2012 12th IEEE-RAS International Conference on Humanoid Robots
Nov.29-Dec.1, 2012. Business Innovation Center Osaka, Japan

Vision-Based Odometric Localization for Humanoids using a Kinematic EKF

Giuseppe Oriolo, Antonio Paolillo, Lorenzo Rosa, Marilena Vendittelli

Abstract— **We propose an odometric system for localizing a walking humanoid robot using standard sensory equipment, i.e., a camera, an Inertial Measurement Unit, joint encoders and foot pressure sensors. Our method has the prediction-correction structure of an Extended Kalman Filter. At each sampling instant, position and orientation of the torso are predicted on the basis of the differential kinematic map from the support foot to the torso, using encoder data from the support joints. The actual measurements coming from the camera (head position and orientation reconstructed by a V-SLAM algorithm) and the Inertial Measurement Unit (torso orientation) are then compared with their predicted values to correct the estimate. The filter is made aware of the current placement of the support foot by an asynchronous update mechanism triggered by the pressure sensors. An experimental validation on the humanoid NAO shows the satisfactory performance of the proposed method.**

I. Introduction

Localization, as an essential feature of autonomous mobile robots, has been the subject of intensive research in the last decades. For single-body mobile robots, a simple form of localization is obtained by keeping track of motion displacements with proprioceptive sensors (e.g., wheel encoders or inertial sensors); this process, known as *dead reckoning* or *odometry*, provides a reasonably accurate position estimate in the short term, provided that the operating conditions are appropriate (e.g., for wheeled mobile robots, low velocities, flat floor, non-slippery terrain, etc.). Odometry is much lighter if compared with full-blown localization or SLAM (Simultaneous Localization And Mapping) methods, because it does not use or maintain a representation of the environment in the form of a map, and may be sufficient for autonomous short-range operation if appropriately complemented, e.g., by visual control. Moreover, odometry is used as a low-level module in many advanced localization schemes.

For multibody robots such as humanoids, however, pure odometry is hard to achieve, even in the most favorable conditions. This is due to several reasons, such as the the high number of degrees of freedom and their serial arrangement causing the amplification of actuation inaccuracies due to backlash or flexibility at the joints, the non-negligible dynamic effects appearing also at low speeds and generating undesired moments that can make the feet slip, the nature itself of biped locomotion which is characterized by a discontinuous contact with the ground, and others. As a consequence, odometric localization methods for humanoids must exploit the availability of exteroceptive sensors, such as cameras and range finders.

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, rosa, vendittelli}@dis.uniroma1.it

Existing methods for localizing humanoid robots can be roughly classified in three main approaches: (i) odometric localization (ii) localization over an a priori known map (iii) SLAM methods.

Most odometric localization methods for humanoid robots rely on visual information, and in particular on a technique known as Visual Odometry (VO), which tracks the apparent motion of visual features [1]. VO has been used to reconstruct the pose of cameras mounted on humanoid robots in [2] and [3]. In these works the camera is assumed to move freely, and thus the estimation procedure is not aware of the humanoid locomotion model. A visual odometry algorithm with improved robustness to motion blur due to walking motion is proposed in [4]. Also in this case, only the motion of the camera frame is estimated, without contribution from the other sensors of the robot. Although computationally light, pure visual odometry cannot provide feedback information at a sufficiently high frequency in a control loop when the image acquisition rate is low, as in low-cost humanoids.

There are also works that perform odometric localization using only encoder and inertial sensor data, such as [5], where odometry is used to reconstruct a 3D map of the environment by incorporating successive laser scans. As expected, the pose estimate is affected in this case by an error that builds up over time. Other odometric localization techniques are based on the comparison between the current camera image and previously registered images, e.g., see [6]. These methods, however, require previous information about the environment.

Methods for humanoid localization on a priori known maps of the environment are given in [7] and [8], in which measurements from a laser range finder are integrated with odometry computed from proprioceptive sensors.

Among the methods based on SLAM it is worth citing [9], which integrates walking pattern generator data in the prediction model of the Extended Kalman Filter (EKF) embedded in the monocular Visual SLAM (V-SLAM) algorithm proposed in [10]. A modification of the EKF component of a V-SLAM module is also proposed in [11], using information extracted from known landmarks and inertial measurements. In [12], a SLAM method is developed for humanoid navigation based on odometry and data from laser range finders mounted on the feet.

