

# Vision-Based Corridor Navigation for Humanoid Robots

Angela Faragasso   Giuseppe Oriolo   Antonio Paolillo   Marilena Vendittelli

**Abstract**—We present a control-based approach for visual navigation of humanoid robots in office-like environments. In particular, the objective of the humanoid is to follow a maze of corridors, walking as close as possible to their center to maximize motion safety. Our control algorithm is inspired by a technique originally designed for unicycle robots and extended here to cope with the presence of turns and junctions. The feedback signals computed for the unicycle are transformed to inputs that are suited for the locomotion system of the humanoid, producing a natural, human-like behavior. Experimental results for the humanoid robot NAO are presented to show the validity of the approach, and in particular the successful extension of the controller to turns and junctions.

## I. INTRODUCTION

Humanoid robots have been the subject of a lively research activity in the last years. This is due to the challenging scientific problems that they present to researchers as well as to the potential impact of these robots in real applications. In fact, the high flexibility and dexterity of these systems would allow to accomplish tasks that can hardly be carried out by other robots, both in variability and complexity.

Until now, research activity has primarily focused on fundamental issues related, e.g., to stability of locomotion, composition of motion and manipulation tasks, whole body motion, motion planning and other basic problems in humanoid robot control. Achieving fully autonomous humanoids, however, will require an increased exploitation of the exteroceptive sensing capabilities of these complex robotic systems. On the other hand, the recent availability of experimental platforms with reasonable cost and built-in basic functionalities, like robust and reactive locomotion systems, allows to investigate and design sensor-based control paradigms that are sufficiently abstract to be portable on different humanoid robots.

In this paper, we consider the problem of corridor navigation in office-like environments for a humanoid robot equipped with a camera. Available literature mainly proposes methods for navigation in indoor environments based on localization on known maps [1], [2] or preregistered images of the environment [3].

In particular, we propose an Image-Based Visual Servoing (IBVS) approach to generate walking commands that force the humanoid to walk at the center of the corridor using only visual information and natural features, and with no a priori knowledge of the environment. The few works that

have used the visual servoing paradigm to generate walking motion of humanoids include [4], where a Position Based Visual Servoing has been used to move the robot to specified locations, and [5] that proposes a motion generation approach to the animation of human characters based on the concept of IBVS.

Our method relies on the corridor following algorithm developed in [6] for a mobile robot with unicycle kinematics. With respect to the numerous works available in the literature for indoor navigation of wheeled mobile robots [7], [8], this is a map-less method for corridor navigation with formally proved heading and position convergence. We adapt this approach to humanoid navigation and generalize the control strategy to allow the humanoid to successfully negotiate turns and junctions. The resulting indoor navigation method does not require switching among different control techniques [9] or complex image processing [10] to achieve navigation tasks like turning around a corner: the navigation task is always achieved using the same control law. The main contribution of this paper is therefore the design and experimental validation of a simple and robust vision-based corridor navigation method producing natural motion for humanoid robots.

The paper is organized as follows. In the next section, we specify the control objective and give an overview of the adopted approach. Section III contains a short summary of the visual-based corridor navigation method proposed in [6] for a unicycle robot. Extensions of this method allowing to handle square turns and T-junctions are described respectively in Sect. III-B and Sect. III-C. Section IV describes how the velocity generated by the unicycle IBVS control law can be converted in walking commands for humanoid robots. In particular, this conversion procedure is detailed for the humanoid robot NAO. Section V reports experimental results obtained by using the proposed method for corridor navigation with NAO.

## II. CONTROL OBJECTIVE AND APPROACH

Consider a humanoid whose task requires navigation in a corridor-like maze, i.e., an environment made by long rectilinear tracts contained between two parallel walls and connected by square corners and junctions. Our objective is to design a visual control scheme enabling the robot to walk as close as possible the corridor center, turn at corners and cross junctions to carry on in a certain direction (specified by a higher-level module).

