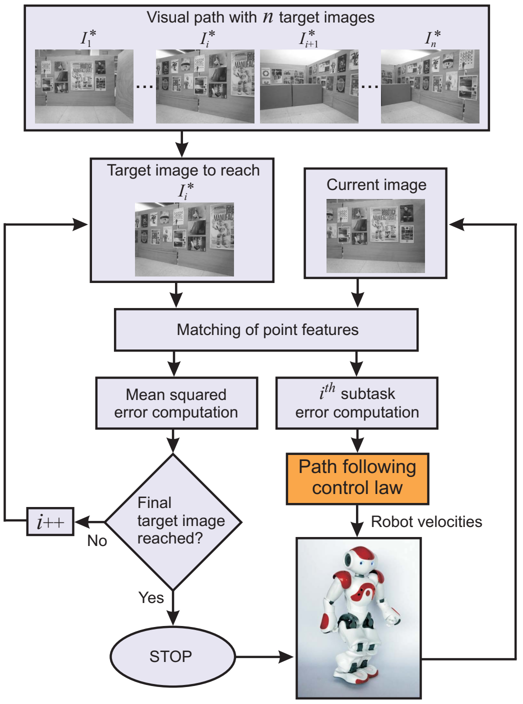

Visual path with n target images
I_1^*
I_i^*
I_{i+1}^*
I_n^*
...
...
Target image to reach
I_i^*
Current image
Matching of point features
Mean squared error computation
i^{th} subtask error computation
Path following control law
Robot velocities
Final target image reached?
No
i++
Yes
STOP

Fig. 1: Visual path following strategy for humanoids.

III. VISUAL PATH FOLLOWING SCHEME

The problem of following a visual path can be treated as a set of n visual servoing subtasks. In this sense, the problem can be solved through any of the two classical approaches of visual servoing: PBVS and IBVS. In this work, we focus on the proposal of a generic controller able to provide smooth robot velocities, which is a critical aspect for the navigation of a humanoid robot. Unlike a wheeled mobile robot [4], [5], a humanoid robot might get unbalanced in the effort to achieve a sudden change of the commanded velocities.

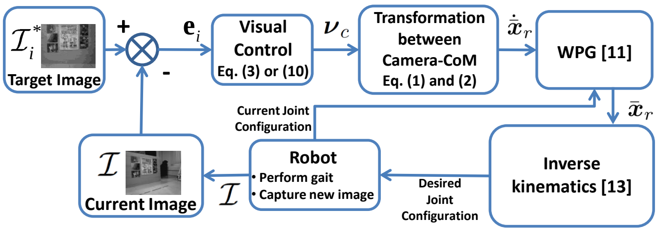


Fig. 2: Visual control scheme for humanoids.

A. Visual control scheme for humanoids

In this work the humanoid locomotion is generated by considering the well-known cart-table model that captures the relation between the acceleration of the CoM and the zero moment point (ZMP) during walking. The velocity of the CoM $\dot{\boldsymbol{x}}_r$ will be obtained by the instantaneous output of a visual servo-controller as detailed below. We use a walking pattern generator (WPG) that considers automatic footstep placements and incorporates additional inequality linear constraints [11]. The outcome of the pattern generator is then used to compute the motion coordination by means of an efficient inverse kinematics method [13].

Let $\boldsymbol{\nu}_m$ be the velocity of the frame attached to the robot's CoM. We assume that a constant transformation ${}^m\boldsymbol{T}_c$ exists between $\boldsymbol{\nu}_m$ and the velocity of the camera reference frame $\boldsymbol{\nu}_c$, so we have:

$$\boldsymbol{\nu}_m = {}^m\boldsymbol{T}_c\boldsymbol{\nu}_c \tag{1}$$

where

$${}^m\boldsymbol{T}_c = \begin{pmatrix} {}^m\boldsymbol{R}_c^T & -{}^m\boldsymbol{R}_c^T[\boldsymbol{r}]_\times \\ 0 & {}^m\boldsymbol{R}_c^T \end{pmatrix} \in \mathbb{R}^{6\times 6},$$

with $[\boldsymbol{r}]_\times \in \mathbb{R}^{3\times 3}$ representing a screw-symmetric matrix and ${}^m\boldsymbol{R}_c \in SO(3)$. This constant transformation means that the robot's head is fixed w.r.t. the robot's body. Then, the input for the WPG can be expressed as:

$$\dot{\boldsymbol{x}}_r = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \boldsymbol{\nu}_m. \tag{2}$$