978-1-4673-1369-8/12/$31.00 ©2012 IEEE

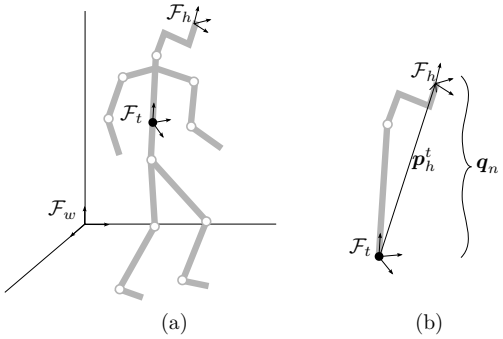

Fig. 1. Relevant frames: (a) $\mathcal{F}_w$ (world), $\mathcal{F}_t$ (torso) and $\mathcal{F}_h$ (head); (b) enlarged view of the torso-to-head kinematic chain with the definition of the neck joint variables $\boldsymbol{q}_n$.

In this paper, we devise an odometric localization method that maintains a consistent estimate of the pose (position and orientation) of a frame attached to the robot torso and can be used in unknown, unstructured environments. This is achieved using measurements from sensors that are found in the standard equipment of humanoid robots, i.e., cameras, inertial sensors, encoders and foot pressure sensors. In particular, visual information coming from the head camera is fed to a monocular V-SLAM algorithm that acts as a sensor enhancement and supplies a measurement of the head pose.

The choice of a V-SLAM algorithm rather than a visual odometry for the head pose reconstruction is motivated by its higher accuracy, obtained at the cost of an increased computational complexity which however does not preclude a real-time implementation. For example, localization systems integrating V-SLAM information with inertial data have been successfully developed for UAVs [13].

The structure of our algorithm is that of an Extended Kalman Filter, in which the estimate prediction is computed using the differential kinematics from the support foot to the torso and the relevant joint encoder readings. For the correction, we use as measurements the head pose coming from the V-SLAM algorithm and the torso orientation provided by the Inertial Measurement Unit. The filter is made aware of the current placement of the support foot by an asynchronous update mechanism triggered by the foot pressure sensors.

The paper is organized as follows. In the next section, we introduce the relevant definitions and formulate the odometric localization problem. Section III contains the description of the proposed method, while Sect. IV presents the experimental results obtained on the humanoid robot NAO. Section V concludes the paper and hints at some possible future work.

II. Problem formulation

Loosely speaking, *odometric localization* consists in maintaining a real-time estimate of the robot 'placement' in the world by keeping track of motion displacements measured via proprioceptive and/or exteroceptive sensors. In this section, we formalize this problem for a humanoid by introducing the basic geometry and defining the sensory equipment of the robot.

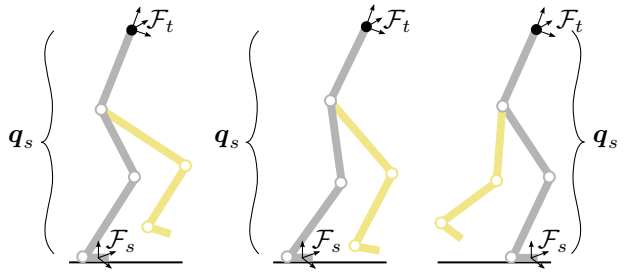

Fig. 2. During locomotion, the base of the kinematic chain is the support foot, whose representative frame $\mathcal{F}_s$ moves discontinuously upon each step completion. The identity of the support joints must be changed accordingly.

With reference to Fig. 1a, denote by $\mathcal{F}_w$ the (fixed) world frame and by $\mathcal{F}_t$, $\mathcal{F}_h$, respectively, the (moving) torso and head frames. Also denote with $\boldsymbol{p}_t$, $\boldsymbol{o}_t$ ($\boldsymbol{p}_h$, $\boldsymbol{o}_h$) the position and orientation of $\mathcal{F}_t$ ($\mathcal{F}_h$) with respect to $\mathcal{F}_w$. As shown in Fig. 1b, $\mathcal{F}_t$ and $\mathcal{F}_h$ are kinematically related through the neck joints, whose configuration is denoted by $\boldsymbol{q}_n$. In particular, we have

$$\boldsymbol{p}_h = \boldsymbol{p}_t + \boldsymbol{R}_t \, \boldsymbol{p}_h^t \tag{1}$$

and

$$\boldsymbol{o}_h = \boldsymbol{\Omega}(\boldsymbol{R}_h) = \boldsymbol{\Omega}(\boldsymbol{R}_t \boldsymbol{R}_h^t), \tag{2}$$

where $\boldsymbol{R}_t$ is the rotation matrix from $\mathcal{F}_w$ to $\mathcal{F}_t$, $\boldsymbol{p}_h^t$ is the position of $\mathcal{F}_h$ with respect to $\mathcal{F}_t$, $\boldsymbol{\Omega}(\cdot)$ is a function that extracts the value of the orientation in the chosen coordinates (e.g., R-P-Y angles) from a rotation matrix, and $\boldsymbol{R}_h^t$ is the rotation matrix from $\mathcal{F}_t$ to $\mathcal{F}_h$. Clearly, both $\boldsymbol{p}_h^t$ and $\boldsymbol{R}_h^t$ are functions of the neck joint angles $\boldsymbol{q}_n$.

