Visual-Inertial Ego-Motion Estimation for Humanoid Platforms

Konstantine Tsotsos
Computer Science Department
UCLA, Los Angeles – CA 90095
Email: ktsotsos@cs.ucla.edu

Alberto Pretto
Department of Information Engineering
University of Padova – 35131 Padova, Italy
Email: alberto.pretto@dei.unipd.it

Stefano Soatto
Computer Science Department
UCLA, Los Angeles – CA 90095
Email: soatto@ucla.edu

***Abstract*—We describe an ego-motion estimation system developed specifically for humanoid robots, integrating visual and inertial sensors. It addresses the challenge of significant scale changes due to forward motion with a finite field of view by using recent sparse multi-scale feature tracking techniques. Additionally, it addresses the challenge of long-range temporal correlation due to walking gaits by employing a kinematic-statistical model that does not require accurate knowledge of the robot dynamics and calibration. Our system achieves performance comparable to the state of the art at a fraction of the (inertial measurement unit) cost, on a challenging dataset that we have created.**

I. Introduction

Ego-motion estimation refers to the inference of the rigid motion of a sensor platform in three dimensions (3-D). We are interested in the specific case where the platform is mounted on a humanoid robot. After several decades of research and development in inertial navigation, one can find suitable inertial measurement units (IMUs) that will achieve the desired performance, at a suitable cost. Unfortunately, the performance point required for stabilization as well as navigation and mapping of a humanoid robot often corresponds to a price point that is prohibitively high. Therefore, it is common to attempt to fuse different sensor data in order to achieve the same performance at a lower cost (or higher performance at equal cost). This process has accelerated lately with the availability of mobile platforms such as phones and tablets that include inertial sensors (accelerometers and gyrometers) as well as global positioning (GPS) and cameras, albeit all of modest quality. In addition, the fusion of multiple sensor data enables the simultaneous reconstruction of a coarse representation of the environment that can be used for other tasks such as obstacle avoidance, recognition, localization (loop closure), etc. Unfortunately, sensor fusion introduces different failure modes that significantly complicate the analysis and performance evaluation of the resulting platforms.

We are interested in the specific case of integration of visual and inertial measurements into one coherent ego-motion state. These sensors are readily available in most humanoid platforms, and can be made to be cheap, small and light enough to fit in compact and energy-efficient platforms. They are both passive (they do not require broadcasting a signal and measuring the return, which is expensive in terms of energy), and exhibit complementary failure modes: The quality of ego-motion estimates from imaging data depends on the field of view (FOV), the spatial distribution of the scene, occlusions, photometric characteristics, illumination, etc. none of which affects inertial measurements. Equivalently, the quality of inertial measurements depends on temperature, gravity, drifts and biases, none of which affect imaging data.

A. Related Work and Contributions

There has been a significant amount of work in the integration of visual and inertial sensors, which we will review shortly. However, all exhibit characteristics that limit their applicability to humanoid platforms.

Typically, motion states are modeled as $k-$th order random walks: The unknown pose is modeled as the integration of the unknown (linear and angular) velocity, which is modeled as the integration of the unknown acceleration, which is the integration of the unknown jerk (derivative of acceleration), etc. Absence of knowledge is propagated up until a sufficiently high order of derivative can be represented as uninformative "white noise". While this process works for sensor platforms that are attached to vehicles that move smoothly on the ground or in the air, it does not work well for humanoids, for high order derivatives exhibit significant correlation structure due to the quasi-periodicity of walking gaits as well as acceleration spikes due to contact forces.

Furthermore, humanoids are typically anthropomorphic and are equipped with forward facing monocular or stereo vision systems. Cost, weight and size constraints usually condition the choice of sensors that have to serve both navigation tasks (that favor the widest FOV) and focused tasks such as recognition, manipulation etc. (that favor a small FOV) aided by active gaze control. With one or more relatively small FOV camera, forward motion typically causes significant changes in *scale*, which is a challenge for visual tracking. Also, forward motion is the most prone to local minima [1]. On the other hand, humanoids often tend to a task while walking, which can cause extended periods of sideways motion (direction of heading not aligned with the optical axis), without visuals on impending obstacles.

Research supported by ARO 56765-CI and AFOSR FA9550-12-1-0364

While in principle the walking cycle is known from actuator commands, and the kinematic and dynamic characteristics are known, propagating them to forces at sensor nodes requires accurate calibration that is impractical, and also subject to change if the robot acquires a load.

In this manuscript, we describe a method for ego-motion estimation that extends existing visual-inertial integration schemes to the specific case of humanoid motions. It does not require accurate kinematic or dynamic modeling of the sensor platform. Instead, it *(a)* exploits recent developments in multi-scale feature tracking [2] to enable longer tracks despite significant scale changes, and *(b)* modifies the statistical motion model to take into account higher order derivatives to enable capturing temporal correlations of the walking gaits. We evaluate our approach in comparison to existing (generic) visual-inertial integration systems including [3], [4], [5], and to a state-of-the-art IMU-aided visual odometry system [6]. Because of the lack of benchmark datasets for this task, we have used published results of competing approaches, that have been gathered from moving ground vehicles. Therefore, the comparison is biased in their favor since such motions are usually closer to the statistical model of a (typically second-order) random walk. Nevertheless, we achieve comparable to state-of-the-art performance on more challenging sequences captured during walking motion (Sect.) V. It is also important to note that we achieve these results with IMUs that are one to two orders of magnitude cheaper than those employed by competing schemes.

