

THE UNIVERSITY of EDINBURGH

Edinburgh Research Explorer

Drift-free humanoid state estimation fusing kinematic, inertial and LIDAR sensing

Citation for published version:
Fallon, MF, Antone, M, Roy, N & Teller, S 2014, Drift-free humanoid state estimation fusing kinematic, inertial and LIDAR sensing. in *Humanoid Robots (Humanoids), 2014 14th IEEE-RAS International Conference on.* Institute of Electrical and Electronics Engineers (IEEE), pp. 112-119. https://doi.org/10.1109/HUMANOIDS.2014.7041346

Digital Object Identifier (DOI):
10.1109/HUMANOIDS.2014.7041346

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

Published In:
Humanoid Robots (Humanoids), 2014 14th IEEE-RAS International Conference on

General rights
Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.

OPEN ACCESS

Download date: 29. Mar. 2024

Drift-Free Humanoid State Estimation fusing Kinematic, Inertial and LIDAR sensing

Maurice F. Fallon[1], Matthew Antone, Nicholas Roy and Seth Teller[2]

***Abstract*— This paper describes an algorithm for the probabilistic fusion of sensor data from a variety of modalities (inertial, kinematic and LIDAR) to produce a single consistent position estimate for a walking humanoid. Of specific interest is our approach for continuous LIDAR-based localization which maintains reliable drift-free alignment to a prior map using a Gaussian Particle Filter. This module can be bootstrapped by constructing the map on-the-fly and performs robustly in a variety of challenging field situations. We also discuss a two-tier estimation hierarchy which preserves registration to this map and other objects in the robot's vicinity while also contributing to direct low-level control of a Boston Dynamics Atlas robot. Extensive experimental demonstrations illustrate how the approach can enable the humanoid to walk over uneven terrain without stopping (for tens of minutes), which would otherwise not be possible. We characterize the performance of the estimator for each sensor modality and discuss the computational requirements.**

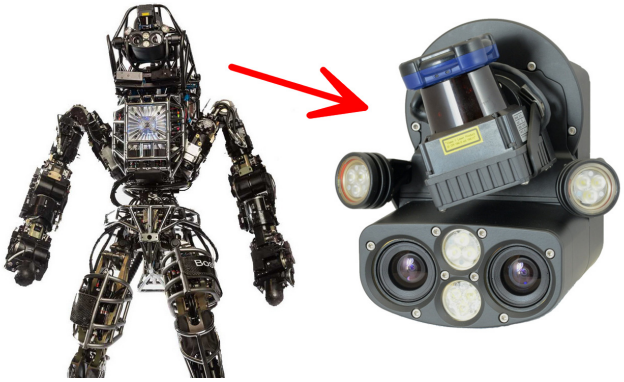


Fig. 1. The Atlas robot contains 28 hydraulically actuated joints, and its primary sensing is provided by the Carnegie Robotics Multisense SL sensor head which is equipped with a rotating LIDAR scanner and a stereo camera. (photo credits: Boston Dynamics and CRL)

I. Introduction

Dynamic locomotion of legged robotic systems remains an open and challenging research problem whose solution will enable humanoids to perform tasks in and reach places inaccessible to wheeled or tracked robots. Several research institutions are developing walking and running robots with a range of form factors, from all-terrain quadrupeds operating outdoors to experimental bipedal runners which have yet to leave the laboratory.

Locomotion is a typical closed-loop control problem whose primary input consists of the state of the robot — namely, the 6-DOF pose and velocity of its pelvis, as well as the configuration of its joints. Accurate, timely estimates of the robot state not only facilitate effective control for dynamic whole-body motions such as walking, but also enable a greater degree of task autonomy via consistent knowledge of the surrounding environment and the locations of objects within it.

A. Related Work

One common class of estimation method is based on dynamics (e.g., [1]), and relies on knowledge of the controller outputs and a motion model of the humanoid to infer the state of the robot's centers of mass and pressure. Errors in link center-of-mass modeling and the presence of unpredictable forces (e.g., from the robot's support/power tether or external contacts) are accounted for by appending an additional process model for each class of disturbance.

In [2], the authors extend this approach and apply it to the Atlas robot (which we are also using in our work). They discuss the computational challenges of formulating a single extended Kalman filter (EKF) for a humanoid with many degrees of freedom, and propose instead to estimate the pelvis position and joint dynamics in separate filters.