In the method to be proposed, an important part will be played by kinematic computations. In particular, during the single support phase of a step, these computations will hinge on the support foot, which represents the base of an open kinematic chain whose endpoint is the origin of $\mathcal{F}_t$. For this reason, we attach a *support frame* $\mathcal{F}_s$ to that foot; moreover, we define *support joints*[1] those located between $\mathcal{F}_s$ and $\mathcal{F}_t$, and denote their configuration by $\boldsymbol{q}_s$. As shown in Fig. 2, at the completion of each step $\mathcal{F}_s$ jumps to a new placement, which is the current placement of the former swinging foot; accordingly, the support joints must also be redefined.

Coming to the sensory equipment, we consider a rather standard gear for a humanoid, i.e., a monocular camera located in the robot head, an Inertial Measurement Unit (IMU) mounted on the torso, joint encoders and foot pressure sensors. While the last two are essential for kinematic computations, only the camera and the IMU will actually appear in the measurement model of our localization filter. In particular:

- An off-the-shelf V-SLAM algorithm is used to compute a measure of the position and orientation $\boldsymbol{p}_h$, $\boldsymbol{o}_h$ of the head frame $\mathcal{F}_h$ from camera images. The combination of camera and V-SLAM algorithm is considered as an intelligent sensor described by a black box.

[1]These include the joints of the support leg and those of the pelvis.

- The only IMU measurement used is the orientation $\boldsymbol{o}_t$, because velocity data are often too noisy and sometimes simply unavailable, as in the case of the NAO humanoid used for our experiments.

Our version of the odometric localization problem can now be defined more precisely. Given initial estimates $\hat{\boldsymbol{p}}_{t,0}, \hat{\boldsymbol{o}}_{t,0}$ for the position and orientation of the torso, update $\hat{\boldsymbol{p}}_t, \hat{\boldsymbol{o}}_t$ continuously as the humanoid moves, using measurements of $\boldsymbol{p}_h$, $\boldsymbol{o}_h$ coming from the camera plus V-SLAM, measurements of $\boldsymbol{o}_t$ coming from the IMU, joint readings from the encoders and foot pressure sensors to detect step completion.

Clearly, once a reliable estimate is known for the pose of the torso, one may reconstruct the pose of any other part of the humanoid body through direct kinematics.

III. The proposed method

Our proposed method for vision-based odometric localization of a humanoid has the prediction-correction structure of an Extended Kalman Filter. In short, at each sampling instant, a pose for the torso is predicted using the differential kinematic map from $\mathcal{F}_s$ to $\mathcal{F}_t$; in this phase, joint encoders data for the support joints are used. The prediction is then corrected on the basis of the difference between the actual measurements coming from the camera (head position and orientation reconstructed by a V-SLAM algorithm) plus the IMU (torso orientation) and their expected values. The filter is made aware of the current placement of the support foot by an asynchronous update mechanism triggered by the foot pressure sensors. Below, we detail this conceptual structure.

Collect the position and orientation of $\mathcal{F}_t$ with respect to $\mathcal{F}_w$ in a *pose* vector $\boldsymbol{x} = (\boldsymbol{p}_t, \boldsymbol{o}_t)$, which will be the state to be estimated by our filter. As a state-transition model for $\boldsymbol{x}$ we will adopt the following:

$$\dot{\boldsymbol{x}} = \boldsymbol{J}(\boldsymbol{q}_s, \boldsymbol{o}_s)\dot{\boldsymbol{q}}_s, \tag{3}$$

where $\boldsymbol{J}(\boldsymbol{q}_s, \boldsymbol{o}_s)$ is the Jacobian matrix of the kinematic map from $\mathcal{F}_s$ to $\mathcal{F}_t$ and $\boldsymbol{o}_s$ denotes the orientation of $\mathcal{F}_s$. This is a *kinematic*[2] model, in that the support joint velocities $\dot{\boldsymbol{q}}_s$ act as control inputs. Note that the evolution of $\boldsymbol{q}_s$ and $\boldsymbol{o}_s$ are not described by this model. The value of $\boldsymbol{q}_s$ will be simply read from the corresponding joint encoders. As for $\boldsymbol{o}_s$, its dynamics is asynchronous, because the support foot placement is constant during a step and changes discontinuously upon its completion. A separate mechanism will be then used to update $\boldsymbol{o}_s$ when needed.

