2012 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 7-12, 2012. Vilamoura, Algarve, Portugal

On-board Odometry Estimation for 3D Vision-based SLAM of Humanoid Robot

SungHwan Ahn, Sukjune Yoon, Seungyong Hyung, Nosan Kwak, and Kyung Shik Roh

***Abstract*— This paper addresses a vision-based 3D motion estimation framework for humanoid robots, which copes with human-like walking pattern. A humanoid robot, called Roboray, is designed for dynamic walking control with heel-toe motion like a human. In spite of stability and energy efficiency of the dynamic walking, it accompanies larger swaying motion and more uncertainty in camera movement than the conventional ZMP (Zero Moment Point)-based walking does. The framework effectively uses on-board odometry information from the robot to improve the performance of the vision-based motion estimation. To accomplish this, we propose an on-board odometry filter which fuses kinematic odometry, visual odometry, and raw IMU data. And the odometry filter is combined with vision-based SLAM to provide accurate motion model, so it enhances the SLAM estimates. Experimental results in indoor environment verify that the framework can successfully estimate the pose of Roboray in real-time.**

I. Introduction

Recently, a number of biped humanoid robots have emerged and their walking and motion planning methods have been stabilized in performance [1], [2]. As development of humanoid robots which aims to mimic human motion is more emphasized, novel methods such as dynamic walking control (torque servo-based method) [3] and heel-toe motion generation [4] that involve human-like locomotion are introduced. They are essential for navigation and exploration in more realistic environment because of stability and energy efficiency in walking [5].

To achieve accurate localization and geometric representation of the world for humanoid robots, vision-based SLAM (Simultaneous Localization And Map building) is widely used due to small size, lower energy consumption and rich information of camera. Many good SLAM solutions are already being proposed in mobile robotics research [6] or hand-held devices [7]. However, they still have challenging problems for humanoid robots because they should deal with the inherited properties of erratic swaying motion, various uncertainty, and resulting motion blur.

The following methods try to make use of vision-based approaches for humanoid robot. Various studies on visual odometry [8], [9] and localization with a known map [10] were proposed and evaluated. In a seminal work, Stasse *et al.* [11] utilized the monoSLAM system [12] for HRP-2. It was the first implementation of real-time vision-based SLAM for a humanoid robot assisted by pattern generator and inertial sensing. The other SLAM approach [13] used a stereo camera to get 3D point cloud data. Then robust Rao-Blackwellized particle filter was applied to estimate the robot pose and the shape of a 3D grid map concurrently.

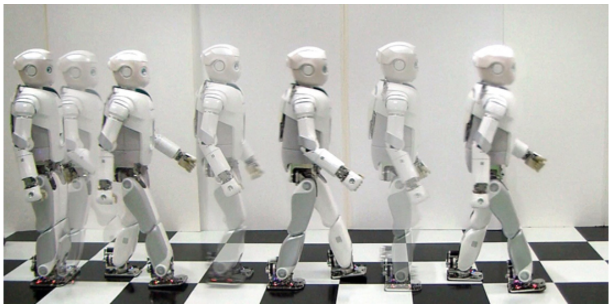


Fig. 1. A multiple frame shot of dynamic walking Roboray

Above methods work well in the conventional position servo-based method like ZMP (Zero Moment Point)-based walking control which has relatively small and smooth movement of the upper body of the robot. However, when we apply them to a dynamic walking robot, especially with heel-toe motion, in our tests, they cannot be well-conditioned to estimate the robot pose any more. The reason is that they basically assume a motion model of smooth camera movement.

In this paper, we propose a vision-based 3D motion estimation method integrated with on-board odometry information for a human-like walking robot. It consists of two modules, EKF (Extended Kalman Filter)-based odometry estimation module and vision-based SLAM module. 1) In the odometry module, forward kinematics model of the robot and encoder values of each joint are used to obtain kinematic odometry which means the robot pose relative to a starting point. Visual odometry is also calculated with sequential stereo images to provide additional information to the odometry estimation. Then, a combination of gyro rate and acceleration measured from IMU (Inertial Measurement Unit) sensor is fused with the relative pose data obtained from the kinematic odometry and the visual odometry. 2) The output of the fused motion estimation, which operates in real-time at up to $100Hz$, is applied to the vision-based SLAM module via the form of relative pose change and its covariance. It improves the SLAM estimates in that it can compensate motion errors and provide more accurate prediction model.

The proposed vision-based SLAM method coupled with

S. Ahn, S. Yoon, S. Hyung, N. Kwak, and K. S. Roh are with Samsung Advanced Institute of Technology (SAIT), Samsung Electronics Co., LTD., Giheung, 446-712, Korea {sunghwan.ahn, sukjune.yoon, seungyong.hyung, nosan.kwak, kyung.roh}@samsung.com

978-1-4673-1735-1/12/S31.00 ©2012 IEEE

odometry information has the following properties.

1) It is unnecessary to set up additional sensors to obtain odometry information. The on-board sensors such as joint encoders, force sensors and IMU are only requirements to implement the odometry. Humanoid robots are commonly equipped with those sensors for walking and balancing.
2) It successfully copes with rapid inter-image changes that are observed while the robot is walking. It enriches the performance of visual feature tracking by consistently providing the changes of the robot pose even with heel-strike and toe-off walking motion.
3) It can reduce adverse effect of odometry error on SLAM by maintaining two separate modules. When failure in odometry estimation is detected, the odometry module is simply restarted and so the SLAM module can receive the odometry information consistently. Consequently, it helps to reduce estimation errors and increase a chance of loop closure of SLAM.