Notice that only the translational components of $\dot{\boldsymbol{x}}_r$ are needed for the WPG while the angular term is used to construct the linear inequality constraints that define the next footstep placement [11]. The whole process for the visual control tailored to humanoids is shown in Fig. 2.

B. Visual Control Strategies

The aim of the visual servo-control approach is to drive a vector of m image features s to a desired value s^* [1]. In the context of visual path following, that goal must be accomplished for each target image (n visual servoing subtasks). Thus, the following subtask error can be defined:

$$\mathbf{e}_i = \mathbf{s} - \mathbf{s}_i^* \in \mathbb{R}^m.$$

The relationship between the camera velocities and the change of the image features vector is given by $\dot{\mathbf{s}} = \mathbf{L}_s \boldsymbol{\nu}_c$, where $\mathbf{L}_s \in \mathbb{R}^{m\times 6}$ is an interaction matrix. In this paper, we concern for selecting an appropriate features vector of dimension 6, i.e. $m = 6$, to have a square interaction matrix. In this paper $\mathbf{s}_i^*$ is a vector of constant values and then $\dot{\mathbf{e}}_i = \dot{\mathbf{s}}$. Thus, the time derivative of the error for each subtask is given by $\dot{\mathbf{e}}_i = \mathbf{L}_s \boldsymbol{\nu}_c$. A typical choice of $\boldsymbol{\nu}_c$ to impose an exponential decay of the error is as follows:

$$\boldsymbol{\nu}_c = -\lambda \hat{\mathbf{L}}_s^{-1} \mathbf{e}_i, \tag{3}$$

with λ a positive control gain. Notice that we will propose a six-dimensional features vector that allows us to have an invertible estimated interaction matrix $\hat{\mathbf{L}}_s$.

The classical approach to compute the camera velocity vector given in (3) has the disadvantage of being discontinuous at the instant when a new target image is given. In the sequel of this section, we describe the selected visual features for feedback and in the Sect. IV the proposed controller to achieve smooth transitions between subtasks. The switching to the next target image is given when the *image error* ε is below a threshold T_ε, i.e.:

$$\varepsilon = \frac{1}{k} \sum_{j=1}^{k} \left\| \mathbf{p}_j - \mathbf{p}_j^* \right\| < T_\varepsilon. \tag{4}$$

where k is the total number of corresponding point features. In the following two subsections, we describe two different control schemes that can be used individually to carry out the sequence of n visual servoing subtasks one by one. In the controllers description, we will consider that the robot has to reach the location associated to the i^{th} target image $\mathcal{I}_i^*$, however, to simplify the notation we do not use the subscript i to denote the subtask index.

1) Position-based scheme (3D): Let us denote by $\mathcal{C}$ and $\mathcal{C}_i^*$ the reference frames associated to the current camera pose (translation and rotation) and the i^{th} target pose respectively. A vector $\mathbf{t}$ expressed in $\mathcal{C}_i^*$ represents the translation between $\mathcal{C}$ and $\mathcal{C}_i^*$ and $\mathbf{R}$ is the rotation matrix between those reference frames. A features vector for a position-based control can be defined as follows:

$$\mathbf{s} = (\mathbf{t}, \theta\mathbf{u}) \in \mathbb{R}^6 \tag{5}$$

where $\theta\mathbf{u}$ represents the axis/angle parametrization of the rotation matrix (Rodrigues' formulation). Since the features vector encodes the translation and rotation w.r.t. the target frame, in this case $\mathbf{s}^d = 0$ and therefore $\mathbf{e} = \mathbf{s}$.