The output model expresses the measured variables $\boldsymbol{y}$ as a function $\boldsymbol{h}$ of the system state $\boldsymbol{x}$. In our case, these variables are the head pose reconstructed by the V-SLAM algorithm and the torso orientation measured by the IMU. Therefore

[2]The reasons for not using a truly *dynamic* state-transition model are essentially two. The first is that the dynamic equations of a humanoid are vey complex, often unknown and in any case their integration would be time-consuming. The second is that the inputs of such a model would be the joint torques, whose measurements are typically unavailable.

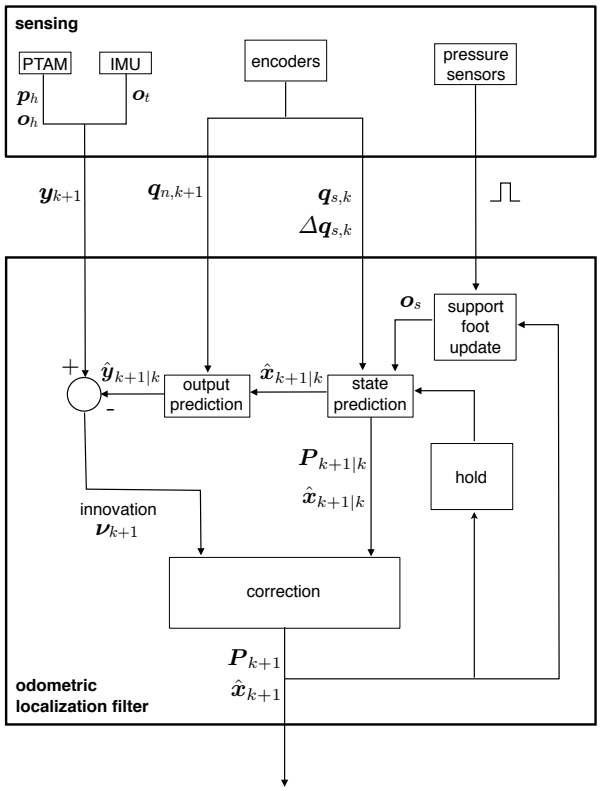

Fig. 3. A block scheme of the proposed odometric localization method. Note the asynchronous update mechanism triggered by the pressure sensors used to make the filter aware of the current orientation of the support foot.

we have:

$$\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{q}_n) = \begin{pmatrix} \boldsymbol{p}_t + \boldsymbol{R}_t\, \boldsymbol{p}_h^t \\ \boldsymbol{\Omega}(\boldsymbol{R}_t\, \boldsymbol{R}_h^t) \\ \boldsymbol{o}_t \end{pmatrix}, \tag{4}$$

having used eqs. (1–2). Note that the output is also a function of the neck joints configuration through $\boldsymbol{p}_h^t$ and $\boldsymbol{R}_h^t$. Like $\boldsymbol{q}_s$ in the state-transition model, the value of $\boldsymbol{q}_n$ is read from the joint encoders.

Using a subscript k to denote the value of a variable at time kT, where T is the sampling time, a discrete-time perturbed system can be associated to model (3–4) as follows:

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + T\boldsymbol{J}(\boldsymbol{q}_{s,k}, \boldsymbol{o}_s)\dot{\boldsymbol{q}}_{s,k} + \boldsymbol{v}_k \tag{5}$$

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{q}_{n,k}) + \boldsymbol{w}_k, \tag{6}$$

where $\boldsymbol{v}_k$, $\boldsymbol{w}_k$ are white gaussian noises with zero mean and covariance matrices $\boldsymbol{V}_k \in \mathbb{R}^{6\times6}$, $\boldsymbol{W}_k \in \mathbb{R}^{9\times9}$, respectively. Once again, note how the orientation $\boldsymbol{o}_s$ of the support frame is asynchronous with respect to all the other variables.

We are now ready to write down the equations of the proposed odometric localization filter, whose block scheme is illustrated in Fig. 3.

A. State prediction

The state prediction $\hat{\boldsymbol{x}}_{k+1|k}$ is generated from the current estimate using eq. (5):

$$\hat{\boldsymbol{x}}_{k+1|k} = \hat{\boldsymbol{x}}_k + \boldsymbol{J}(\boldsymbol{q}_{s,k}, \boldsymbol{o}_s)\Delta\boldsymbol{q}_{s,k},$$

where the encoder reading difference $\Delta \boldsymbol{q}_{s,k} = \boldsymbol{q}_{s,k+1} - \boldsymbol{q}_{s,k}$ is used to approximate the velocity input term $T\dot{\boldsymbol{q}}_{s,k}$.