This paper is organized as follows. A problem definition of vision-based motion estimation for a human-like walking robot is explained in Section II and on-board odometry estimation is organized in Section III. Section IV presents a 3D vision-based SLAM method for a humanoid robot, combining with the on-board odometry information. Then, Section V shows the experimental results of the proposed method with our humanoid robot, Roboray, and conclusion follows.

II. Dynamic Walking Control with Heel-toe Motion Generation

A. A Biped Humanoid Robot, Roboray

We developed a biped humanoid robot, Roboray, whose height and weight are $150cm$ and $60kg$ respectively. It has a total of 54 DOFs which are 6 for each leg, 1 for the torso, 3 for the head, 7 for each arm, and 12 for each hand. The legs are designed for having compliant joints by adopting cable-driven actuator mechanism with low control gain.

The joint compliance of Roboray is helpful for its dynamic walking control to allow the robot to stably interact with the ground by dissipation; however, it has more chances to increase swaying motion of the camera, located at the head, especially for human-sized Roboray.

B. Dynamic Walking Control

Dynamic walking control of humanoid robots, called as torque servo-based method, controls the joint torque to achieve the desired torque to walk. For Roboray, an adaptive control framework of dynamic walking is developed [5]. It has advantages in energy efficiency and stability to surroundings with human-like walking pattern. Figure 1 is a multiple frame shot of Roboray's walking, which shows apparently human-like knee-stretched locomotion.

The dynamic walker, on the other hand, has more disadvantages to implement vision-based approaches than the conventional ZMP walker. While walking, its periodic controlled-falling motion and its stretched knee make the camera to sway back and forth like an inverted pendulum. As a result, rapid change between consecutive images and motion blur in an image shot more frequently happen in the dynamic walker.

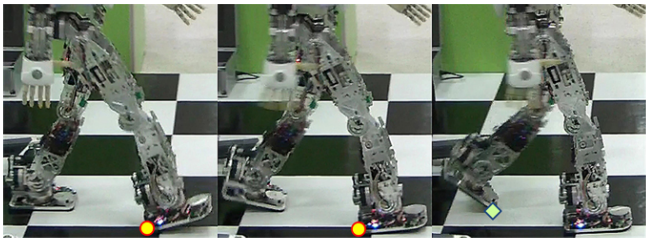


Fig. 2. Heel-toe motion, which is comprised of heel-strike (circle) and toe-off (diamond) motion, generated by dynamic walking Roboray

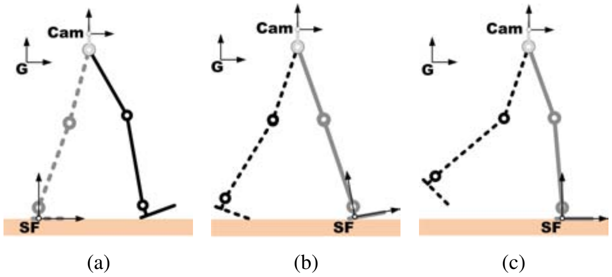


Fig. 3. Illustration of the heel-toe motion of Fig. 2, which are organized into (a) heel-strike phase, (b) opposite heel-rise phase, and (c) opposite toe-off phase

C. Heel-toe Motion Generation

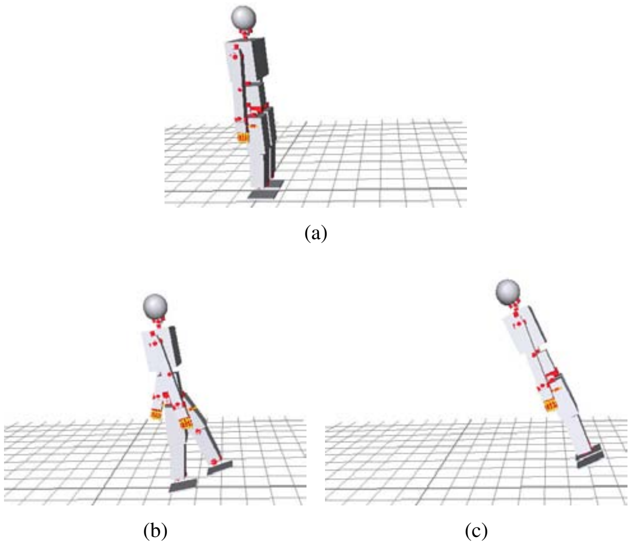
A state machine controller [5] of Roboray is able to implement various walking styles including heel-toe motion by using gait primitives. It helps Roboray to walk with a natural gait cycle in the same manner of a human [14]. The gait cycle plays an important role in force absorption and transmission for walking [15]. The heel-toe motion of Roboray on a flat ground is shown in Fig. 2. It consists of three phases, 1) heel-strike phase (The heel of the right foot initially contacts the ground.), 2) opposite heel-rise phase (The left foot is in the preparation stage for swinging.), and 3) opposite toe-off phase (The right foot fully contacts on the ground, then the toe of the left foot takes off.)

When the heel-toe motion is generated, the height of the camera is no longer constrained to be constant in sagittal plane because the pelvis of the robot goes up and down repeatedly. Moreover, the heel-strike phase increases the impact of the foot on the ground and it leads to fluctuation in the motion of the camera. They aggravate the aforementioned problems of the dynamic walking control, plus the vision-based approaches cannot utilize the assumption of smooth motion of camera any more.

III. On-board Odometry Information Fusion

A. Kinematic Odometry Information

In case of a humanoid robot, odometry known as dead-reckoning can be easily obtained by forward kinematics with the length of each link and the measured angle of each joint.