An EKF-based estimator is presented in [3] for a quadruped that uses a sensor-based prediction model and creates filter corrections using foothold measurements. This approach incorporates the positions of footholds into the state vector (using a point model for each foot) and gives particular consideration to consistency and observability analysis. Recently this approach was extended to bipedal locomotion [4] with results presented on a simulated SARCOS robot. The primary contribution was extension of the algorithm to a biped with a full foot plate, which requires a 6 DOF constraint on the foot frame.

Finally, there has also been work in coupling odometry estimates to a higher-level navigation system. Localization of a quadruped (in 3 DOF) against a prior terrain map was explored by [5] using a particle filter.

All of the above works utilize only proprioceptive sensing. While visual mapping is an active area of research — for instance [6] demonstrated loop-closing and visual keyframe registration to known landmarks in a laboratory setting — it has not been widely adopted for field operations. High computational cost, latency, and sensitivity to environmental conditions (e.g., illumination and visual texture content) all present substantial challenges for robust vision-based mapping onboard a humanoid. Hornung *et al.* [7] utilized a LIDAR sensor to localize an Aldebaran NAO robot within a

[2]The authors are with the Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, USA. Maurice F. Fallon is also with the Department of Informatics, University of Edinburgh, UK. mfallon,antone,nickroy,teller@csail.mit.edu

multi-level environment, thus circumventing many problems inherent to visual sensing; our work also adopts LIDAR as the primary exteroceptive sensor for localization.

B. Overview

This paper presents a state estimation algorithm that combines measurements from three distinct sensing modalities — inertial, leg kinematics, and LIDAR — into a single consistent estimate of the robot's pelvis link via probabilistic fusion. The estimator does not require elaborate dynamics models; like [4], we couple foot placements and leg kinematics with inertial predictions in an EKF framework, and like [2], we consider pelvis pose separately from joint states. Our primary novel contributions are (1) incorporation of exteroceptive sensing to achieve reliable drift-free alignment to a prior map while walking using a Gaussian Particle Filter (GPF) to apply position corrections derived from each LIDAR scan; and (2) extensive experimental validation of the algorithm on a real humanoid robot.

The estimator is demonstrated using the Boston Dynamics (BDI) Atlas humanoid (Figure 1) provided to our research team for the ongoing DARPA Robotics Challenge (DRC). The LIDAR-based localization component was specifically motivated by the slow locomotion rates achieved in the DRC Trials in December 2013: due to position drift on the order of 2 cm per footstep, teams typically took just two steps at a time to traverse uneven terrain, pausing periodically to manually re-localize the robot and to create new motion plans with respect to the environment. As we push toward enabling greater autonomy in task execution — e.g., walking for several minutes at a time with manipulation actions interspersed — a continuous localization capability becomes critically important, as it allows the robot to retain accurate and consistent reference to terrain maps and objects of interest in its vicinity.

In Section II we present an overview of our requirements for state estimation and discuss two different use cases for our approach. Then in Sections III–V we discuss how each sensor stream can be abstracted to a basic probabilistic measurement suitable for fusion.

Finally in Section VII we benchmark the performance of the algorithm and present results from a series of extended duration walking experiments, executed passively with BDI's native controller totaling approximately one hour. A clear operational benefit is demonstrated: alignment to the prior model enables the robot to continuously traverse uneven terrain without stopping, and thus operate up to four times more quickly than previously possible.

II. Requirements

The ultimate goal of our research efforts [8] is to develop a system that enables a humanoid robot to operate at a semi-autonomous level with human interaction at a task level, as depicted in Figure 2: "walk over to the drill and use it to cut a circular hole in the wall".

Executing such actions requires the ability to precisely and continuously localize, thus enabling the robot to walk to a goal without interruption. An alternative would be to encode safety sequences such as stopping short of a goal and then stepping conservatively into position which, as well as being inefficient, adds complexity to an otherwise simple action.