Experimental studies [11], [12] on human locomotion resulted in the formulation of a gait model according to which, on long distance walks in uncluttered environments,

The authors are with the Dipartimento di Ingegneria Informatica, Automatica e Gestionale, Sapienza Università di Roma, via Ariosto 25, 00185 Roma, Italy. Email: {oriolo, paolillo, vendittelli}@dis.uniroma1.it, angelafar86@gmail.com

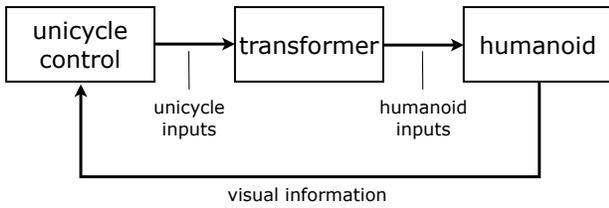


Fig. 1. A block scheme of the proposed control approach for corridor navigation of humanoid robots.

the orientation of our body is tangent to the path for most of the time. In other words, this kind of human paths resemble very closely those typical of nonholonomic wheeled mobile robot, such as the unicycle.

The corridor navigation problem fits the long distance walk scenario quite well. Motivated by this, and assuming that the walking speed and direction of the humanoid can be controlled by sending reference velocities to its locomotion system, we shall first generate velocity commands for a virtual unicycle robot and then convert these commands in real-time to reference velocities for the humanoid. An important feature of our method is that we will use only visual information as feedback. See Fig. 1 for an illustration of the proposed approach.

### III. UNICYCLE CORRIDOR NAVIGATION

Our unicycle control algorithm relies on the algorithm proposed in [6]. In the following, we first give a summary of its basic structure and then propose original extensions for handling turns and junctions.

#### A. Basic Scheme

Consider a wheeled mobile robot with unicycle kinematics in a corridor. Its generalized coordinates are  $(x, y, \theta)$  as defined in Fig. 2; note in particular that the  $y$  axis is parallel to the corridor walls. The kinematic model of this system is readily written as

$$\begin{aligned} \dot{x} &= v \sin \theta \\ \dot{y} &= v \cos \theta \\ \dot{\theta} &= \omega \end{aligned} \quad (1)$$

where  $v$  and  $\omega$  are respectively the vehicle driving and steering velocity. In the rest of this section, we assume that  $v$  is constant and chosen in advance while  $\omega$  is the actual available control input (a common setting for path tracking control of mobile robots).

Assume that a pinhole camera is rigidly mounted on the robot, at a fixed orientation with respect to the ground and the optical axis aligned with the robot heading. For simplicity, we suppose that the camera focus lies on the vertical axis passing through the representative point on the robot (the midpoint of the wheels axle, if the unicycle is a differential-drive vehicle); as a consequence, the camera has the same velocity of the unicycle. This assumption can be easily removed to account for nonzero displacements.

The control objective is to navigate at the center of the corridor using information from the camera. The relevant

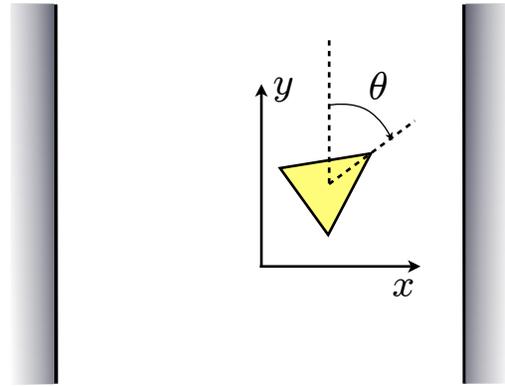


Fig. 2. Unicycle in a corridor. The configuration of the robot is expressed with respect to a fixed reference frame with origin at the center of the corridor.

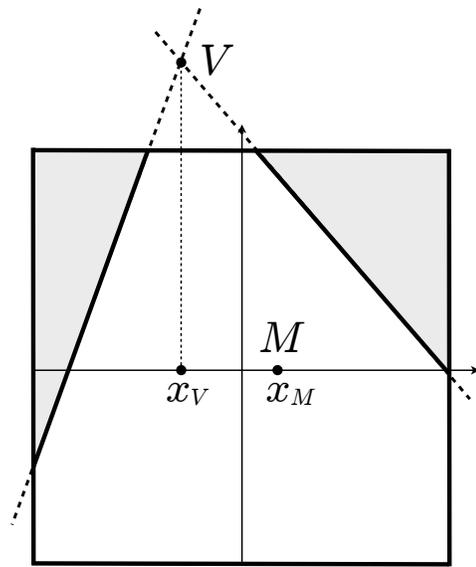


Fig. 3. A typical image seen by the robot during corridor navigation. The vanishing point  $V$  is the point where the two corridor guidelines meet, while the middle point  $M$  bisects the segment between the intersection points of the same lines with the horizontal axis of the image plane.