Several authors have investigated the problem of inferring a robot's ego-motion using images: for instance one of the best-known visual SLAM systems, presented in 2003 by Davison [7]. He proposed a bearing only solution to the SLAM problem fusing feature tracks inside an Extended Kalman Filter (EKF) framework. In [8], the authors describe an image-based approach for tracking the trajectory of a stereo camera based on a quadrifocal relationship between the image intensities within adjacent views of the stereo pair. In [9] an accurate and fast incremental motion reconstruction algorithm that uses a local bundle adjustment method to improve motion estimation accuracy. Klein and Murray [10] proposed an effective Visual SLAM approach based on edgelets tracking that exploits local bundle adjustment and a key-frame-based re-localization method in order to recover from tracking failures. For the specific case of humanoid motions, a first successful validation of a Visual SLAM was presented by Stasse *et al.* in 2006 [11], using a HRP-2 humanoid as an experimental platform. This system exploits the framework presented in [7], showing loop closure properties for short (a few meters) closed trajectories. In [12], the authors addressed the problem of motion blur in images acquired by humanoid robots by proposing an epipolar geometry-based visual odometry system that takes advantage of a feature detection and tracking scheme that explicitly models the presence of motion blur inside the images.

Among competing visual-inertial schemes, we include an extension of the SFM system [13] to include two stereo cameras (pointing in opposite directions) and a cheap IMU [5]. Jones et al. in [14] and Jones and Soatto in [3] address the problem of automatically calibrating the camera-IMU system and demonstrate results of an integrated loop-closure approach on large loops, the same problem discussed by [15]. Incremental bundle adjustment alongside visual-inertial navigation has been exploited in [6], while a novel measurement model to deal with visual and inertial data inside an EKF framework is presented in [4].
An exhaustive survey on the best-known vision-based ego-motion estimation techniques can be found in [16].

In the next section we address the aforementioned point *(a)* by describing a tracking approach that enables dealing with significant scale changes. In the following section we address point *(b)* by showing a simple kinematic-statistical motion model that enables dealing with long-range temporal correlation of the derivative chain typical of walking gaits, without explicitly modeling the robots' dynamics.

II. Tracking under severe scale changes

Forward motion (motion along the optical axis) causes a (non-isotropic) scale change in the domain of the image. It depends on the shape of the scene, and can cause significant deformations, including (self) occlusions. Even if one restricts attention to *co-visible* portions of the scene (away from occlusions), scale can cause non-smooth changes of the value of the image. Here, co-visible portions refer to the regions of the image that remain visible through a viewpoint change. While a rescaling of the image domain is a group in the continuous limit of infinite resolution [17], because of spatial quantization there are catastrophic phase changes [18] in the response of feature detection functionals [19]. Because sparse feature trackers such as [20] rely on the persistence of the response of feature detection functionals, they fail under the genetic effects (births and deaths of extrema) arising from significant scale changes. Multi-scale versions of [20], for instance [21] implemented in OpenCV [22] fail to address this issue because the multi-scale selection process is not consistent with the topology of the response of co-variant detector functionals.

This has been recently addressed by [2] in an approach called "tracking on the selection tree" (TST) whereby tracking is performed by running a co-variant detection functional[1] at multiple scales on one image, and then testing for topological consistency in temporally adjacent images. This means that, *at a given scale, an isolated extremum in one image has one and only one corresponding extremum in the next image at the same or adjacent scale.*

We refer the reader to [2] for details on the construction of these trees. Here we just remark that the implementation is a modification of the standard multi-scale version of Lucas and Kanade's tracker (MLK) in OpenCV. Tracking

[1] Although [2] use [20] as a co-variant detection mechanism, any other local feature detection can be used instead.

is performed at multiple scales until a fine enough scale is found where topologically consistent correspondence cannot be established and therefore the track is terminated and the (similarity) motion at the current scale is reported. Motion at coarser scales is only used to register subsequent scales so as to bring extrema into local correspondence. Violation of topological consistency can be due to violation of any of the three assumptions underlying TST: (i) co-visibility (occlusion), (ii) Lambertian reflection, (iii) constant illumination.

This approach allows the tracker to maintain higher quality tracks throughout the image by automatically rejecting features violating the constraint. Therefore, TST is more robust to tracks that degrade due to scale change, foreshortening of a planar surface, occlusions, or low texture than the standard MLK tracker. This difference is illustrated in the example in Fig. 1 that compares the results of TST and MLK. The first 2 frames ((a) to (d)) are taken from a video where the camera is moved (by hand) around a typical indoor office environment that a humanoid robot might encounter (including high-contrast calibration grids ubiquitous in Computer Vision laboratories). The room contains many low texture surfaces and an office divider that acts as an occluder. The checkerboards provide easily trackable features that can be used for ground truth evaluation. The video includes significant occlusion and foreshortening effects that expose the limitations of the MLK tracker and the strengths of TST. Motion estimation results on this dataset are also shown in section V (Handheld motion 2). The final frame ((e) and (f)) is taken from a video exhibiting severe scale change. The camera begins sitting on surface of the table in the center of the image, and is moved directly backwards by approximately 3m causing a significant reduction in scale of the scene on the tabletop.