(a)

(b)

(c)

Fig. 4. Resulting motion estimation of the kinematic odometry [(a)→(b) →(c)]. It was tested on real sensor data acquired by Roboray. Due to inadequate ground contact of heel-toe motion, Roboray gets to believe that it is going up in the air.

Figure 3 illustrates the walking sequence of Fig. 2 to describe odometry calculation in detail. The stance foot (gray), which has a reference axis (SF) for forward kinematics, is switched between the left foot (dotted line) and the right foot (solid line) as the robot walks. It can be determined by the ground contact condition which can be measured from the force sensor on the ankle.

In this paper, the kinematic odometry denotes the camera pose with respect to the global coordinate, G, at time k as

$$ {}^{G}\mathbf{T}_{Cam}(k) = \begin{bmatrix} {}^{G}\mathbf{r}_{Cam}(k) & {}^{G}\mathbf{q}_{Cam}(k) \end{bmatrix}^T \tag{1}$$

where ${}^{G}\mathbf{r}_{Cam}$ is position and ${}^{G}\mathbf{q}_{Cam}$ is orientation with quaternion. The kinematic odometry can be recursively calculated via a 6DOF rigid body transformation, $\otimes$:

$$ {}^{G}\mathbf{T}_{SF}(k-1) = {}^{G}\mathbf{T}_{Cam}(k-1) \otimes {}^{Cam}\mathbf{T}_{SF}(k-1) \tag{2}$$

$$ {}^{G}\mathbf{T}_{Cam}(k) = {}^{G}\mathbf{T}_{SF}(k-1) \otimes {}^{SF}\mathbf{T}_{Cam}(k). \tag{3}$$

First, the pose of the stance foot with respect to G at time $(k-1)$ is obtained from previous forward kinematics. Then, the kinematic odometry is updated by using the relative pose between the stance foot and the camera at time k.

Especially with the dynamic walking control, the desired trajectory of the pattern generator of the ZMP walking [11] cannot be utilized for odometry calculation because of the nature of the torque control method. For that reason, this kinematic odometry becomes more useful for motion estimation of dynamic walking Roboray. However, the accuracy of the kinematic odometry highly depends on contact condition between the foot and the ground, and it is strongly affected by considerable uncertainty resulting from slippage and friction of the foot. In addition, the heel-strike phase and the opposite heel-rise phase could disturb the stance foot to be fully contacted on the ground as shown in Fig. 3(b). It makes the reference axis (SF) not to be located parallel to the ground, and the uncertainty in the foot contact point leads to inaccurate motion estimation like Fig. 4.

B. Visual Odometry Information

To make up for the kinematic odometry and improve ego-motion estimation, performing what is known as visual odometry is addressed. For easier implementation, it makes use of calibrated stereo camera installed on Roboray's head. Basically our visual odometry adopts a generally known approach [16] which is comprised of two steps; 1) incremental visual odometry that accomplishes initial relative pose estimation by using RANSAC-based three point algorithm and 2) local bundle adjustment over recent 5 images to refine the relative estimation.

The vision-only method is still suffering in the aforementioned human-like walking pattern because feature tracking could be frequently lost in many cases of swaying motion. Nevertheless, from the point of view of sensor fusion, it is important that estimate covariance of uncertainty is additionally available through the parameter optimization of the bundle adjustment as well as the relative pose of camera obtained from inter-image changes.

C. Fusion with raw IMU Information

The ego-motion estimation of on-board odometry is organized based on an expanded idea of the IMU filter of quaternion estimation [17] which uses acceleration and gyro rate as a control input.

1) IMU prediction: The estimate state $\mathbf{x}$ is composed of position ${}^{G}\mathbf{r}$, orientation ${}^{G}\mathbf{q}$, and velocity ${}^{G}\mathbf{v}$ with respect to the global coordinate. Also acceleration bias ${}^{B}\delta\mathbf{a}$ and gyro rate bias ${}^{B}\delta\boldsymbol{\omega}$ with respect to the body coordinate of stereo camera is added to have more accurate estimation.

$$ \mathbf{x} = \begin{bmatrix} {}^{G}\mathbf{r} & {}^{G}\mathbf{q} & {}^{G}\mathbf{v} & {}^{B}\delta\mathbf{a} & {}^{B}\delta\boldsymbol{\omega} \end{bmatrix}^T \tag{4}$$

In order to implement the prediction process with EKF, the derivative of state $\mathbf{x}$ is derived from kinematic relations with zero-mean additive Gaussian noise $\boldsymbol{\nu}$ as follows:

$$ \dot{\mathbf{x}} = \begin{bmatrix} {}^{G}\dot{\mathbf{r}} & {}^{G}\dot{\mathbf{q}} & {}^{G}\dot{\mathbf{v}} & {}^{B}\dot{\delta\mathbf{a}} & {}^{B}\dot{\delta\boldsymbol{\omega}} \end{bmatrix}^T = \begin{bmatrix} {}^{G}\mathbf{v} \\ \frac{1}{2}\Omega[{}^{B}\boldsymbol{\omega} - {}^{B}\delta\boldsymbol{\omega} + {}^{B}\boldsymbol{\nu}_{\boldsymbol{\omega}}]{}^{G}\mathbf{q} \\ R[{}^{G}\mathbf{q}]({}^{B}\mathbf{a} - {}^{B}\delta\mathbf{a} + {}^{B}\boldsymbol{\nu}_{\mathbf{a}}) - [2{}^{G}\boldsymbol{\Omega}\times]{}^{G}\mathbf{v} + {}^{G}\mathbf{g} \\ {}^{B}\boldsymbol{\nu}_{\delta\mathbf{a}} \\ {}^{B}\boldsymbol{\nu}_{\delta\boldsymbol{\omega}} \end{bmatrix} \tag{5}$$