visual features are the images of the edges between the corridor floor and the walls, hereafter called *corridor guidelines*. Two useful features that can be reconstructed from the guidelines are the *vanishing point*  $V$  and the *middle point*  $M$  (see Fig. 3). The first is defined as the intersection point of the corridor guidelines. The second is the midpoint of the segment between the intersection points of the same lines with the horizontal axis of the image plane.

Using perspective geometry, the abscissas of the vanishing and the middle point can be expressed as

$$\begin{aligned} x_V &= k_1 \tan \theta \\ x_M &= k_2 \frac{x}{\cos \theta} + k_3 \tan \theta, \end{aligned} \quad (2)$$

where  $k_i$ ,  $i = 1, 2, 3$  are constant parameters related to the camera intrinsic characteristics (focal length and scale fac-

tors) and configuration (height and orientation with respect to the ground).

The expressions in (2) show that  $x_V$  depends only on the orientation of the robot, while  $x_M$  is also related to its position. Since the objective is to traverse the corridor staying as close as possible to its center, it is reasonable to use both these variables to correct position and orientation errors. In fact, convergence of  $x_V$  to zero would only imply that the robot is correctly aligned with the corridor; whereas convergence of both  $x_V$  and  $x_M$  to zero entails that the robot is traveling at the center of the corridor as desired.

Differentiating the expressions (2) with respect to time and using the vehicle model (1) provides the dynamics of the visual features

$$\dot{x}_V = \frac{x_V^2 + k_1^2}{k_1} \omega$$

$$\dot{x}_M = \frac{k_2 v}{k_1} x_V + (k_3 + \frac{x_M x_V}{k_1}) \omega.$$

Feedback linearization and stabilization of the dynamics of  $x_M$  lead to the following control law:

$$\omega = \frac{k_1}{k_1 k_3 + x_M x_V} \left( -\frac{k_2 v}{k_1} x_V - k_p x_M \right), \quad (3)$$

where  $k_p$  is a positive gain. A simple Lyapunov analysis can be used to show that the above choice of  $\omega$  guarantees asymptotic convergence of both  $x_M$  and  $x_V$  to zero.

Since visual features are inevitably affected by noise, it would be advisable to replace  $x_V$  and  $x_M$  with a filtered version in any practical implementation of the control law (3); a first-order low-pass filter is a typical choice.

The above vision-based control algorithm implicitly assumes that the image features are always defined, i.e., that the corridor guidelines are always in the camera field of view. However, this is not true in the presence of turns and/or junctions. While approaching these (Figs. 4 and 5), one or both the corridor guidelines will be lost, making it impossible to determine the vanishing and the middle point. In the rest of this section, we propose an extension of the visual control scheme that can handle these special situations.

### B. Turns

Consider a mobile robot approaching a right turn at a square corner, as shown in Fig. 4. Some sample camera images collected during a turn are shown in Fig. 6 (for the sake of clarity, these are not actual images but simply schematic drawings).

In the first frame, early in the approaching phase, the corridor guidelines are still clearly detectable in the image. The right guideline is however shorter than the left one, and will gradually disappear from the image as the robot comes closer to the corner. In the second frame, the right guideline has disappeared from the image. In this situation, the right side of the image takes the place of the disappeared guideline to compute  $x_V$  and  $x_M$ , which move to the right and force the robot to start turning in the same direction. The third frame shows the camera image when the robot

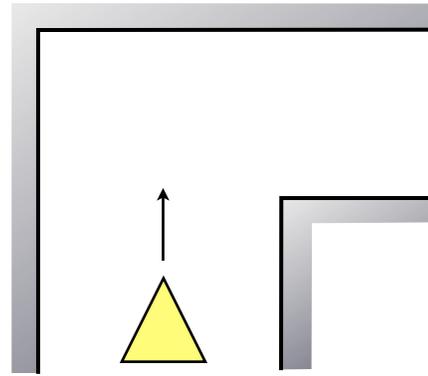


Fig. 4. Approaching a right turn.