The translational and rotational velocities result to be decoupled according to (3) as detailed in [1] and the velocity vector is given by:

$$\boldsymbol{\nu}_c = \begin{bmatrix} \boldsymbol{v}_c \\ \boldsymbol{\omega}_c \end{bmatrix} = \begin{bmatrix} -\lambda \mathbf{R}^T \mathbf{t} \\ -\lambda \theta \mathbf{u} \end{bmatrix}. \tag{6}$$

Considering that the control gain accomplishes $\lambda > 0$ and that the camera pose is accurately estimated, the velocity vector (6) yields to an exponentially stable error dynamics of the form $\dot{\mathbf{e}} = -\lambda \mathbf{e}$.

There are several options to recover the relative pose between the current camera frame $\mathcal{C}$ and the target one $\mathcal{C}_i^*$. An option that does not require to know the 3D structure of the scene or a 3D model of an object in the scene is the homography matrix decomposition [14]. The homography matrix $\mathbf{H}$ can be estimated using only the images $\mathcal{I}$ and $\mathcal{I}_i^*$. The homography encodes the relative pose between the reference frames $\mathcal{C}$ and $\mathcal{C}_i^*$ as follows:

$$\mathbf{H} = \mathbf{R} + \frac{\mathbf{t}}{d^*} \mathbf{n}^{*T} \tag{7}$$

where $\mathbf{R}$ and $\mathbf{t}$ are the rotation matrix and translation vector as defined above, $\mathbf{n}^*$ is the unitary vector expressed in $\mathcal{C}_i^*$ normal to a plane π, and d^* is the distance from π to $\mathcal{C}_i^*$.

Thus, it is possible to decompose $\mathbf{H}$ according to (7) to obtain $\mathbf{R}$ and $\mathbf{t}$, which are necessary to implement the control law (6). An efficient algorithm to carry out such decomposition is proposed in [15]. It is well known that the decomposition of $\mathbf{H}$ generates two geometrically valid solutions, however, only one of them is physically admissible. The correct solution can be selected taken the solution associated to the normal vector whose third component (n_z) will be the largest. We emphasize that although the homography model is valid for planar scenes, it is possible to estimate a homography associated to a virtual plane for non-planar scenes using the algorithm proposed in [14].

2) Image-based scheme (2D): The homography model has also proved to be efficient for 2D visual servo-control. A control scheme that formulates a control law directly in terms of the homography matrix is proposed in [16]. Error vectors for translation and rotation are directly computed from the homography matrix. The core idea in that scheme is that the reference frames $\mathcal{C}$ and $\mathcal{C}_i^*$ coincide, if and only if the homography matrix $\mathbf{H}$ is equal to the identity matrix $\mathbf{I}$. Under that notion, a task function $\mathbf{s} = \mathbf{e} \in \mathbb{R}^6$ locally isomorphic to the camera's pose is defined. The task function $\mathbf{e} = (\mathbf{e}_v^T, \mathbf{e}_\omega^T)^T$ is null if and only if the camera reaches the target pose and it is given by:

$$\begin{aligned} \mathbf{e}_v &= (\mathbf{H} - \mathbf{I})\mathbf{m}^*, \\ [\mathbf{e}_\omega]_\times &= \mathbf{H} - \mathbf{H}^T \end{aligned} \tag{8}$$

where $\mathbf{m}^*$ is any point in the target image $\mathcal{I}_i^*$ that belongs to the plane (virtual plane for non-planar scenes) that defines $\mathbf{H}$, and $[\mathbf{e}_\omega]_\times$ represents the skew-symmetric matrix for the vector $\mathbf{e}_\omega = (e_{\omega x}, e_{\omega y}, e_{\omega z})^T$.

Thus, the task function $\mathbf{e}$ can be estimated using only the images $\mathcal{I}$ and $\mathcal{I}_i^*$, without the need of obtaining the 3D structure of the target pose (d^*). The homography can be estimated using the method of [14] for non-planar scenes. The time-derivative of the task function gives the equation $\dot{\mathbf{e}} = \mathbf{L}\boldsymbol{\nu}_c$, where $\mathbf{L}$ is a square interaction matrix. In [16] it is shown that this control scheme is efficient even without using the interaction matrix. Thus, the linear control

$$\boldsymbol{\nu}_c = \begin{bmatrix} \boldsymbol{v}_c \\ \boldsymbol{\omega}_c \end{bmatrix} = \begin{bmatrix} -\lambda \mathbf{e}_v \\ -\lambda \mathbf{e}_\omega \end{bmatrix} \tag{9}$$

with $\lambda > 0$, yields an error dynamics locally exponentially stable. The local stability of the task function is guaranteed for any 3D structure and any point $\mathbf{m}^*$ [16].