with

$$ \boldsymbol{\nu} = \begin{bmatrix} {}^{B}\boldsymbol{\nu}_{\mathbf{a}} & {}^{B}\boldsymbol{\nu}_{\delta\mathbf{a}} & {}^{B}\boldsymbol{\nu}_{\boldsymbol{\omega}} & {}^{B}\boldsymbol{\nu}_{\delta\boldsymbol{\omega}} \end{bmatrix}^T, \tag{6}$$

where $\boldsymbol{\Omega}$ is the earth rotation rate of Coriolis effect and ${}^{G}\mathbf{g}$ is the acceleration of gravity. The control inputs of EKF, ${}^{B}\mathbf{a}$ and ${}^{B}\boldsymbol{\omega}$, are acceleration and gyro rate measured from IMU, considering a rigid body transformation between stereo camera at the head and IMU on the pelvis. The related functions are defined like this: $R[\mathbf{q}]$ is the rotation matrix of quaternion $\mathbf{q}$, $[\mathbf{y}\times]$ is the cross product matrix of vector $\mathbf{y}$, and

$$\Omega[\boldsymbol{\omega}] = \begin{bmatrix} 0 & -\boldsymbol{\omega}^T \\ \boldsymbol{\omega} & -[\boldsymbol{\omega}\times] \end{bmatrix}, \tag{7}$$

The computation of the derivative of estimate covariance $\mathbf{P}$ is achieved by the Lyapunov equation.

$$\dot{\mathbf{P}} = \mathbf{FP} + \mathbf{PF}^T + \mathbf{GQG}^T \tag{8}$$

where Jacobians of $\dot{\mathbf{x}}$ with respect to the state and the noise are calculated as

$$\mathbf{F} = \frac{\partial \dot{\mathbf{x}}}{\partial \mathbf{x}}, \ \mathbf{G} = \frac{\partial \dot{\mathbf{x}}}{\partial \boldsymbol{\nu}}, \tag{9}$$

respectively and corresponding covariance of noise uncertainty $\boldsymbol{\nu}$ is obtained as

$$\mathbf{Q} = E[\boldsymbol{\nu}\boldsymbol{\nu}^T]. \tag{10}$$

The derivatives of (5) and (8) can be used in prediction propagation of the filter system where the predicted state $\mathbf{x}$ and its covariance $\mathbf{P}$ can be calculated via numerical integration with sampling time Δt such as the Runge-Kutta method.

2) IMU update: The measurement update process is possible to compensate orientation errors of the estimate state. And the result could be reflected to the other state elements via the estimate covariance as well. It is performed with the difference between actual measurement ${}^B\mathbf{a}$ of acceleration of IMU and predicted measurement $\mathbf{v}_R$ of gravity vector in the body coordinate which is computed by the estimated quaternion.

$$\dot{\mathbf{x}} = -\mathbf{PH}^T\mathbf{R}^{-1}({}^B\mathbf{a} - {}^B\delta\mathbf{a} - \mathbf{v}_R) \tag{11}$$

$$\dot{\mathbf{P}} = -\mathbf{PH}^T\mathbf{R}^{-1}\mathbf{HP} \tag{12}$$

where

$$\mathbf{v}_R = R[{}^G\bar{\mathbf{q}}]{}^G\mathbf{g}, \tag{13}$$

$\bar{\mathbf{q}}$ is the conjugate of a quaternion, and $\mathbf{R}$ is measurement covariance of acceleration. In (11) and (12), the measurement sensitivity matrix $\mathbf{H}$ is denoted as follows:

$$\mathbf{H} = \begin{bmatrix} \mathbf{0}_{3\times 3} & \mathbf{H}_R & \mathbf{0}_{3\times 9} \end{bmatrix} \tag{14}$$

with

$$\mathbf{H}_R = -2\frac{1}{||{}^G\mathbf{q}||^2}[\mathbf{v}_R\times]\Xi[{}^G\mathbf{q}]^T \tag{15}$$

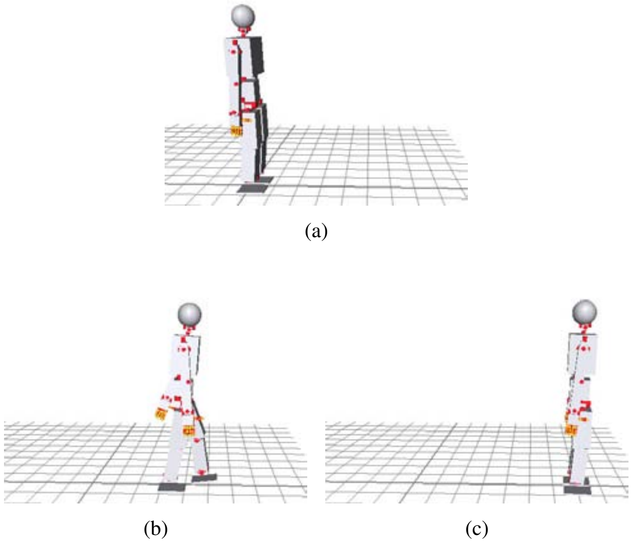
where

$$\Xi[\mathbf{q}] = \begin{bmatrix} -\mathbf{q}_r^T \\ q_0\mathbf{I}_{3\times 3} + [\mathbf{q}_r\times] \end{bmatrix} \tag{16}$$

for

$$\mathbf{q} = \begin{bmatrix} q_0 & \mathbf{q}_r^T \end{bmatrix}^T. \tag{17}$$

The derivatives of (11) and (12) can be augmented to the state propagation equations of (5) and (8) respectively; then, as a result, the updated state $\mathbf{x}$ and its covariance $\mathbf{P}$ can be obtained at once.