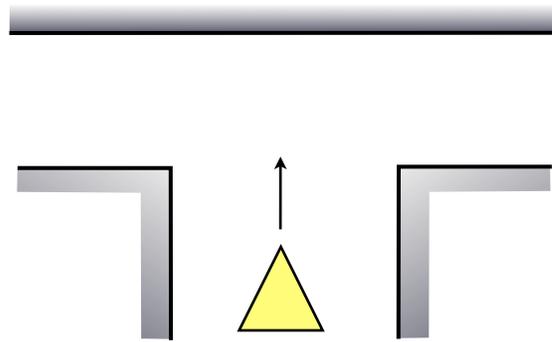


Fig. 5. Approaching a T-junction from below.

is in the middle of the turn. In this phase, the left corridor guideline on the exit side of the turn has been detected and is used together with the right side of the image, still forcing the robot to turn right. When both the guidelines are visible again the robot automatically resumes its normal navigation behavior at the center of the corridor (fourth frame).

Note that replacing the disappearing guidelines with the sides of the images causes a discontinuity in the values of  $x_V$  and  $x_M$ . However, the filtering action mentioned before smooths this jump, ultimately resulting in a continuous feedback signal.

Left turns are detected by the loss of the left corridor guideline and are handled in a specular way.

### C. T-Junctions

The guideline replacement strategy used for turns is also effective in the case of T-junctions. Consider a mobile robot approaching a T-junction from below, as shown in Fig. 4, and assume that a right turn must be executed. Sample camera images collected during the maneuver are shown in Fig. 7.

At the start of the approaching phase, shown in the first frame, the corridor guidelines are still visible but they gradually disappear as the robot comes closer to the junction. When both are lost, the lateral sides of the image take their place to compute the visual features, as shown in the second

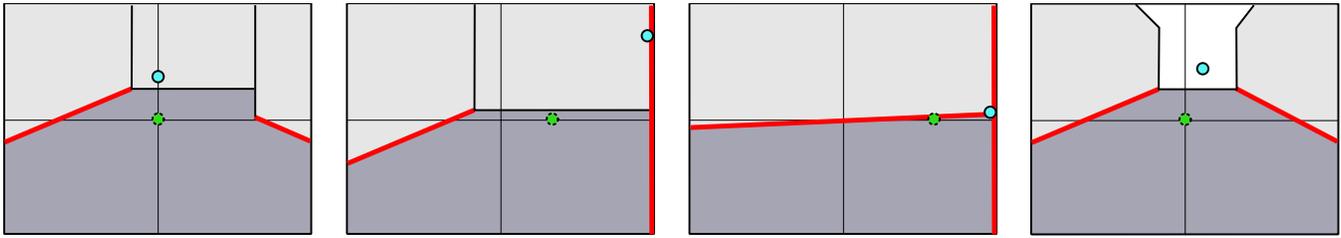


Fig. 6. Camera images during a right turn, with the visual features used to compute the feedback signal. Red (thick) lines are the guidelines used to compute the vanishing point (the cyan dot with solid contour) and the middle point (the green dot with dashed contour). In the second and third frames the right side of the image is used in place of the right corridor guideline, which has disappeared from the image.

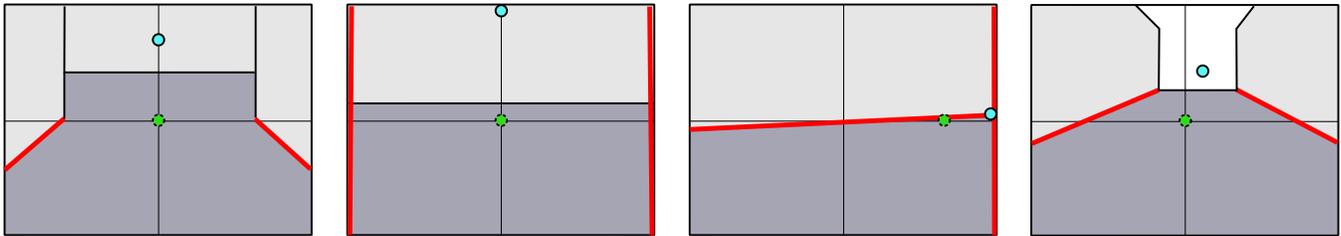


Fig. 7. Camera images during a right turn at a T-junction. As before, red (thick) lines are the guidelines used to compute the vanishing point (the cyan dot with solid contour) and the middle point (the green dot with dashed contour). In the second and third frames the corresponding side of the image is used in place of the disappeared corridor guideline. Note that in the second frame the vanishing point (which would be located at the infinity) has been projected on the top side of the image.