Linearization of the discrete-time state equation (5) immediately leads to the following covariance prediction:

$$\boldsymbol{P}_{k+1|k} = \boldsymbol{P}_k + \boldsymbol{V}_k. \tag{7}$$

B. Output prediction

The predicted output is calculated as

$$\hat{\boldsymbol{y}}_{k+1|k} = \boldsymbol{h}(\hat{\boldsymbol{x}}_{k+1|k}, \boldsymbol{q}_{n,k+1}),$$

with $\boldsymbol{h}(\cdot)$ defined in eq. (4). This computation requires the value of the neck joints, read via the corresponding encoders.

C. Correction

Correction of $\hat{\boldsymbol{x}}_{k+1|k}$ is performed based on the value of the innovation, i.e., the difference between the measured and the predicted output:

$$\boldsymbol{\nu}_{k+1} = \boldsymbol{y}_{k+1} - \hat{\boldsymbol{y}}_{k+1|k}.$$

The state estimate is then corrected as follows:

$$\hat{\boldsymbol{x}}_{k+1} = \hat{\boldsymbol{x}}_{k+1|k} + \boldsymbol{G}_{k+1}\boldsymbol{\nu}_{k+1},$$

where $\boldsymbol{G} \in \mathbb{R}^{6\times 9}$ is the Kalman gain matrix:

$$\boldsymbol{G}_{k+1} = \boldsymbol{P}_{k+1|k}\boldsymbol{H}_{k+1}^T(\boldsymbol{H}_{k+1}\boldsymbol{P}_{k+1|k}\boldsymbol{H}_{k+1}^T + \boldsymbol{W}_{k+1})^{-1}$$

and

$$\boldsymbol{H}_{k+1} = \left.\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\hat{\boldsymbol{x}}_{k+1|k}}.$$

The actual expression of $\boldsymbol{H}_{k+1}$ will depend on the choice of coordinates for representing orientations.

Finally, the correction equation for the covariance matrix is

$$\boldsymbol{P}_{k+1} = \boldsymbol{P}_{k+1|k} - \boldsymbol{G}_{k+1}\boldsymbol{H}_{k+1}\boldsymbol{P}_{k+1|k}.$$

D. Support foot update

When the foot pressure sensors detect a transition from a single to a double support phase (i.e., completion of a step), the support frame is displaced to the next support foot. Since the differential kinematics (3) do not depend on $\boldsymbol{p}_s$, only the orientation $\boldsymbol{o}_s$ of $\mathcal{F}_s$ is updated[3]. To this end, a forward kinematic computation is carried out from $\mathcal{F}_t$ (which now acts as a base frame placed at its current estimated pose) to $\mathcal{F}_s$, using the encoders to read the value of the new support joints (see Fig. 2).

[3]If the humanoid is moving on a flat floor, only the yaw angle of $\mathcal{F}_s$ needs to be updated.

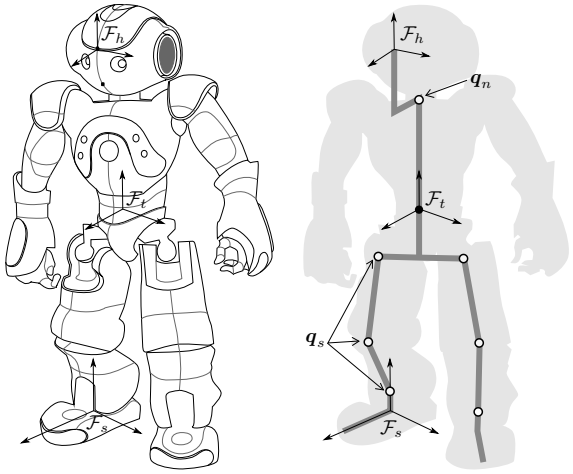

Fig. 4. NAO and the kinematic chains of interest in this paper.

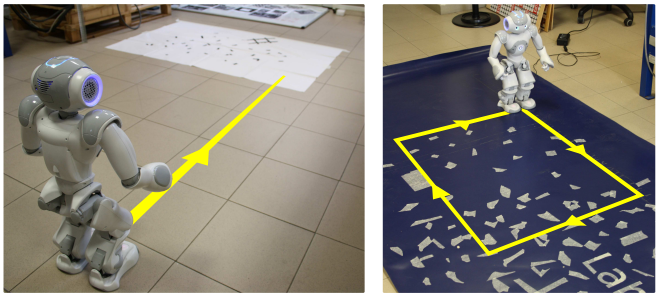

Fig. 5. Experimental scenarios: The robot is commanded to walk along a straight line in the first experiment (left) and a square path in the second experiment (right).