Both trackers are based on the same image features. In both cases the trackers are limited to 200 features at all times, far more than are typically employed for ego-motion estimation. More are used here in order to better illustrate the differences between trackers. For both trackers, when tracks are lost, new features are acquired up to the limit of 200. The key insight that can be garnered from these examples is the following: in all three cases of challenging situations, the majority of tracks propagated by MLK become incorrect (occlusions) or uninformative (foreshortening, scaling) and these cannot be distinguished from good tracks through visual information alone. Conversely, the constraints imposed by TST allow the tracker to reject such tracks before they degrade, without using additional information (such as motion estimates for re-projection error) and allowing new and more useful features (up to a maximum number) to be acquired.

Figure 1(a) shows tracks from the MLK tracker clustered along an occlusion boundary as it moves rightward across the image. In (b), we can see that TST has automatically rejected these features because they violate its topological constraints, allowing new and more useful features to be acquired. Figure 1 (c) shows that as the checkerboards become more foreshortened, the feature tracks from the MLK tracker tend to converge into a cluster. In (d) we can see that TST has rejected these features before they group together, allowing it to acquire better features spread throughout the scene. Finally, in (e) the scale change has clearly caused the tracks from MLK to converge towards a point, whereas in (f) they have been automatically rejected and new features sampled from the scene by TST. MLK's poor performance near the center of contraction in the image (as observed in (e)) is ubiquitous when undergoing scale changes in the small, indoor environments we are targeting. This problem is avoided through our use of TST and its robustness to scale change.

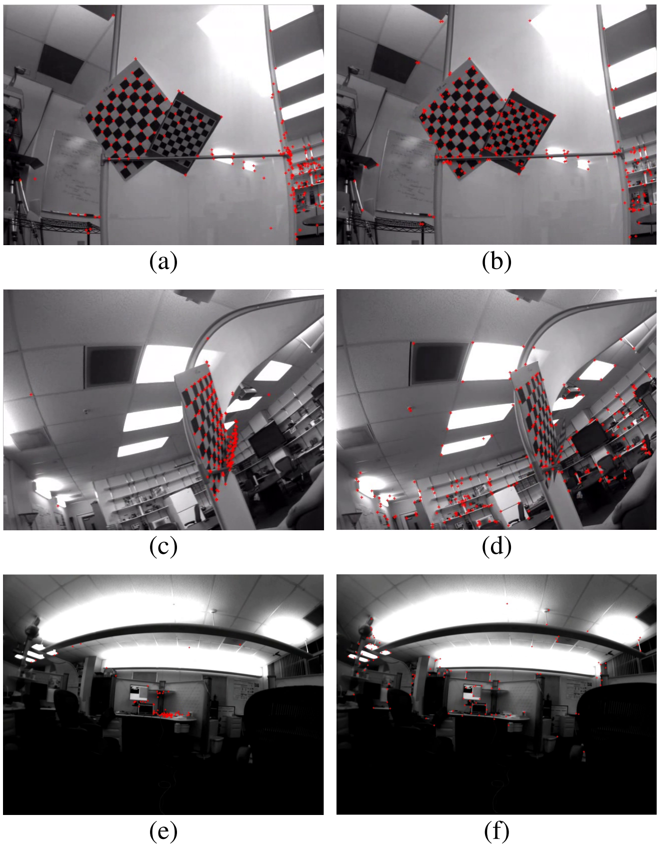
All of these poor tracks are eliminated immediately by the topological constraints of TST, preventing them from negatively impacting estimates of the robot's ego-motion and allowing new features to be tracked in cases of occlusions, foreshortening, scaling, and poor texture.

It is important to note that during normal operation of state estimation, some, but not all, of these poor tracks can be detected and eliminated from the filter by monitoring their measurement innovation. While this can be effective, it does not prevent the tracks from negatively impacting the state estimate before their re-projection error has passed a threshold. The use of TST eliminates this problem as the tracks are rejected at the level of the tracker before they can have a significant impact on the performance of the filter. This is supported by the uniform improvement in state estimation performance, discussed in section V, when using TST compared to the MLK tracker.

III. Motion Model

Our state estimation system employs an Extended Kalman Filtering approach, where IMU readings, as well as feature tracks, are used as explicit measurements inside the EKF update step, while the prediction step is triggered in between measurements. We therefore have adopted a tight integration model, whereby all measurements (inertial and visual) contribute to a common underlying state [23]. This is different from a loose integration model, whereby inertial and visual measurements independently produce an ego-motion estimate, and they are fused at the level of rigid motions [24]. Also, among tight integration models, we choose to represent all measurements as such – that is as output of the dynamical system that generates the data. This is in contrast to modeling motion measurements as ("noiseless") *inputs* to the system, and visual measurements as outputs. Inertial errors are then represented as modeling errors.