frame. In this case, the vanishing point is located at the infinity but its abscissa  $x_V$  — the only quantity needed to compute the feedback control — is zero, as can be verified by a simple limit computation. Hence, the robot continues to advance until the guideline of the corridor edge in front of the robot becomes sufficiently close, according to a given threshold, to the horizontal axis of the image plane. This triggers the turning maneuver, that is obtained by computing the features using that guideline and, depending on the desired turning direction, the appropriate of the image sides to compute the features. To allow the computation of the middle point, the horizontal guideline is slightly rotated (in the clockwise direction for a right turn, counterclockwise for a left turn). The third frame of Fig. 7 shows the case of a right turn. As before, when both the guidelines come back in the camera field of view the robot automatically resumes its normal navigation behavior at the center of the corridor (fourth frame).

#### D. Simulations

The visual navigation approach described in the previous sections has been validated by running Webots simulations for a unicycle wheeled robot. For image processing we have used the OpenCV library [13]. In particular, we used Canny's algorithm for edge detection and the Hough transform to identify lines in the image. Corridor guidelines are found in the processed image by discarding all lines whose slope and position in the image exceed properly defined limits.

Simulation results are not shown here for lack of space but can be seen at the web page <http://www.dis.uniroma1.it/~labrob/research/VisNav4Hum.html>,

where additional implementation details and parameter values are also provided.

#### IV. FROM UNICYCLES TO HUMANOIDS

The differential equations (1) represent an abstract mathematical model of mobility. For wheeled mobile robots, this model accounts for the nonholonomic constraints deriving from the rolling wheels; i.e., it has a physical motivation due to the mechanical nature of the motion generation system. In real robots, however, the actual control inputs could be different from the velocity commands  $v$  and  $\omega$  in (1). For differential-drive vehicles, for example, these commands must be transformed to rotational speed for the wheels. By analogy, we need to convert the unicycle feedback commands in admissible inputs for the locomotion system of a humanoid.

Humanoid robots are, by definition, endowed with omnidirectional walk capability. The mobility model (1) is therefore admissible for these systems and the velocity commands  $v$  and  $\omega$  generated for the unicycle control can be converted in admissible inputs for the low-level locomotion controller so that the humanoid follows the same path as the unicycle.

For the humanoid robot NAO, which we have used as experimental platform, this paradigm can be instantiated by using the built-in method *setWalkTargetVelocity*, which allows direct control of the driving and steering velocity of the robot. Since the most recent command overrides all previous commands, this function can be called with arbitrary frequency, thus providing the reactive behavior required by visual-based control.

The input arguments of *setWalkTargetVelocity* are the desired cartesian displacement (step length and direction), the

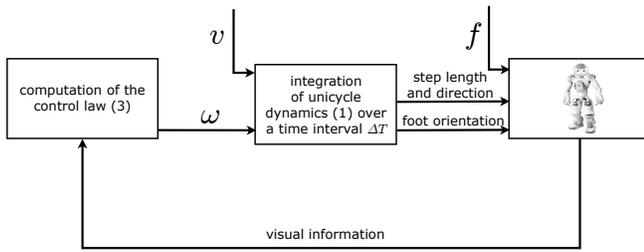


Fig. 8. The control loop for the humanoid robot NAO.

relative orientation of the feet in one step and the normalized step frequency  $f$ . The walking speed can be controlled either by varying the step length or the step frequency.

To determine the input values of the *setWalkTargetVelocity*, we integrate the unicycle equations (1) under the control law (3) over a time interval  $\Delta T$ , thus determining the step length and direction and the desired relative feet orientation at the end of the step. The step frequency can be assigned an arbitrary value within the admissible bounds.