IV. Experimental results

The experimental platform used to validate our odometric localization method is the NAO humanoid developed by Aldebaran Robotics. Figure 4 shows the robot with the frames and the kinematic chains of interest for the present study. There are 5 degrees of freedom in each leg, 1 in the pelvis, and 2 in the neck; therefore, we have $\boldsymbol{q}_s \in \mathbb{R}^6$ and $\boldsymbol{q}_n \in \mathbb{R}^2$. The built-in sensors we used are: the CMOS digital camera with a 58° field of view mounted on the forehead; the IMU located in the chest, which provides a measure of the torso roll and pitch angles but no yaw (hence the output vector $\boldsymbol{y}$ is actually 8-dimensional in our case); the magnetic rotary encoders with 0.1° resolution available at the joints; and the force sensitive resistors located under the feet, that measure the applied pressure through a change in resistance.

As a V-SLAM module, we selected the PTAM algorithm [14]. To avoid overburdening the robot CPU, images at 320×240 resolution are sent to an external computer that runs PTAM and makes available a measurement of the pose of NAO's head at 13 Hz. Thus, our kinematic EKF runs at the same rate, although multiple prediction steps are taken to exploit the fact that IMU and encoder readings are available at a higher rate (about 93 Hz). The support foot is updated asynchronously when the force sensitive resistors indicate transition from single to double support.

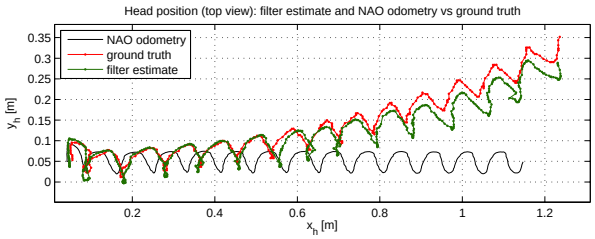

Fig. 6. First experiment. Head position: comparison among filter estimate, built-in NAO odometry and ground truth.

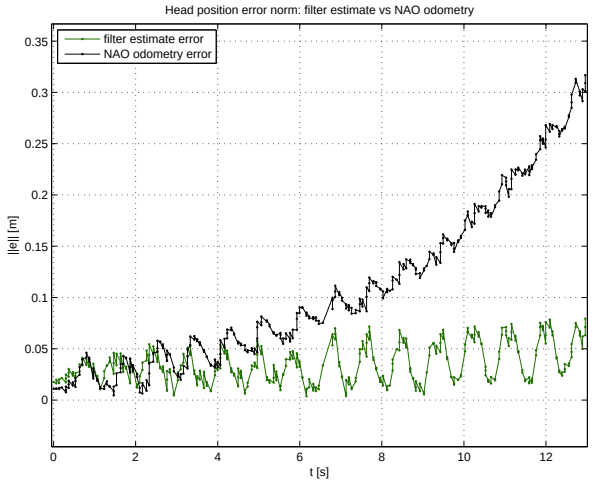

Fig. 7. First experiment. Head position error norm: comparison between filter estimate and built-in NAO odometry.

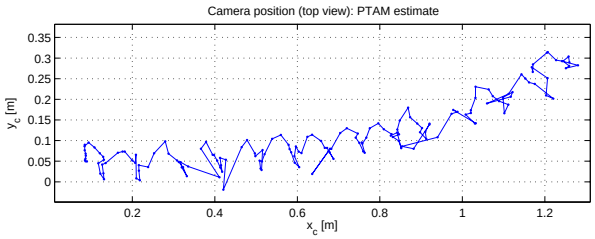

Fig. 8. First experiment. Camera position: estimates computed by PTAM algorithm.

Our experiments begin with an initialization phase, in which NAO performs a sideways motion on the spot to collect visual information and allow PTAM to recover the metric scale. After that, motion commands are sent to the robot using the NAO APIs, and in particular the *setWalkTargetVelocity* function. This is an open-loop command, and therefore the robot will soon veer off the nominal course due to non-idealities (most importantly, foot slippage). To obtain a baseline for performance assessment, we used a ceiling camera to record the whole experiment. An automatic feature tracker extracts the x_h, y_h coordinates of the head from the images, thus providing a ground truth; this is compared[4] with the value of the same coordinates reconstructed using the filter estimate for the torso pose and computing forward kinematics from the torso to the head. For comparison, we also stored the built-in NAO odometry provided by the *getPosition* function.