The state of the dynamical model that we employ is then the common underlying motion, represented as (discrete-time) random walk with four levels of differentiation of the translation states. Therefore, the derivative of translational acceleration, ξ, called "jerk," is explicitly represented as a state, and modeled as a Brownian motion, whose input is assumed to be uninformative ("white") noise. The complete motion model in discrete time is shown in equation (1). Here the subscript t refers to the time index, and dt is the length



(a) (b) (c) (d) (e) (f)

Fig. 1. Comparison of MLK ((a) and (c)) and TST ((b) and (d)) on two segments of a video featuring challenging occlusions and foreshortening effects. The camera motion in both cases is to the left, moving horizontally past the office divider in the top row and circling it in the bottom row. Note the features (incorrectly) clustered along the occlusion boundary and converging to a single point in frames (a) and (c) respectively for MLK. In frame (e) we see the tracks from MLK have clustered into the center of the image due to significant scale change and become uninformative. These poor tracks have been rejected by TST in frames (b), (d), and (f).

of time between discrete steps. The full set of model states for motion is as follows (all states are elements of $\mathbb{R}^3$): T for translation, Ω for orientation, V for translational velocity, ω for rotational velocity, a for translational acceleration, w for rotational acceleration, and ξ for translational jerk. Finally, as mentioned above, n_{ξ_t} and n_{w_t} are white noise processes.

In equation (1), $R(\Omega_t) \doteq \exp(\widehat{\Omega}_t)$ is the rotation matrix corresponding to the rotation vector Ω_t, where $\widehat{\Omega}_t$ is the skew-symmetric matrix corresponding to Ω_t, and $Log_{SO(3)}(R(\Omega_t)) \doteq \Omega_t$ is the rotation vector Ω_t corresponding to the rotation matrix $R(\Omega_t)$ [25].

$$\begin{cases} T_{t+dt} = T_t + V_t dt + \frac{1}{2} a_t dt^2 + \frac{1}{6} \xi_t dt^3 \\ \Omega_{t+dt} = Log_{SO(3)}(R(\Omega_t) R(\omega_t dt) R(\frac{1}{2} w_t dt^2)) \\ V_{t+dt} = V_t + a_t dt + \frac{1}{2} \xi_t dt^2 \\ \omega_{t+dt} = \omega_t + w_t dt \\ a_{t+dt} = a_t + \xi_t dt \\ w_{t+dt} = w_t + n_{w_t} \\ \xi_{t+dt} = \xi_t + n_{\xi_t} \end{cases} \tag{1}$$

Compared with a standard second-order random walk, this model presents several advantages. First, translational jerk enables capturing the sharp accelerations due to contact forces caused by footsteps that would otherwise end up in the acceleration residual and contain significant temporal correlation that is detrimental to any on-line filtering scheme. The customary approach to handing such correlations is to augment the model with additional states, which is precisely the purpose of adding the jerk state ξ_t. Note that we do not insert rotational jerk as a state since contact events do not usually produce impulsive rotations. Second, we use a third-order approximation of positional kinematics, and a second-order approximation for rotational ones. These we refer to as ‘full kinematics’ throughout the paper, as opposed to more common minimal first-order approximations for kinematics (referred to as ‘minimal kinematics’). Given the fact that inertial measurement rates are usually higher than image frame-rate, time synchronization between the samples as well as proper integration beyond first-order Euler are critical to the functioning of the filter.

Note that there are no “noise terms” on the first five blocks of states, since the model is just a deterministic integrator and there are no modeling errors. This would not be the case if ω_t and α_t were replaced by their measured counterparts from inertial sensors. In that case, the inevitable measurement noise would have to be encoded as modeling error.

In addition to the motion states, there are a number of unknown parameters as well as sensor biases that have to be identified. As customary, we include them in the state of the model with a simple first-order random walk dynamic, with covariance of the model error being a tuning parameter to be determined using customary procedures such as the cumulative periodogram [26]. We discuss the measurement model in section IV-A.

IV. Implementation Details

A. Sensor Measurements

In addition to motion states, we must infer accelerometer biases, represented by a vector $\alpha_{bias} \in \mathbb{R}^3$, gyrometer biases, $\omega_{bias} \in \mathbb{R}^3$, and gravity $\gamma \in \mathbb{R}^3$ in order to properly formulate the measurement equations. We will assume that the alignment between the camera reference frame and the body frame of the IMU is a known transformation $g_{cb} = (R_{cb}, T_{cb}) \in SE(3)$, determined as part of a calibration procedure [23].

While gravity could be determined based on location, we want to avoid lengthy initializations, which are necessary due to the uncertainty in the orientation of the sensor platform relative to an earth-centered reference frame. Therefore, we enforce the norm of gravity as a pseudo-measurement constraint by assigning the norm of γ to that inferred from tabulates, but otherwise include it as a state with trivial dynamics $\gamma_{t+dt} = \gamma_t$. Since gravity is constant, it is initialized with a large error covariance and therefore allowed to float during the initial transient, after which it remains essentially constant. Biases are modeled as random walks,

$\alpha_{bias}(t+dt) = \alpha_{bias}(t) + n_{\alpha_{bias}}(t)$ with $n_{\alpha_{bias}}$ a realization from a zero-mean Gaussian noise process whose covariance is inferred from characteristics of the sensor. Similar assumptions are made for the gyro bias. Note: indexing of time (t) for accelerometer and gyrometer terms differs from the subscripts used for other states in the model (shown in equation (1)) for the sake of clarity.