IV. CONTROLLER FOR SMOOTH TRANSITION

Any of both previous controllers, (6) and (9), are of the form of the generic controller (3). These controllers, introduced in the WPG according to Fig. 2, are able to drive the humanoid robot to the location associated to the target image $\mathcal{I}_i^*$. However, in the following of the visual path the error is large when a new target image is given and the robot velocities are discontinuous at the switching of each subtask.

In order to avoid such discontinuities, we propose to introduce a smooth transition function $h(t)$ that penalizes the large error at the beginning of each subtask. The proposed control law written in a generic form is given by:

$$\boldsymbol{\nu}_c = -\lambda h\left(t\right)\hat{\mathbf{L}}_s^{-1}\mathbf{e}_i. \tag{10}$$

The introduction of the transition function $h(t)$ in the control law has the effect of having a time-varying gain. The generic control law (10) can be implemented as any of the two controllers of the previous subsections: the PBVS (6) or the IBVS (9). An adequate transition function is as follows:

$$h(t) = \begin{cases} h_0 + \frac{1}{2}\left(1-h_0\right)\left(1-\cos\left(\frac{\pi(t-t_0)}{t_f-t_0}\right)\right), & t_0 \le t \le t_f, \\ 1, & t > t_f. \end{cases}$$

The function $h(t)$ has a minimum value h_0, from which it increases smoothly up to the unity. Thus, after t_f, the maximum value λ of the control gain is applied. It is worth noting that the minimum value of $h(t)$ allows the robot to achieve a continuous motion (without stopping) while the discontinuities in the velocities are significantly reduced. It is important to set adequately the duration of the transition function $t_f - t_0$ to ensure that the maximum control gain is applied at least during some time for every subtask.

Without loss of generality, we have assumed in (10) that the same value of the gain λ is used for each component of the velocity vector, however, it is possible to use different control gains for each one of the components as it will be described in the experimental results. An important aspect of the control law (10) is that the time-dependent function $h(t)$ makes the closed loop dynamics time-dependent. This is:

$$\dot{\mathbf{e}}_i = -\lambda h\left(t\right)\mathbf{L}_s\hat{\mathbf{L}}_s^{-1}\mathbf{e}_i. \tag{11}$$

Consequently, (11) is a non-autonomous system with equilibrium point $\mathbf{e}_i = \mathbf{0}$. An stability analysis for this system must consider the property of non-autonomy [17]. Let us define the following candidate Lyapunov function:

$$V\left(\mathbf{e}_i\right) = \frac{1}{2}\mathbf{e}_i^T\mathbf{e}_i, \tag{12}$$

that is continuously differentiable, positive definite and decrescent as required [17]. The last two properties mean that the inequality $W_1\left(\mathbf{e}_i\right) \le V\left(\mathbf{e}_i\right) \le W_2\left(\mathbf{e}_i\right)$ is accomplished for all $\mathbf{e}_i \in \mathbb{R}^6$. Because the candidate Lyapunov function (12) does not explicitly depend on time, $W_1\left(\mathbf{e}_i\right)$ and $W_2\left(\mathbf{e}_i\right)$ are trivially found to be $W_1\left(\mathbf{e}_i\right) = W_2\left(\mathbf{e}_i\right) = V\left(\mathbf{e}_i\right)$.