We report the results of two experiments. On the basis of empirical observations, the covariance matrices were set to:

$$\begin{aligned} \boldsymbol{V} &= \text{diag}\{2.5, 2.5, 2.5, 10, 10, 10\} \cdot 10^{-5} \\ \boldsymbol{W} &= \text{diag}\{25, 25, 25, 1, \ldots, 1\} \cdot 10^{-2} \end{aligned}$$

using meters for positions and radians for orientations. In the first experiment, the robot is commanded to walk in straight line at a velocity of 0.1 m/s (see Fig. 5, left). As shown in Figs. 6–7, the proposed odometric localization performs rather well, providing an estimate close to the ground truth. The built-in NAO odometry does not reflect the fact that the robot departs from the nominal path. In terms of RMS of the error, we have a value of 0.039242 m and 0.134535 m, respectively, for the proposed filter and the built-in odometry. Looking at the camera[5] position estimated by PTAM, shown in Fig. 8, one may appreciate the benefits of the proposed filter in smoothing this estimate and making it consistent with a humanoid motion.

In the second experiment, the robot is commanded to walk along a square path at 0.1 m/s (see Fig. 5, right). In particular, to test the filter under the full omnidirectional walk of which NAO is capable, the robot marches forward along the first side, sideways (right) along the second, backwards along the third and sideways (left) along the fourth. The results, shown in Figs. 9–11, show again that the proposed odometric localization scheme is quite accurate and outperforms the built-in odometry. In particular, the RMS of the error is 0.049657 m with the former and 0.128314 m with the latter.

Movie clips of the above two experiments are included in the video accompanying this paper. We have extensively tested our odometric localization algorithm in different environments and settings, obtaining similarly good results. In particular, the filter proved to be quite robust to changes in the commanded motion of the robot as well as to temporary failures of the V-SLAM algorithm, even in the absence of a specific mechanism for detecting these failures and rejecting the associated measurements.

V. Conclusions

We have described a method for odometric localization of a humanoid robot using visual information. The proposed algorithm has the prediction-correction structure of an Extended Kalman Filter. At each sampling instant, a pose for the torso is predicted using the differential kinematic map from the support foot to torso itself; in this phase, joint encoders data for the support joints are used. The prediction is then corrected on the basis of the difference between the actual measurements coming from the camera (head position and orientation reconstructed by a V-SLAM algorithm) plus the

[4]This comparison does not directly concern the orientation estimate for the torso. However, the head coordinates clearly depend on such estimate through the torso-to-head kinematics. Moreover, our objective is to design an odometric localization method for locomotion tasks, and therefore a simple cartesian validation appears to be sufficient for a preliminary analysis.

[5]There is in fact a small displacement between the head and the camera frames.

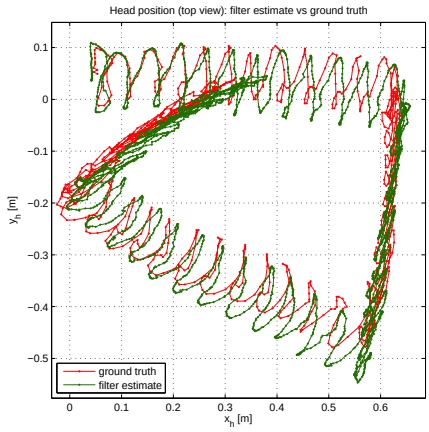

Fig. 9. Second experiment. Head position: comparison between filter estimate and ground truth.

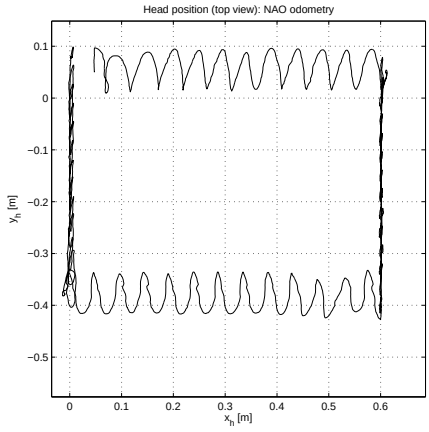

Fig. 10. Second experiment. Head position: built-in NAO odometry.

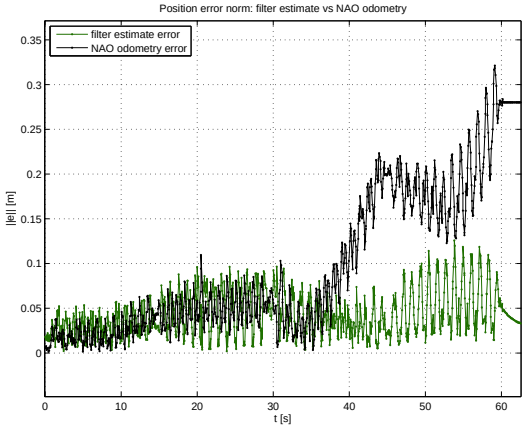

Fig. 11. Second experiment. Head position error norm: comparison between filter estimate and built-in NAO odometry.