The complete control scheme is illustrated in Fig. 8. The visual information provided by NAO's onboard camera is used to compute the steering velocity  $\omega$  according to the feedback control law (3). In correspondence of turns and junctions, where the corridor guidelines are lost, the visual features  $x_V$  and  $x_M$  are computed in the way described in Sects. III-B and III-C, respectively.

The computed steering velocity  $\omega$  and a chosen constant value of the input  $v$  are used to integrate the unicycle equations (1) for generating the desired robot displacement, which is sent to NAO in terms of the cartesian displacement and relative orientation of the feet at the end of a step. The value of the normalized frequency  $f$  is an exogenous input that can be set arbitrarily between 0 and 1, with the maximum corresponding to a step frequency of 2.381 Hz.

Note that the input transformation described above allows to control the path of a point between the feet and the position and orientation of the feet along this path. If the feet always move tangentially to the path, the torso will follow with reasonable approximation. The same approach has been used used, for example, to solve motion planning problems for humanoid robots: a bounding box is often considered for planning the path of the robot and then a sequence of footsteps is placed along the path [14], [15]. In visual-based control, the error due to the approximation mentioned above is corrected by the feedback action, as confirmed by our experiments.

It is also worth noting that the main source of image noise is the sway motion of the humanoid during walking gait. We purposely chose not compensate for this effect, but the first order digital filter, introduced to smooth the discontinuities in the visual features measures and the effect of other noise sources, proved to be effective for canceling out also the noise due to the walking.

## V. EXPERIMENTAL RESULTS

Experimental validation of the proposed visual navigation controller has been carried out on the small-size humanoid NAO. This robot is equipped with a CMOS camera with a  $58^\circ$  field of view mounted on its forehead. The camera intrinsic parameters have been determined using the MATLAB camera calibration toolbox. In the experiments, the head does not move with respect to the torso and the pitch angle has been set to  $20^\circ$ , meaning that the camera looks slightly down while NAO walks. Images used to compute the visual features are extracted from a video stream with a 7 Hz frame rate and a resolution of  $320 \times 240$ ; image processing is performed with the same techniques in Sect. III-D. A first-order filter with a cutoff frequency of 0.1 rad/sec was used to smooth out  $x_V$  and  $x_M$ . The driving velocity has been set to  $v = 0.5$  m/s and the proportional gain in the  $\omega$  expression has been chosen as  $k_p = 1.2$ . The normalized step frequency was set to  $f = 0.95$  and the unicycle dynamics (1) was integrated over a time interval of  $\Delta T = 0.44$  s which, given the chosen value of the step frequency, corresponds to the time in which NAO completes one step. This was mandatory: a smaller value of  $\Delta T$  would have simply resulted in a shorter step, slowing down NAO's walk. The above value was determined experimentally and represents a reasonable tradeoff between accuracy of motion and gracefulness of walk.

The presented experiments were performed in a corridor tailored to the size of NAO. In the first experiment, NAO starts walking off the center of a straight corridor. Figure 9 reports four snapshots from the video accompanying the paper showing how NAO rapidly converges to the corridor center and keeps walking along it.

Snapshots from a right turn experiment are shown in Fig. 10. The robot stays safely at the center of the corridor even while turning around the corner, confirming the validity of the visual features replacement strategy described in Sect. III-B.

An analogous validation was the objective of the experiment reported in Fig. 11, which shows snapshots from a right turn at a T-junction. The visual features were computed using the strategy illustrated in Sect. III-C.

The three experiments are shown integrally in the video accompanying this paper. In particular, the live video feed from the robot camera with the reconstructed  $x_V$  and  $x_M$  is included in the clips.

## VI. CONCLUSIONS

We have presented a visual control approach for humanoid indoor navigation. In particular, the control objective is to follow a corridor walking as close as possible to its center, and executing safe turns at corners and junctions.

The proposed algorithm is based on a controller designed for unicycle-like mobile robots, which was extended to effectively cope with the presence of turns and junctions without changing the control strategy. Assuming that the robot is cruising at constant speed, the controller provides the angular velocity that corrects its position and orientation based on the visual feedback. The feedback signal is then transformed to



Fig. 9. First experiment: vision-based corridor navigation with off-centered start. The first two snapshots show NAO starting away from the corridor center but rapidly recovering it. The last two snapshots illustrate how NAO is able to keep walking at the center of the corridor.