The time-derivative of the candidate Lyapunov function is

$$\dot{V} = -\lambda h\left(t\right)\mathbf{e}_i^T\mathbf{L}_s\hat{\mathbf{L}}_s^{-1}\mathbf{e}_i.$$

To show that the equilibrium point $\mathbf{e}_i = \mathbf{0}$ of the non-autonomous system (11) is asymptotically stable, a positive definite function $W_3\left(\mathbf{e}_i\right)$ must be found such that:

$$\frac{\partial V}{\partial t} + \frac{\partial V}{\partial \mathbf{e}_i}\dot{\mathbf{e}}_i \le -W_3\left(\mathbf{e}_i\right). \tag{13}$$

The term $\partial V/\partial t$ is zero because $V\left(\mathbf{e}_i\right)$ is not a function of time. Additionally, we have that:

$$\frac{\partial V}{\partial \mathbf{e}_i}\dot{\mathbf{e}}_i = \dot{V} = -\lambda h\left(t\right)\mathbf{e}_i^T\mathbf{L}_s\hat{\mathbf{L}}_s^{-1}\mathbf{e}_i. \tag{14}$$

Given that $h(t)$ satisfies a lower bound condition:

$$h(t) \ge h_0 > 0.$$

Then, we have:

$$\frac{\partial V}{\partial \mathbf{e}_i}\dot{\mathbf{e}}_i = -\lambda h\left(t\right)\mathbf{e}_i^T\mathbf{L}_s\hat{\mathbf{L}}_s^{-1}\mathbf{e}_i \le -\lambda h_0\mathbf{e}_i^T\mathbf{L}_s\hat{\mathbf{L}}_s^{-1}\mathbf{e}_i. \tag{15}$$

Therefore, we can set $W_3\left(\mathbf{e}_i\right) = \lambda h_0\mathbf{e}_i^T\mathbf{L}_s\hat{\mathbf{L}}_s^{-1}\mathbf{e}_i$ and asymptotic stability is guaranteed if and only if $\mathbf{L}_s\hat{\mathbf{L}}_s^{-1} > 0$, i.e., the matrix $\mathbf{L}_s\hat{\mathbf{L}}_s^{-1}$ is positive definite. This is the typical stability condition for a visual servo-controller, however, in this case the additional condition $h(t) \ge h_0 > 0$ on the transition function must be satisfied. For the PBVS (6), the condition $\mathbf{L}_s\hat{\mathbf{L}}_s^{-1} > 0$ holds globally for a good estimation of the pose parameters. For the IBVS (9), the condition $\mathbf{L}_s\hat{\mathbf{L}}_s^{-1} > 0$ holds locally around $\mathbf{e}_i = \mathbf{0}$, as shown in [16].

V. EXPERIMENTAL EVALUATION

We implemented the proposed navigation controller within a NAO humanoid robot. The top camera mounted on the robot's head was used in the experiments. The images were obtained at a frame rate of 12Hz with a resolution of 640 × 480 pixels. The image features were acquired as follows: first, a corner detector based on [18] was used, which is implemented in the function `goodFeaturesToTrack` of the OpenCV library. Then, we assigned a SIFT descriptor [19] to each detected point. A robust matcher based on RANSAC matched all the points between the current image and the corresponding target image. Finally, a tracking algorithm based on a sparse iterative version of the Lucas-Kanade optical flow in pyramids was utilized. We used the function `calcOpticalFlowPyrLK` from OpenCV. The tracking of points from a camera mounted on a humanoid robot is not trivial due to the jerky camera movements generated by the robot's gait. Thus, the tracker was experimentally tuned to deal with this unavoidable behavior. We evaluated experimentally the performance of the tracker with the NAO robot. It was found that the tracker is able to work properly at the maxima forward, lateral and rotational velocities whereas the maximum step size are 0.04m for forward direction, 0.14m for lateral and 0.349rad for rotation. The `HLM` function of ViSP library was used to compute the homography for planar and non-planar scenes [14].

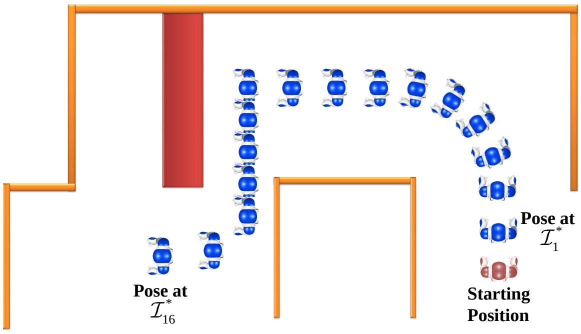


Fig. 3: Path to be followed by the robot in the experiments.