(a)

(b)

(c)

Fig. 5. Resulting motion estimation of the proposed method fusing with kinematic odometry, visual odometry, and raw IMU data [(a)→(b)→(c)]. It uses the same sensor data of Fig. 4 and it can offer more accurate odometry information to the SLAM module.

3) Update with relative odometry data: Kalman update with relative pose is adopted to fuse kinematic and visual odometry data into the IMU filter framework, which is similar to the stochastic cloning Kalman filter [18] with no landmark. There are two main reasons to organize the update structure as above. 1) Using relative data of kinematic and visual odometry are more reliable and less sensitive to errors than using accumulated data. 2) It has an advantage in asynchronous update with visual odometry for $30Hz$ and kinematic odometry for $100Hz$.

Initially, the augmented state vector $\mathbf{x}_{aug}$ and its covariance $\mathbf{P}_{aug}$ is constructed as the form of followings, which simply have a copy of the current camera pose as $({}^G\mathbf{r}_{vo}, {}^G\mathbf{q}_{vo})$ for visual odometry and $({}^G\mathbf{r}_{odo}, {}^G\mathbf{q}_{odo})$ for kinematic odometry.

$$\mathbf{x}_{aug} = \begin{bmatrix} {}^G\mathbf{x} & {}^G\mathbf{r}_{vo} & {}^G\mathbf{q}_{vo} & {}^G\mathbf{r}_{odo} & {}^G\mathbf{q}_{odo} \end{bmatrix}^T \tag{18}$$

$$\mathbf{P}_{aug} = \begin{bmatrix} \mathbf{P}_{x,x} & \mathbf{P}_{x,vo} & \mathbf{P}_{x,odo} \\ \mathbf{P}_{x,vo}^T & \mathbf{P}_{vo,vo} & \mathbf{P}_{vo,odo} \\ \mathbf{P}_{x,odo}^T & \mathbf{P}_{vo,odo}^T & \mathbf{P}_{odo,odo} \end{bmatrix} \tag{19}$$

The augmented state vector can be propagated via the IMU prediction and update identically, but the augmented elements are kept constant because they should be used to estimate relative measurement with respect to the time of augmentation. When the relative odometry information from visual or kinematic odometry is available at time k, we can update the augmented state vector

$$\mathbf{x}_{aug,k} = \begin{bmatrix} {}^G\mathbf{x}_k & {}^G\mathbf{r}_{k-n_y} & {}^G\mathbf{q}_{k-n_y} \end{bmatrix}^T, \tag{20}$$

which includes the augmented camera pose at previous time $(k - n_y)$ for $y \in \{vo, odo\}$. Here, the measurement model for Kalman update represents the relative pose change of the camera from time $(k-n_y)$ to k.

$$h_y(\mathbf{x}_{aug,k}) = \begin{bmatrix}\Delta\mathbf{r}_y \\ \Delta\mathbf{v}_y\end{bmatrix} = \begin{bmatrix}R[{}^G\bar{\mathbf{q}}_{k-n_y}]({}^G\mathbf{r}_k - {}^G\mathbf{r}_{k-n_y}) \\ \rho[{}^G\bar{\mathbf{q}}_{k-n_y} \times {}^G\mathbf{q}_k]\end{bmatrix} \tag{21}$$

where $\rho[\mathbf{q}]$ is Rodrigues' rotation formula for quaternion $\mathbf{q}$. After Kalman update, the augmented state and its correlated covariance should be eliminated and re-augmented with the newly updated camera pose at time k for next update. And the update process operates repeatedly whenever information is received from visual or kinematic odometry.

The proposed on-board odometry fusion is a kind of relative localization methods to estimate ego-motion only. Therefore, it has remaining problems on accumulating odometry error as the robot walks. However, reduced error of the proposed odometry as shown in Fig. 5 makes it possible to increase the performance of SLAM remarkably by providing more precise motion model.

IV. 3D Vision-based SLAM

A. Motion Model of Vision-based SLAM

As a motion model of vision-based SLAM, a constant velocity model [11], [12] is most often used to provide better tracking performance during normal operation of smooth camera motion. For dynamic walking control, an interacting method with different motion models [19] would be more appropriate to adapt to various motion changes of the humanoid robot.

In this paper, the motion model is sucessfully superseded by the proposed on-board odometry as follows:

$$\mathbf{x}(k|k-1) = \begin{bmatrix}\mathbf{r}^G(k|k-1) \\ \mathbf{q}^G(k|k-1)\end{bmatrix} = \begin{bmatrix}\mathbf{r}^G(k-1|k-1) + \Delta\mathbf{r}^G(k) \\ \mathbf{q}^G(k-1|k-1) \times \Delta\mathbf{q}^B(k)\end{bmatrix} \tag{22}$$

where $\Delta\mathbf{r}^G(k)$ and $\Delta\mathbf{q}^B(k)$ are relative pose data from the proposed odometry filter. Because the motion model only requires relative pose difference and its covariance as an input, the odometry filter module in section III can be organized separately from the SLAM module.