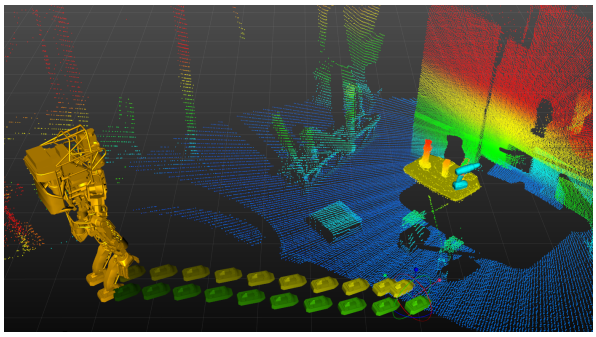


Fig. 2. A 3D rendering of the robot with a footstep plan leading towards a fitted model of a (cyan) drill on a (yellow) table.

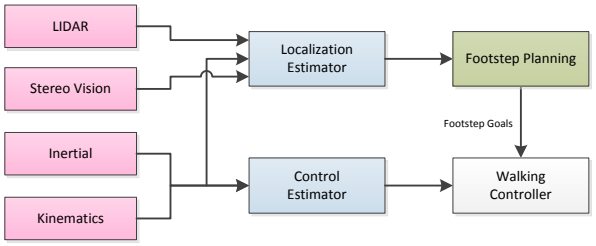


Fig. 3. To allow possibly discontinuous exteroception corrections we propose a hierarchy in which these corrections are fed to the walking controller indirectly and at low rate (in this case as footstep goals via our planner). Note that the integration of stereo vision is ongoing work.

To achieve this, we envisage a navigation system that comprises two concurrent state estimators (see Figure 3). The first estimator, used within our closed-loop locomotion controller [9], produces a stable estimate of position and velocity with high rate (333Hz) and low latency and, crucially, without any discontinuities such as would be produced by a map alignment correction.

However, proprioception is subject to drift and thus is not amenable to task-level autonomy, which requires constant long-term localization within the environment but can better tolerate small instantaneous adjustments. We therefore maintain a second localization estimate that incorporates exteroception data from LIDAR (or vision sensors) to remove global drift, but can allow discontinuous corrections of the robot's position. (These corrections would typically be 1cm or less, and applied at the 40Hz framerate of the LIDAR). This two-tiered approach, which is in the spirit of [10], [11], maintains reference to higher-level features of interest used by our robot/operator team. Here we focus primarily on this second, exteroceptive, localization mode; however, in practice we use the same algorithms and software framework for both modes.

III. Integration

Both estimators utilize the same integrator algorithm, originally described in [12] and used for a micro aerial vehicle

(MAV), with simple software configuration flags enabling or disabling the various input sensors.

Following the notation described therein, we wish to estimate the position and orientation of the robot's kinematic root link, the pelvis, as well as its linear and angular velocities. The full state vector is defined as $x = [w_b^T \ v_b^T \ R \ \triangle^T]^T$ and each component is as follows:

- angular velocity, $w_{b,b} \in \mathbb{R}^3$
- linear velocity, $v_{b,b} \in \mathbb{R}^3$
- orientation, $R_{b,w} \in SO_3$
- position, $\triangle_{b,w} \in \mathbb{R}^3$

Both velocity components are estimated in the (body) pelvis frame, while the position and orientation of the pelvis are expressed in a fixed world frame. We exclude both the robot's contact foot position and the joint states from this state vector and filter them separately (as do others including [2]).

The pelvis of the Atlas humanoid contains the robot's primary Inertial Measurement Unit (IMU), located 9 cm behind the pelvis link position and rotated by 45°. The sensor is a KVH 1750-IMU comprised of Fiber Optic Gyroscopes (FOG) and Micro Electromechanical Systems (MEMS) accelerometers of tactical grade.

The state estimate and its associated covariance are updated using an Extended Kalman Filter (EKF). The prior distribution is propagated using a process model driven by the IMU measurements: the rotation rates w_b and the accelerations a_b are both sensed in the IMU frame and transformed to the body frame[1] before being integrated. As discussed in [12], orientation uncertainty is expressed in exponential coordinates around the body frame.

The IMU sensor provides very accurate raw measurements, but vibrations induced by the hydraulic pressurizer within the robot's torso corrupt the signal. We therefore apply a cascading set of IIR-notch filters to dampen the 85 Hz vibrational component and its harmonics.

Additional states of the EKF are used to maintain rotation rate and acceleration bias estimates, which are computed while the robot stands still at the start of an experiment. Although the estimator supports on-line bias updates, we typically retain these initial values as they tend to remain consistent during a typical experiment.