The visual path was defined by 16 target images over a total distance of 4.34m. A bird's-eye view of the path is depicted in Fig. 3, where the red robot marks the starting position and the sequence of the blue robots marks the 16 target images. The path has been designed to show the adaptability of the vision-guided humanoid locomotion depending on the shape of the visual path. Thus, the path consists in a curved segment, similar to one performed by a differential drive robot, and a rectilinear lateral segment. The red rectangle in Fig. 3 represents a fence-like obstacle which does not occlude the view of the camera. The goal of using this experimental setup is to simulate an environment with a narrow passage.

Regarding the control parameters, it is convenient to use different control gains for translation and for rotation, because in the bipedal locomotion, it might be easier for the robot to move laterally compared to rotate, so that, we used $\lambda = 0.06$ for translation and $\lambda = 0.2$ for rotation. The duration of the transition function $h(t)$ was set experimentally to 40 iterations. We found out experimentally that $h_0 = 0.05$ represents a good compromise between mitigation of the discontinuities of walking velocities and continuity of motion during the robot navigation. The threshold T_ε was set to 13 pixels for the whole path.

We have implemented the visual controls described in Sect. III, without and with smooth transitions, and the results are shown in Figs. 4(a)-4(d). In those figures, we emphasize some details by zooming the interval of the plots marked by an ellipse. Those intervals of the plots correspond to parts of the path where a larger change of direction takes place, and therefore the discontinuities in the velocity ω_z are higher. It is also possible to appreciate discontinuities in the lateral velocity υ_y, during the lateral motion (approximately between the 2000 and 3000 iterations for each plot in all controls).

Figs. 4(a) and 4(b) show the velocities during the whole navigation for the position-based control, without and with smooth transitions respectively. In Fig. 4(a) it can be seen at the zoomed area the discontinuities at the switching of each subtask. In Fig. 4(b), the same discontinuities are smoothed by the transition function. Figs. 4(c) and 4(d) show the results of the image-based control without and with smooth transitions respectively. Notice that the discontinuous behavior of the velocities in Fig. 4(c) is alleviated in Fig. 4(d) by introducing the transition function.

The evaluated controllers solve the global navigation task by driving the robot to a vicinity of the location associated to the final target image with an error less than 6% of the total path length. Nevertheless, the repeatability of the task can be ensured by means of smooth transitions. According to the results a satisfactory estimation of the pose parameters has been achieved for the PBVS. However, the sensitivity of this approach to a good pose estimation reduces the repeatability of the results in comparison to the IBVS. In this sense, the IBVS control outperforms the PBVS control for several trials.

It is worth emphasizing that the threshold T_ε plays an important role on the precision of each subtask. Nevertheless, a small threshold is not always the best option in order to obtain a natural robot motion for navigation. For the experiments, we have achieved a good behavior by fixing T_ε. Finally, the experimental results validate how our visual path following scheme exploits the reactivity of the footstep placements computed by WPG. The humanoid is able to perform forward and lateral steps in indoor environment.

VI. Conclusions

In this paper, we have shown the feasibility of the visual path following approach for humanoid robots navigation in indoor environments. We assumed that a sequence of target images is known. We proposed a generic visual control law that copes with discontinuous robot velocities appeared when the target image is replaced by a new one during humanoid navigation. In particular, we incorporated smooth transition functions in the control laws. The stability of the proposed scheme has been proved theoretically and verified through experiments with a real humanoid platform. The control scheme takes advantage of the reactive footstep placements without the need of any footstep planning to decide the type of walking behavior that better adapts to the visual servoing task, e.g., forward or lateral stepping. We have evaluated both, a position-based and an image-based methods. An important ingredient has been the estimation of an homography associated to a virtual plane for non-planar scenes.

References

[1] F. Chaumette and S. Hutchinson. Visual servo control. Part I: Basic approaches. *IEEE Robotics and Autom. Magazine*, 13(4):82–90, 2006.