We do not model the relative pose (position and orientation) of the accelerometer relative to the gyro, the deviation from orthogonality of their axes, their scaling factors, and other calibration parameters that we assume have been compensated by the IMU in order to provide a measurement of translational acceleration and rotational velocity. The latter is related to the motion states above in a straightforward manner:

$$y_{gyro}(t) = \omega_t + \omega_{bias}(t) + n_{gyro}(t) \tag{2}$$

The former requires two transformations to the inertial frame and the subtraction of gravity:

$$y_{accel}(t) = R(\Omega_t)^T(\alpha_t - \gamma_t) + \alpha_{bias}(t) + n_{\alpha_{accel}}(t) \tag{3}$$

The camera measurements are the positions of feature points $y_t^i \in \mathbb{R}^2$ in a calibrated camera, that are related to the position of points $X_0^i \in \mathbb{R}^3$ relative to the initial time instant via a canonical central projection $y_0 = \pi(X_0)$ with $\pi([X_0, Y_0, Z_0]^T) = [X_0/Z_0, Y_0/Z_0]^T$. We then represent X_0 using y_0 and Z_0 via $X_0 = \bar{y}_0 Z_0$ where $\bar{y}_0 = [y_0^T, 1]^T$ is the homogeneous coordinate of y_0. To enforce the positivity of Z_0, we represent it in exponential coordinates via $Z_0 = e^\rho$ with $\rho \in \mathbb{R}$.

In the moving frame, we then have

$$y_t^i = \pi(R_{cb}(R(\Omega_t)^T(R_{cb}^T(\bar{y}_{0_t}^i e^{\rho_t^i} - T_{cb}) - T_t)) + T_{cb}) + n_t^i \tag{4}$$

where, as usual, n_t^i is a measurement error due to the imaging sensor as well as errors of the feature tracker. This noise is not Gaussian, and typically has heavy tails and possibly multi-modality due to mismatches in self-similar regions. This will be handled by a robust update where measurements that deviate from the mean of the conditional distribution are weighted linearly rather than quadratically [27].

In addition to such mismatches, features appear and disappear due to occlusions and to scale changes as we have discussed in previous sections. Thus the filter includes means to represent groups of features that are simultaneously acquired, and book-keeping features that have disappeared by storing their 3-D location in a map [23].

B. Feature Initialization

In [23], features were initialized through a bootstrapping phase where a separate sub-filter was spawned with groups of features and used the estimated motion to triangulate features until their depth error covariance was below a threshold. Unfortunately, active gaze control in humanoid robots often causes sudden rotational motions that produce motion blur and therefore cause complete loss of track. It is therefore of paramount importance to be able to add features to the state with the shortest possible delay, unlike [23].

We have adopted a considerably simplified approach, whereby the (log) depth of new features are initialized to a uniform prior value and with high uncertainty. Our results have shown that such a prior is effective in the indoor environments we have targeted and that depths very quickly converge to steady-state. Note that this is possible because of the presence of inertial measurements. Direct insertion in the state in the absence of inertial measurements causes the filter to diverge.

V. Experimental Results

To test our method we used a device composed of a low-cost IMU (XSense MTI), providing data at a frequency of 100Hz, rigidly mounted on a black and white camera acquiring 1024×768 images at 30Hz. The entire ego-motion estimation system runs in realtime on a commodity laptop and its stability and robustness have been tested extensively. It is important to note that the IMU we are using is approximately one twentieth the cost of the one used by Jones and Soatto [3] (C-MIGITS), and has a comparable reduction in stability of biases, and increase in measurement noise.

Three main experimental datasets were collected, two by hand and one with the camera-IMU system mounted on a hardhat and worn by an operator to simulate challenging walking motion of a humanoid robot for the system. The two hand-held datasets include significant rotations and general motion. The third was taken to demonstrate the applicability of our system to situations directly relevant to the walking motion of a humanoid robot. In order to simulate this, the camera-IMU system was rigidly attached to the top of a hardhat worn by an operator (Fig. 2) who then recorded a dataset while briskly walking through an indoor environment. Necessarily, the walking motion is largely constrained to a plane and is not as general as in the hand-held datasets. However, no assumptions on the type of motion are used and parameters are identical in all experiments. As no motion capture system was available to record ground-truth, all of our datasets represent closed-loop trajectories (where the start and end points are the same) and we evaluate our results based on drift from the origin at the end of the run. In all three datasets the platform was carefully returned to its approximate original position and orientation. Challenges of this walking motion are discussed in section V-A and results on loop-closure experiments are presented in section V-B.

A. Walking vs. hand-held motion

The majority of visual-inertial ego-motion estimation systems are typically demonstrated using hand-held motions or are mounted on wheeled platforms. When dealing with data from IMUs, particularly low-end IMUs such as we use, this avoids some of the challenging issues that must be faced by a humanoid robot using legged motion. In this case, the impact forces from footsteps during even casual walking lead to significant accelerations and noise measured by the