Fig. 10. Second experiment: negotiating a turn. In the first snapshot NAO is approaching a left turn. The subsequent snapshots show how the robot correctly detects the corner and keeps the center of the corridor also during the turn.



Fig. 11. Third experiment: turning at a T-junction. The first snapshot shows NAO approaching the junction. When the junction is detected the robot takes the specified direction (second and third snapshot) and resumes walking at the center of the corridor (fourth snapshot).

inputs that are compatible with the locomotion system of the humanoid producing a natural, human-like walking behavior.

Experimental results obtained with a NAO humanoid have shown the validity of the approach and the effectiveness of the extended controller in negotiating turns and junctions.

Future work will include:

- image stabilization to compensate for the sway motion of the robot;
- more quantitative assessment of performance in terms of actual robot motion;
- validation of the controller in full-scale maze-like environments, possibly including obstacles in the corridor;
- extension of the visual navigation algorithm to the case of 4-way intersections.

#### REFERENCES

- [1] C. Lutz, F. Atmanspacher, A. Hornung, and M. Bennewitz, "Nao walking down a ramp autonomously," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012.
- [2] I. Back, J. Kallio, and K. Makela, "Enhanced map-based indoor navigation system of a humanoid robot using ultrasound measurements," *Intelligent Control and Automation*, vol. 3, no. 2, pp. 111–116, 2012.
- [3] J. Ido, Y. Shimizu, Y. Matsumoto, and T. Ogasawara, "Indoor navigation for a humanoid robot using a view sequence," *The International Journal of Robotics Research*, vol. 28, no. 2, pp. 315–325, 2009.
- [4] C. Dune, A. Herdt, E. Marchand, O. Stasse, P-B. Wieber, and E. Yoshida, "Vision based control for humanoid robots," in *2011 IROS Workshop on Visual Control of Mobile Robots*, pp. 19–26, 2011.
- [5] N. Courty, E. Marchand, and B. Araldi, "Through-the-eyes control of a virtual humanoid," in *The Fourteenth Conference on Computer Animation*, pp. 74–83, 2001.
- [6] J. M. Toibero, C. M. Soria, F. Roberti, R. Carelli, and P. Fiorini, "Switching visual servoing approach for stable corridor navigation," in *14th International Conference on Advanced Robotics*, pp. 1–6, 2009.
- [7] F. Launay, A. Ohya, and S. Yuta, "A corridors lights based navigation system including path definition using a topologically corrected map for indoor mobile robot," in *2002 IEEE International Conference on Robotics and Automation*, pp. 3918–3923, 2002.
- [8] G. Blanc, Y. Mezouar, and P. Martinet, "Indoor navigation of a wheeled mobile robot along visual routes," in *2005 IEEE International Conference on Robotics and Automation*, pp. 3354–3359, 2005.
- [9] R. Vassallo, H. J. Schneebeli, and J. Santos-Victor, "Visual servoing and appearance for navigation," *Robotics and Autonomous Systems*, vol. 31, no. 1–2, pp. 87–97, 2000.
- [10] Y. Park and H. Suh, "Predictive visual recognition of types of structural corridor landmarks for mobile robot navigation," in *19th IEEE International Symposium on Robot and Human Interactive Communication*, pp. 391–396, 2010.
- [11] K. Mombaur, A. Truong, and J.-P. Laumond, "From human to humanoid locomotion – an inverse optimal control approach," *Autonomous Robots*, vol. 28, pp. 369–383, 2010.
- [12] T.-V.-A. Truong, D. Flavigne, J. Pettre, K. Mombaur, , and J.-P. Laumond, "Reactive synthesizing of human locomotion combining nonholonomic and holonomic behaviors," in *3rd IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechanics*, pp. 632–637, 2010.
- [13] <http://opencv.org/>.
- [14] E. Yoshida, C. Esteves, I. Belousov, J.-P. Laumond, T. Sakaguchi, and K. Yokoi, "Planning 3d collision-free dynamic robotic motion through iterative reshaping," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1186–1198, 2008.
- [15] A. El Khouri, M. Taïx, and F. Lamiroux, "Path optimization for humanoid walk planning: an efficient approach," in *8th International Conference on Informatics in Control, Automation and Robotics*, pp. 179–184, 2011.