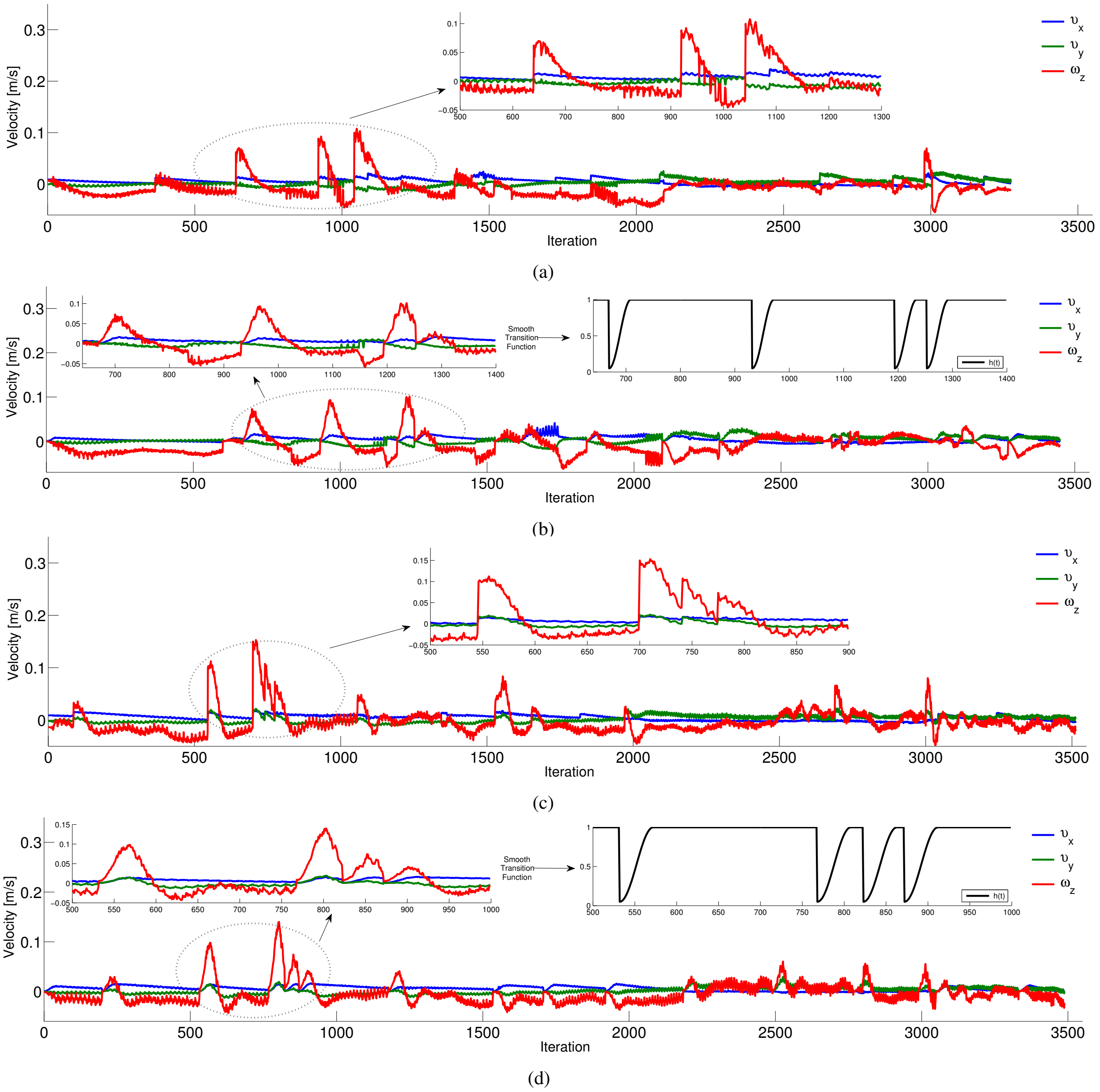
[2] Y. Matsumoto, M. Inaba, and H. Inoue. Visual navigation using view-sequenced route representation. In *IEEE Int. Conf. on Robotics and Automation*, pages 83–88, 1996.