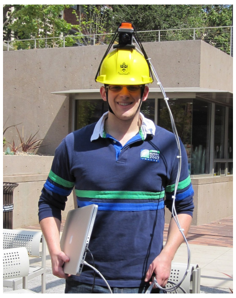
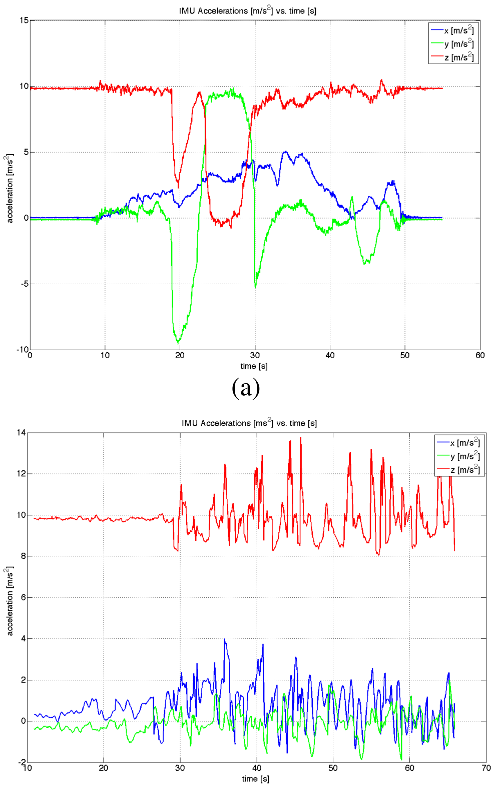


Fig. 2. Walking motion experiment setup.

accelerometers that the system must be robust to. Examples comparing accelerometer data from hand-held, general motion, and head-mounted walking motion are shown in Fig. 3. As can be seen in the comparison, the accelerometer data received during the walking motion exhibits very high amplitude "noise" caused by the impact forces of footfalls. While these accelerations appear to be noise, they are in fact "signal" in the sense of relating to the operator's gait, and these high frequency, large changes in acceleration must be accounted for by the system. In comparison with the smoother hand-held motions, these impact accelerations make motion estimation more challenging due to their magnitude and quasi-periodic nature. As shown in the results of section V-B, our expanded translational and rotational kinematics have improved motion estimation results under such accelerations. Additionally, while there is a qualitative difference in the rotational velocity measurements between the hand-held and walking situations, it is not as significant as the difference seen in the acceleration data. Therefore, we have kept the motion model at one order above the observed data for both acceleration and rotational velocity.



(a)

(b)

Fig. 3. Accelerometer data from hand-held motion (top) and head-mounted walking motion (bottom) over similar time scales, starting from the beginning of motion. Note the high amplitude 'noise' caused by impact forces in the walking case compared to the relatively smooth accelerations experienced in the hand-held case. Note: Large changes in accelerations in the hand-held cases are caused by gravity while the system undergoes rotation.

TABLE I

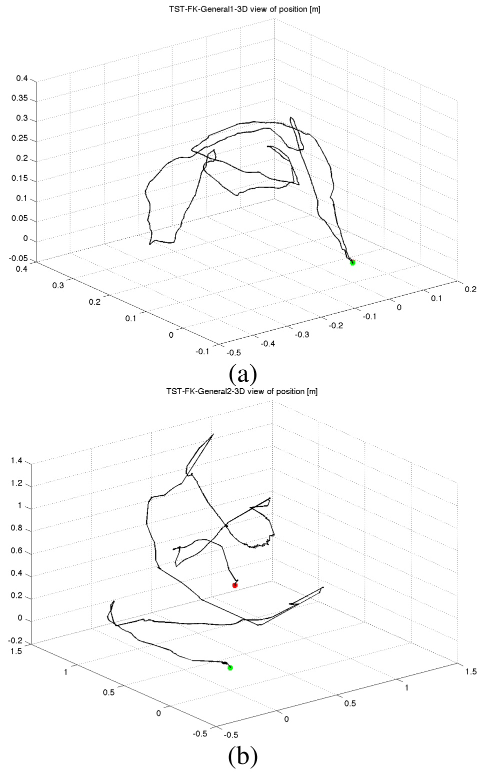
DRIFTS BETWEEN THE START AND THE ENDPOINTS OF THE RECOVERED TRAJECTORY ON SEVERAL DATASETS. *Note: For the handheld sequences a true path length is not available. Therefore, drifts have been normalized by a common estimated length to give a relative measure.*

Path Name	Path Length	Drift MLK/MK	Drift TST/FK
Hand-held motion 1	N/A (55s)	0.09%	0.06%
Hand-held motion 2	N/A (63s)	10.6 %	5.57%
Walking motion	292m (281s)	5.24%	1.01%

B. Closed-loop trajectory experiments

All the followed paths are closed loops in which the starting and end points and orientations are approximately the same. This allows us to evaluate our system by drift on loop-closure at the end of the dataset. As reported in Table I, the combination of TST and full kinematics (FK) used in our system outperforms the combination of the the MLK tracker with the minimal kinematics (MK) proposed in [3].