When using acceleration and gyro rate of IMU, filter divergence is often brought on by inaccurate sensor data which are inconsistent with their uncertainty modeling. To filter out the erratic odometry information, in the proposed odometry filter, validation gating with the Mahalanobis distance is applied.

$$\mathbf{x}^T\mathbf{P}^{-1}\mathbf{x} < \chi^2_{dim(\mathbf{x}),0.95} \tag{23}$$

where the gate threshold is obtained from the inverse chi square cumulative distribution table at a dimension of $\mathbf{x}$ and a significance level 0.95. In case of failing to meet the condition of (23) during odometry estimation, the odometry module simply gets to be reset, and then it can provide the relative data again to the SLAM module. The separate architecture could decrease negative influence of odometry errors on the SLAM estimation directly. While the filter is being reset, the odometry input is unavailable. Meanwhile, a constant position model combined with re-localization could help to reduce filter divergence and maintain tracking consistency of the SLAM module.

B. Vision-based SLAM System

As the basic framework of the SLAM module, the EKF SLAM system with visual features [20] is used. Requirement on relatively small computational burden and usability of stereo camera are considered. The visual feature composed of FAST detector and histogram of gradient descriptor is extracted as visual landmark. The landmark is initialized in the SLAM filter as follows; 1) the feature located within $5m$ is registered as a 3D point landmark and 2) when stereo depth value is more distant or unavailable, the feature is encoded with inverse depth parameterization.

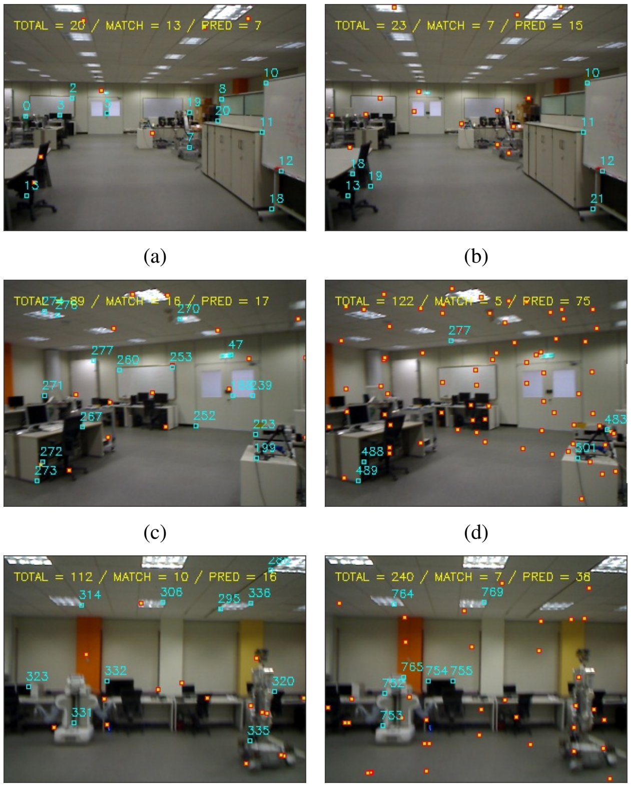
Re-localization in the constructed feature map is important to deal with being lost in feature tracking which frequently happens to dynamic walking robot. In order to speed up the recovery process, the indexing method with Haar coefficients of a local image patch [21] is used. Its quantization table contains the feature candidates which correspond to the specific Haar index. Accordingly, it does not have to search all the features in the map exhaustively, and it can shorten recovery time. In our implementation, the robot can be re-localized within $0.3s$, before the next image is received.

V. Experiments

Roboray is equipped with a stereo camera of VidereDesign at its head, which has the field of view of $60°$. The embedded stereo chip on the camera makes it possible to achieve $30Hz$ stereo processing. In addition, the TOF camera is placed at breast height of Roboray to build a 3D voxel map around the given environment, which is used to evaluate the estimation performance in section V-B.

A. Result: SLAM for Humanoid Robot, Roboray

In order to verify the advantage of the proposed on-board odometry for feature tracking, we compared the proposed SLAM with the vision-only SLAM [20]. The estimated odometry successfully predicts the position of previously registered landmarks at the current image frame. As shown in Fig. 6, the ratio of the matched features is increased even with rapid changes in consecutive images and resulting motion blur. Consequently, the robot pose can be consistently estimated by stable tracking of the landmarks. On the other hand, the estimation of the vision-only SLAM became to be diverged for dynamic walking Roboray with heel-toe motion. Due to the insufficient performance of feature tracking, even loop closing process was impossible at the starting point where Roboray revisited. Figure 7 also shows the performance of the proposed odometry by comparing the total number of registered visual landmarks. The vision-only SLAM frequently misses to track the registered landmarks, needs to add new ones, and then gets to delete the missed ones. They make the peaks in Fig. 7 and increase the number


TOTAL = 20 / MATCH = 13 / PRED = 7
TOTAL = 23 / MATCH = 7 / PRED = 15
TOTAL = 89 / MATCH = 16 / PRED = 17
TOTAL = 122 / MATCH = 5 / PRED = 75
TOTAL = 112 / MATCH = 10 / PRED = 16
TOTAL = 240 / MATCH = 7 / PRED = 38

(a) (b)

(c) (d)

(e) (f)

Fig. 6. Sample image frames from SLAM processes (left column: the proposed SLAM, right column: the vision-only SLAM [20]), which have matched features (cyan with number) and unmatched features (yellow without number). The results are obtained by (a, b) moving forward, (c, d) counterclockwise rotation, and (e, f) moving forward with motion blur.