[3] E. Royer, M. Lhuillier, M. Dhome, and J. M. Lavest. Monocular vision for mobile robot localization and autonomous navigation. *Int. Journal of Computer Vision*, 74(3):237–260, 2007.

[4] J. Courbon, Y. Mezouar, and P. Martinet. Autonomous navigation of vehicles from a visual memory using a generic camera model. *IEEE Trans. on Intelligent Transportation Systems*, 10(3):392–402, 2009.

[5] H. M. Becerra, C. Sagüés, Y. Mezouar, and J. B. Hayet. Visual navigation of wheeled mobile robots using direct feedback of a geometric constraint. *Autonomous Robots*, 37(2):137–156, 2014.

[6] J. Ido, Y. Shimizu, Y. Matsumoto, and T. Ogasawara. Indoor navigation for a humanoid robot using a view sequence. *Int. Journal of Robotics Research*, 28(2):315–325, 2009.



(a)

(b)

(c)

(d)

Fig. 4: **Input velocities to the WPG.** The figure displays the results for the four visual controls: 3D control (a), 3D control with smooth transition (b), 2D control (c) and 2D control with smooth transition (d). The ellipses mark a zoomed region in the velocity plots. Also $h(t)$ is shown at the zoomed iterations for the smoothed controls.

[7] A. Faragasso, G. Oriolo, A. Paolillo, and M. Vendittelli. Vision-based corridor navigation for humanoid robots. In *IEEE Int. Conf. on Robotics and Automation*, pages 3175–3180, 2013.

[8] G. Oriolo, A. Paolillo, L. Rosa, and M. Vendittelli. Vision-based trajectory control for humanoid navigation. In *IEEE-RAS Int. Conf. on Humanoid Robots*, pages 118–123, 2013.

[9] N. Mansard, O. Stasse, F. Chaumette, and K. Yokoi. Visually-guided grasping while walking on a humanoid robot. In *IEEE Int. Conf. on Robotics and Automation*, pages 3041–3047, 2007.

[10] J.-H. Hayet, C. Esteves, G. Arechavaleta, O. Stasse, and E. Yoshida. Humanoid locomotion planning for visually-guided tasks. *Int. Journal of Humanoid Robots*, 9(2):26, 2012.

[11] A. Herdt, H. Diedam, P.-B. Wieber, D. Dimitrov, K. Mombaur, and M. Diehl. Online walking motion generation with automatic footstep placement. *Advanced Robotics*, 24(5-6):719–737, 2010.

[12] C. Dune, A. Herdt, O. Stasse, P.-B. Wieber, K. Yokoi, and E. Yoshida. Cancelling the sway motion of dynamic walking in visual servoing. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 3175–3180, 2010.

[13] O. Kanoun. Real-time prioritized kinematic control under inequality constraints for redundant manipulators. In *Robotics: Science and Systems VII*, Los Angeles, CA, USA, June 2011.

[14] E. Malis, F. Chaumette, and S. Boudet. 2 1/2 visual servoing with respect to unknown objects through a new estimation scheme of camera displacement. *Int. Journal of Comp. Vision*, 37(1):79–97, 2000.

[15] B. Triggs. Autocalibration from planar scenes. In H. Burkhardt and B. Neumann, editors, *Computer Vision - ECCV'98*, volume 1406 of *LNCS*, pages 89–105. Springer Berlin Heidelberg, 1998.

[16] S. Benhimane and E. Malis. Homography-based 2D visual tracking and servoing. *Int. Journal of Robotics Research*, 26(7):661–676, 2007.

[17] H. K. Khalil and J. W. Grizzle. *Nonlinear systems*. Prentice Hall, Upper Saddle River, 2002.

[18] J. Shi and C. Tomasi. Good features to track. In *IEEE Computer Vision and Pattern Recognition*, pages 593–600, 1994.

[19] David G Lowe. Object recognition from local scale-invariant features. In *IEEE Int. Conf. on Computer Vision*, pages 1150–1157, 1999.