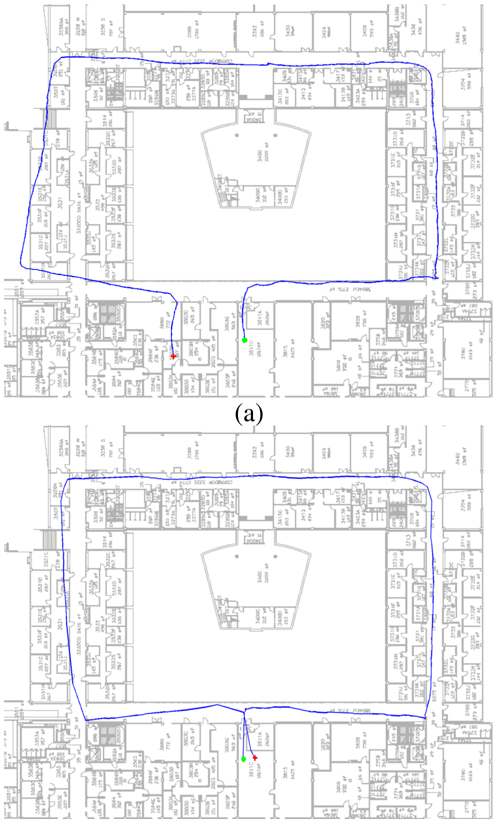
The first hand-held dataset follows simple, general motion for approximately one minute in front of a scene with easily trackable features. This serves as baseline of the ideal case for both our system, and the simpler model with minimal kinematics and the MLK tracker against which we compare. The second hand-held dataset is the sequence with very challenging scene geometry discussed in section II. This sequence includes significant changes in viewpoint of continuously tracked surfaces and many challenging occlusion cases. Here, as expected, the use of TST allows us to significantly outperform the MLK-based system. Figure 4 shows the 3D trajectories of both of these datasets using our system. Note the accuracy of loop closure in the first handheld dataset (a) and the complexity of the motion undertaken. Figure 4 (b) shows our result on the second, much more challenging hand-held dataset. The walking motion experiment is a significantly challenging dataset comprised of brisk walking using the head-mounted system through the hallways of a rectangular building. Throughout the dataset the operator frequently changes gaze direction, simulating the sudden rotational motions discussed in section IV-B. On this challenging example of humanoid legged motion we achieve a drift of 1.01% over a 292m course. Figure 5 compares the trajectories obtained by our system and MLK with minimal



(a)

(b)

Fig. 4. Results of 3D trajectories for the two hand-held experiments((a) and (b) for the first and second respectively). The green dot is the starting location. The bottom case undergoes complex motions in an environment with very challenging geometry continuously over the course of more than one minute. Our system is a significant improvement over MLK with minimal kinematics in this case.



(a)

(b)

Fig. 5. Comparison of results on 292 m loop closure test of Multiscale Lucas-Kanade (MLK) with minimal kinematics (a) vs TST with full kinematics (b). The green dot is the starting location. The correct path follows the simple rectangular loop through the hallways of the building and returns to the starting point. No planar motion assumptions are used. Note the significantly more correct path estimations.

kinematics system when overlaid on the floor plan of the rectangular building.

While Fig. 3 showed the challenges of the accelerations our system undergoes compared to hand-held datasets, Fig. 6 shows the challenges induced by active gaze control on estimating orientation. In order to better illustrate this difference a second walking traversal of the indoor loop was performed, however with gaze direction remaining fixed along the direction of motion and slower walking. This allows us to compare the highly variable vertical-axis rotation estimates (heading) on our challenging walking dataset (Fig. 6 (a)) with the very stable heading estimates from the simplified experiment (Fig. 6 (b)). Due to the challenges in physically closing-the-loop in orientation for a head-mounted system, the drift in the vertical component of orientation in (a) is likely primarily caused by experimenter error as opposed to rotational drift.

In all cases, our system using TST with full kinematics is a significant improvement over the MLK tracker and minimal kinematics.

Finally, we report a comparison (Tab. II) of reported drifts from other state-of-the-art ego-motion estimation systems on their long traversal datasets [6], [3], [4], [5]. The majority of these results are for systems mounted on a wheeled platform, which do not encounter the sharp accelerations caused by a walking gait nor the sharp camera rotations seen in our data. Comparisons to hand-held motions have already been made in section V-A. In spite of the significantly more challenging nature of our experiments (walking ego-motion estimation with frequent, sudden rotations) we remain competitive with the state of the art in wheeled/hand-held ego-motion estimation under less challenging conditions. Note that we are using only a single monocular camera, not stereo cameras as in [6], [5], nor a very expensive and precise IMU as in [3]. These additions would only improve our results, however we can achieve comparable results without them. The results shown from [3] were on a dataset taken in the same environment, but with a high-end IMU and on a wheeled instead of walking platform.

VI. Conclusion

We have successfully developed and tested an ego-motion estimation system ideally suited to the walking motion of humanoid robots. The first challenge of significant scale changes due to forward motions in indoor environments has been handled via TST, a recent sparse multi-scale feature tracking technique. Secondly, we have used a kinematic-statistical model that does not require accurate knowledge

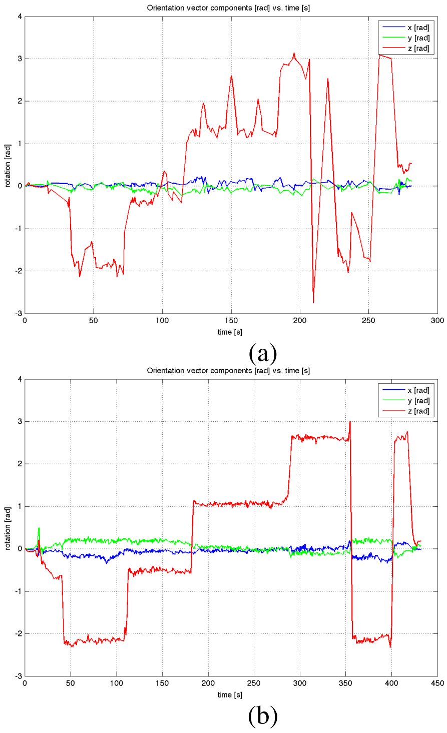


Fig. 6. Results of orientation vector estimates for for the challenging walking motion experiment (a) and the simplified motion with fixed heading (b). Note the significant orientation changes caused by head movement in (a) compared to the stable heading used in (b). The large spike at 210s in (a) is due to the wraparound in magnitude of the rotation vector.