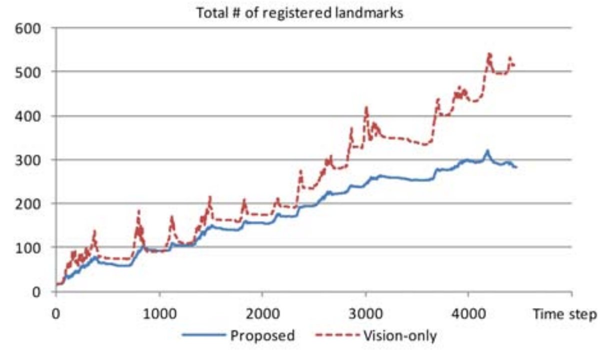
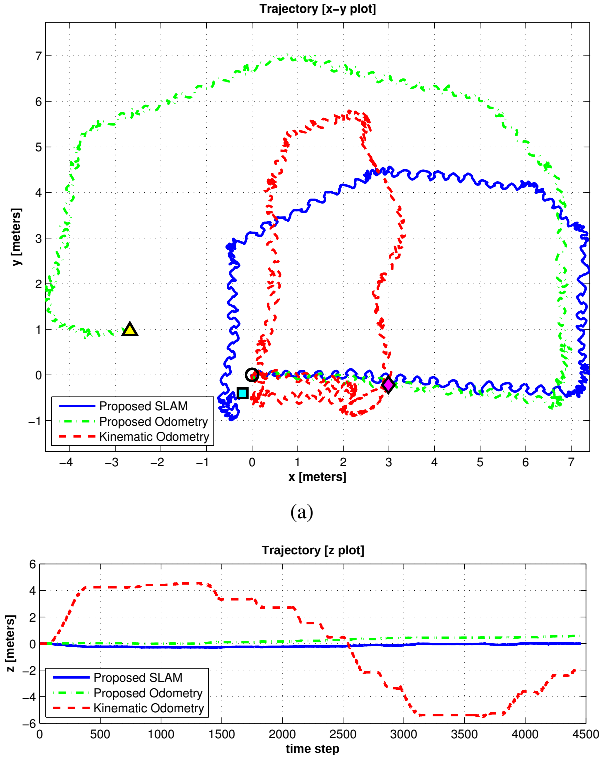


Fig. 7. Total number of visual landmarks registered in the map of the proposed SLAM and the vision-only SLAM [20]. The proposed odometry allows Roboray to estimate its ego-motion with smaller number of landmarks.

of landmarks unnecessarily. The proposed method can re-match more features, and so it needs fewer new features than the vision-only method, thanks to more accurate motion model of the proposed odometry.

We made a comparison between the estimated trajectories of the kinematic odometry, the proposed odometry, and the proposed SLAM (Fig. 8). 1) The kinematic odometry suffers from the problem that the stance foot cannot be fully contacted on the ground. As a result, it has large error in z direction as if it were flying or sinking. 2) In case of the proposed odometry, the error is more reduced with the help of IMU and visual odometry. However, it still have unbounded drift error in rotation and error accumulation in translation. 3) The SLAM method coupled with the odometry remarkably improves the estimation result. Its trajectory can describe the trace of s-curve walking pattern of Roboray in a transverse plane well. At Table I, we can check the final pose of the proposed SLAM (square mark in Fig. 8), compared with that of a ground truth. As the ground truth, the optimized method such as non-linear bundle adjustment between two images which have their own 3D metric data is used (Fig. 9). It shows that the proposed SLAM has small error enough to close the loop around its revisiting place.



(a)

(b)

Fig. 8. Estimated trajectories of the kinematic odometry (red/dash), the proposed odometry (green/dot-and-dash), and the proposed SLAM (blue/solid). The estimated end points are presented by diamond, triangle, and square mark, respectively.

B. Result: 3D Voxel Map Building

The result of the proposed motion estimation can be used to reconstruct the spatial structure of the environment by

TABLE I

COMPARISON OF THE ESTIMATED POSE AT THE END POINT

	position $[m]$			orientation $[°]$		
	x	y	z	roll	pitch	yaw
proposed	-0.208	-0.401	-0.001	-0.944	0.484	8.996
optimized	-0.223	-0.345	0.001	-0.048	0.503	8.689

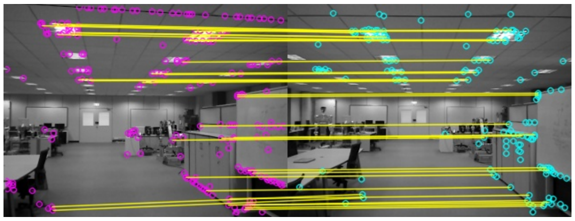


Fig. 9. The result of feature matching between two images captured at the starting point (left) and the end point (right) for non-linear optimization

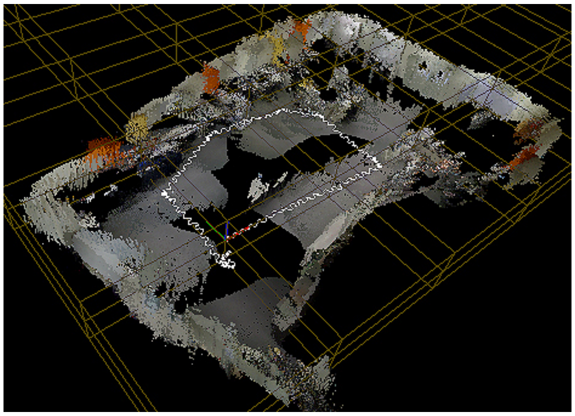


Fig. 10. Bird-eye view of resulting 3D voxel map of the given environment, which includes the estimated path (white)

organizing a 3D voxel map. In other words, we can verify the performance of the estimation result from the shape of the 3D map which is greatly influenced by the accuracy of the motion estimates.

Based on the estimated trajectory, a 3D voxel map of $20m \times 16m$ environment was built with RGB color data of stereo camera and depth data of TOF camera (Fig. 10). Detailed explanation on the map construction will not be forthcoming, since it is not the scope of this paper. The map successfully reflects the appearance of the environment, and we can evaluate how well the proposed method is working.