IMU (torso orientation) and their expected values. The filter is made aware of the current placement of the support foot by an asynchronous update mechanism triggered by the foot pressure sensors. Experimental results on a NAO humanoid have been presented to show the satisfactory performance of the method, in spite of the use of a single low-cost camera. Even better results can be expected, e.g., if a stereo camera system is available on the robot.

Future work will be aimed at several objectives, including:

- bringing the whole localization system on-board by implementing a lightweight version of PTAM (e.g., see [13]), as well as trying other off-the-shelf V-SLAM algorithms for estimating the pose of the head (for example, the multiple mapping capability of PTAMM [15] may be beneficial for long-range localization);
- accurately characterizing the noise associated to PTAM, and incorporating into the EKF the resulting variable uncertainty;
- performing experiments with variable elevation, such as walking up slopes or stairs;
- using the odometric localization data to close the control loop and achieve a robust navigation behavior.

References

[1] D. Scaramuzza and F. Fraundorfer, "Visual odometry Part I: The first 30 years and fundamentals," *IEEE Robotics & Automation Magazine*, vol. 18, no. 4, pp. 80–92, 2011.

[2] Y. Takaoka, Y. Kida, S. Kagami, H. Mizoguchi, and T. Kanade, "3D map building for a humanoid robot by using visual odometry," in *2004 IEEE Int. Conf. on Systems, Man, and Cybernetics*, vol. 5, pp. 4444–4449, 2004.

[3] R. Ozawa, Y. Takaoka, Y. Kida, K. Nishiwaki, J. Chestnutt, J. Kuffner, J. Kagami, H. Mizoguch, and H. Inoue, "Using visual odometry to create 3D maps for online footstep planning," in *2005 IEEE Int. Conf. on Systems, Man, and Cybernetics*, vol. 3, pp. 2643–2648, 2005.

[4] A. Pretto, E. Menegatti, M. Bennewitz, W. Burgard, and E. Pagello, "A visual odometry framework robust to motion blur," in *2009 IEEE Int. Conf. on Robotics and Automation*, pp. 2250–2257, 2009.

[5] J. Chestnutt, Y. Takaoka, K. Suga, K. Nishiwaki, J. Kuffner, and S. Kagami, "Biped navigation in rough environments using on-board sensing," in *2009 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 3543–3548, oct. 2009.

[6] 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.

[7] A. Hornung, K. M. Wurm, and M. Bennewitz, "Humanoid robot localization in complex indoor environments," in *2010 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 1690–1695, 2010.

[8] S. Thompson, S. Kagami, and K. Nishiwaki, "Localisation for autonomous humanoid navigation," in *2006 IEEE-RAS Int. Conf. on Humanoid Robots*, pp. 13–19, 2006.

[9] O. Stasse, A. C. Davison, R. Sellaouti, and K. Yokoi, "Real-time 3D SLAM for a humanoid robot cosidering pattern generator information," in *2006 IEEE/RSJ Int. Conf. on Intellingent Robots and Systems*, pp. 348–355, 2006.

[10] A. J. Davison, "Real-time simultaneous localisation and mapping with a single camera," in *9th Int. Conf. on Computer Vision*, pp. 1403–1410, 2003.

[11] E. Hernandez, J. M. Ibarra, J. Neira, R. Cisneros, and J. E. Lavín, "Visual SLAM with oriented landmarks and partial odometry," in *21st IEEE Int. Conf. on Electrical Communications and Computers*, pp. 39–45, 2011.

[12] R. Tellez, F. Ferro, D. Mora, D. Pinyol, and D. Faconti, "Autonomous humanoid navigation using laser and odometry data," in *2008 IEEE-RAS Int. Conf. on Humanoid Robots*, pp. 500–506, 2008.

[13] S. Weiss, D. Scaramuzza, and R. Siegwart, "Monocular-SLAM–based navigation for autonomous micro helicopters in GPS-denied environments," *Journal of Field Robotics*, vol. 28, no. 6, pp. 854–874, 2011.

[14] G. Klein and D. Murray, "Parallel tracking and mapping for small AR workspaces," in *6th IEEE and ACM Int. Symp. on Mixed and Augmented Reality*, pp. 225–234, 2007.

[15] R. O. Castle, G. Klein, and D. W. Murray, "Video-rate localization in multiple maps for wearable augmented reality," in *12th IEEE Int. Symp. on Wearable Computers*, pp. 15–22, 2008.