TABLE II
COMPARISON OF DRIFT STATISTICS WITH STATE-OF-THE-ART. DRIFT SHOWN AS PERCENTAGE OF TOTAL PATH LENGTH.

Method	Drift: (%)
Konolige and Agraval, 2008 (Stereo and IMU, wheeled platform)	0.3
Jones and Soatto, 2011 (Mono and IMU, wheeled platform)	0.2
Mourikis and Roumeliotis, 2007 (Mono and IMU, wheeled platform)	0.31
Oskiper *et al.*, 2007 (Double Stereo and IMU, hand-held camera)	0.79
Ours (Mono and IMU, fast walking motion)	1.01

of robot dynamics and calibration in order to handle the challenges caused by the sharp and periodic accelerations associated with walking gaits. Our system runs in realtime on a commodity laptop and uses a low-cost inertial measurement unit. In spite of the poorer data quality from this low-grade IMU, we have achieved comparable performance to the state-of-the-art reported drift results on our own datasets containing more challenging motion than other systems were tested on at a fraction of the total system cost.

REFERENCES

[1] A. Vedaldi, G. Guidi, and S. Soatto, "Moving forward in structure from motion," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[2] T. Lee and S. Soatto, "Video-based descriptors for object recognition," *Image and Vision Computing*, 2011.

[3] E. Jones and S. Soatto, "Visual-inertial navigation, mapping and localization: A scalable real-time causal approach," *International Journal of Robotics Research*, January 2011.

[4] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint Kalman filter for vision-aided inertial navigation," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Rome, Italy, April 10-14 2007, pp. 3565–3572.

[5] T. Oskiper, Z. Zhu, S. Samarasekera, and R. Kumar, "Visual odometry system using multiple stereo cameras and inertial measurement unit," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[6] K. Konolige and M. Agrawal, "Frameslam: From bundle adjustment to real-time visual mapping," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1066–1077, Oct. 2008.

[7] A. J. Davison, "Real-time simultaneous localisation and mapping with a single camera," in *ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision*, 2003.

[8] A. Comport, E. Malis, and P. Rives, "Accurate quadrifocal tracking for robust 3d visual odometry," in *IEEE International Conference on Robotics and Automation*, 2007, pp. 40–45.

[9] E. Mouragnon, M. Lhuillier, M. Dhome, F. Dekeyser, and P. Sayd, "Realtime localization and 3d reconstruction," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2006, p. 10271031.

[10] G. Klein and D. Murray, "Improving the Agility of Keyframe-based SLAM," in *In Proc. European Conference on Computer Vision (ECCV)*, 2008.

[11] O. Stasse, A. J. Davison, R. Sellaouti, and Y. Kazuhito, "Real-Time 3D SLAM for a Humanoid Robot considering Pattern Generator Information," in *Proc. of the 2006 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2006.

[12] A. Pretto, E. Menegatti, M. Bennewitz, W. Burgard, and E. Pagello, "A visual odometry framework robust to motion blur," in *Proc. of the IEEE International Conference on Robotics and Automation (ICRA09)*, 2009.

[13] D. Nistér, O. Naroditsky, and J. Bergen, "Visual odometry," in *Proc. of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004.

[14] E. S. Jones, A. Vedaldi, and S. Soatto, "Inertial structure from motion and autocalibration," in *Workshop on Dynamical Vision*, October 2007.

[15] J. Kelly and G. S. Sukhatme, "Visual-inertial sensor fusion: Localization, mapping and sensor-to-sensor self-calibration," *International Journal of Robotics Research*, vol. 30, no. 1, pp. 56–79, January 2011.

[16] D. Scaramuzza and F. Fraundorfer, "Visual odometry: Part i - the first 30 years and fundamentals," *IEEE Robotics and Automation Magazine*, vol. 18, 2011.

[17] T. Lindeberg, "Principles for automatic scale selection," KTH, Stockholm, CVAP, Tech. Rep., 1998.

[18] T. Poston and I. Stewart, *Catastrophe theory and its applications*. London: Pitman, 1978.

[19] S. Soatto, *Steps Toward a Theory of Visual Information*. http://arxiv.org/abs/1110.2053, Technical Report UCLA-CSD100028, September 13, 2010 2010.

[20] B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *DARPA81*, 1981, pp. 121–130.

[21] S. Baker and I. Matthews, "Equivalence and efficiency of image alignment algorithms," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, 2001.

[22] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[23] E. Jones, "Large scale visual navigation and community map building," Ph.D. dissertation, University of California at Los Angeles, June 2009.

[24] A. Mourikis, "Characterization and optimization of the accuracy of mobile robot localization," Ph.D. dissertation, ProQuest, 2008.

[25] Y. Ma, S. Soatto, J. Kosecka, and S. Sastry, *An invitation to 3D vision, from images to models*. Springer Verlag, 2003.

[26] M. Bartlett, *An Introduction to Stochastic Processes*. Cambridge University Press, 1956.

[27] P. Huber, *Robust statistics*. New York: Wiley, 1981.