VI. Conclusion

In this paper, we develop a vision-based 3D motion estimation method utilizing on-board odometry information. The enhanced odometry estimation is embodied with kinematic odometry, visual odometry and raw IMU data. It improves SLAM estimation in the situation of the dynamic walking with heel-toe motion as well as the conventional ZMP-based walking.

Roboray was tested in a small indoor environment with the simplest version of SLAM estimation; however, our architecture also has a capability of sub-mapping SLAM estimation, which makes Roboray to step forward in larger environment. In future work, after improving walking ability for uneven terrain, we plan to test Roboray to localize in a building-sized space. And, for another future work, we will empower Roboray with an integrated solution of autonomous navigation, which is organized with localization, mapping and path planning.

Acknowledgment

The authors would like to thank the university of Bristol, Andrew Gee, Pished Bunnun, Andrew Calway, and Walterio Mayol-Cuevas for implementing SLAM system via a collaborative research project.

References

[1] T. Takenaka, T. Matsumoto, and T. Yoshiike, “Real Time Motion Generation and Control for Biped Robot - 1st Report: Walking Gait Pattern Generation,” *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 1084–1091, 2009.

[2] K. Kaneko, F. Kanehiro, M. Morisawa, K. Miura, S. Nakaoka, and S. Kajita, “Cybernetic Human HRP-4C,” *Proc. of 9th IEEE-RAS Int. Conf. on Humanoid Robots*, pp. 7–14, 2009.

[3] Boston Dynamics, “Petman”, `http://www.bostondynamics.com/robot_petman.html`, 2010.

[4] Y. Ogura, K. Shimomura, H. Kondo, A. Morishima, T. Okubo, S. Momoki, H. Lim, and A. Takanishi, “Human-like Walking with Knee Stretched, Heel-contact and Toe-off Motion by a Humanoid Robot, ” *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 2497–2502, 2006.

[5] B. Lim, M. Lee, J. Kim, J. Lee, J. Park, K. Seo, and K. Roh, “Control Design to Achieve Dynamic Walking on a Bipedal Robot with Compliance,” *Proc. of IEEE Int. Conf. on Robotics and Automation*, accepted, 2012.

[6] B. Siciliano and K. Oussama, eds., *Handbook of Robotics*, Springer Verlag, Heidelberg, 2008.

[7] R. O. Castle and D.W. Murray, “Keyframe-based Recognition and Localization during Video-rate Parallel Tracking and Mapping,” *Image and Vision Computing*, vol. 29, pp. 524–532, 2011.

[8] S. Kagami, Y. Takaoka, Y. Kida, K. Nishiwaki, and T. Kanade, “Online Dense Local 3D World Reconstruction from Stereo Image Sequences,” *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 3858–3863, 2005.

[9] A. Pretto, E. Menegatti, M. Bennewitz, W. Burgard, and E. Pagello, “A Visual Odometry Framework Robust to Motion Blur,” *Proc. of IEEE Int. Conf. on Robotics and Automation*, pp. 2250–2257, 2009.

[10] S. Oßwald, A. Hornung, and M. Bennewitz, “Learning Reliable and Efficient Navigation with a Humanoid,” *Proc. of IEEE Int. Conf. on Robotics and Automation*, pp. 2375–2380, 2010.

[11] O. Stasse, A. J. Davison, R. Sellaouti, and K. Yokoi, “Real-time 3D SLAM for Humanoid Robot considering Pattern Generator Information,” *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 348–355, 2006.

[12] A. J. Davison, I. D. Reid, N. D. Molton, and O Stasse, “MonoSLAM: Real-Time Single Camera SLAM,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 1052–1067, 2007.

[13] N. Kwak, O. Stasse, T. Foissotte, and K. Yokoi, “3D Grid and Particle based SLAM for a Humanoid Robot,” *Proc. of 9th IEEE-RAS Int. Conf. on Humanoid Robots*, pp. 62–67, 2009.

[14] M. W. Whittle, “Gait Analysis: An Introduction,” Butterworth-Heinemann, 4th edition, 2010.

[15] J. M. Czerniecki, “Foot and Ankle Biomechanics in Walking and Running,” *American Journal of Physical Medicine and Rehabilitation*, pp. 246–252, 1988.

[16] M. Agrawal and K. Konolige, “Rough Terrain Visual Odometry,” *Proc. of Int. Conf. on Advanced Robotics*, 2007.

[17] F. L. Markley, “Attitude Estimation or Quaternion Estimation?,” *Journal of Astronautical Sciences*, vol. 52, no. 1–2, pp. 221–238, 2004.

[18] A. I. Mourikis, S. I. Roumeliotis, and J. W. Burdick, “SC-KF Mobile Robot Localization: A Stochastic Cloning Kalman Filter for Processing Relative-State Measurements,” *IEEE Trans. on Robotics*, vol. 23, no. 4, pp. 717–730, 2007.

[19] J. Civera, A. J. Davision and M. M. Montiel, “Interacting Multiple Model Monocular SLAM,” *Proc. of Int. Conf. on Robotics and Automation*, pp. 3704–3709, 2008.

[20] D. Chekhlov, M. Pupilli, W. Mayol-Cuevas, and A. Calway, “Robust Real-time Visual SLAM Using Scale Prediction and Exemplar Based Feature Description,” *Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pp. 1–7, 2007.

[21] D. Chekhlov, W. Mayol-Cuevas, and A. Calway, “Appearance Based Indexing for Relocalisation in Real-time Visual SLAM,” *Proc. of British Machine Vision Conf.*, pp. 363–372, 2008.