In the subsequent sections we will describe how each individual sensing modality is used to form Kalman measurement updates to the state vector. This information is summarized in Table I.

Dimension	Pos $\triangle$	Orient R	Velocity v_b	Ang Rate w_b	Accel a_b
Accelerometers					✓
Gyroscopes				✓	
Leg sensing	✗	✗	✓	✗	
LIDAR	✓	✓			

TABLE I

CONTRIBUTION OF VARIOUS SENSORS TO THE FILTERED STATE ESTIMATE. MODES OF INTEGRATION FOUND TO BE USEFUL ARE MARKED ✓AND THOSE NOT USED HERE (FOR A VARIETY OF REASONS) ARE INDICATED ✗.

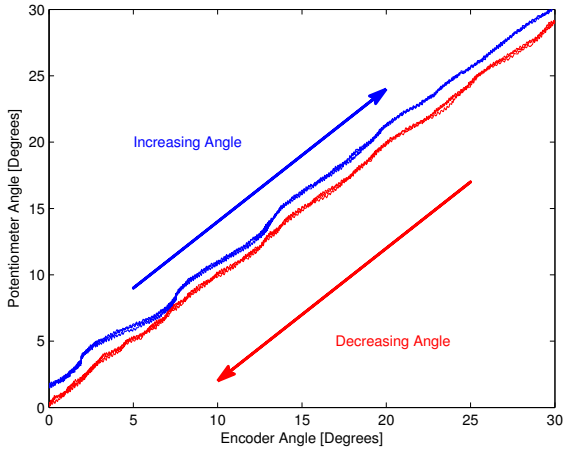


Fig. 4. Illustrative comparison between LVDT potentiometer and direct encoder joint sensing for the robot's left shoulder joint. The joint was actuated between its limits in each direction 3 times illustrating non-linear directional effects. Such effects are not sensed by the leg joint sensors.

IV. LEG KINEMATICS

The robot has two legs, each with six joints: three at the hip, a knee joint and two angle joints. As with many leg kinematic integration algorithms [1], [3], our approach assumes that the robot's stance foot maintains non-slipping contact with the ground during part of the gait and that this foot is stationary. This allows instantaneous velocity and position measurements of the robot's pelvis to be inferred via forward kinematics. Of course in practice perfectly clean and stable ground contact is seldom achieved, we assert that for short periods (the sample time of our sensors) these assumptions are reasonable.

On the Atlas robot, the angle of each leg joint is sensed by measuring the travel of its hydraulic actuator using an LVDT (Linear Variable Differential Transformer) and then computing a transformation through the joint linkage. This model does not account for flexion of the joint linkage when loaded or backlash when a joint changes direction.

While the robot's arm joints are not fully equivalent to the legs, they do contain additional encoder sensors that directly sense the joint angle with lower sample noise. Figure 4 shows a comparison between the LVDT and encoder values measured while continuously actuating the left shoulder joint in each direction. The difference of about a degree illustrates the backlash issue. Because of these un–sensed effects, leg kinematic integration is subject to position drift at rates that vary due to factors such as dynamism of the walking gait and controller execution.

A. Contact Classification

At the base of the locomotion algorithm, a gait transition detector infers the current stage of the walking motion and then decides which of the feet has stationary contact with

[1]The manufacturer-provided orientation estimate was not used but is compared with in Section VII.

the ground. We use a Schmitt trigger with a threshold of 575 N to classify contact forces sensed by the robot's 3-axis foot force-torque sensors and detect whether either foot is in contact. A simple state machine then decides which foot is the most reliably in contact and thus will provide the basis for kinematic measurements. The evolution of the foot contact classifier is demonstrated in the upper plot in Figure 5.

In the specific case of walking up stairs, the toe of the trailing foot can be used to push the robot's pelvis forward and upward while not being in stationary contact (known as "toe off"). In this situation we ensure that the leading foot is assigned to be the primary fixed foot.

We also classify other events in the gait cycle. Striking contact is determined when a rising force of 20–30 N is maintained for more than 5 msec. Breaking contact is determined when force falls below 275 N in the opposite direction. Because these events create unrealistic measurements, the EKF integrates these measurments with higher measurement covariance.

We note that when the robot is in a double support stance, information from both legs could be leveraged to provide additional kinematic measurements. For simplicity we currently neglect this information.

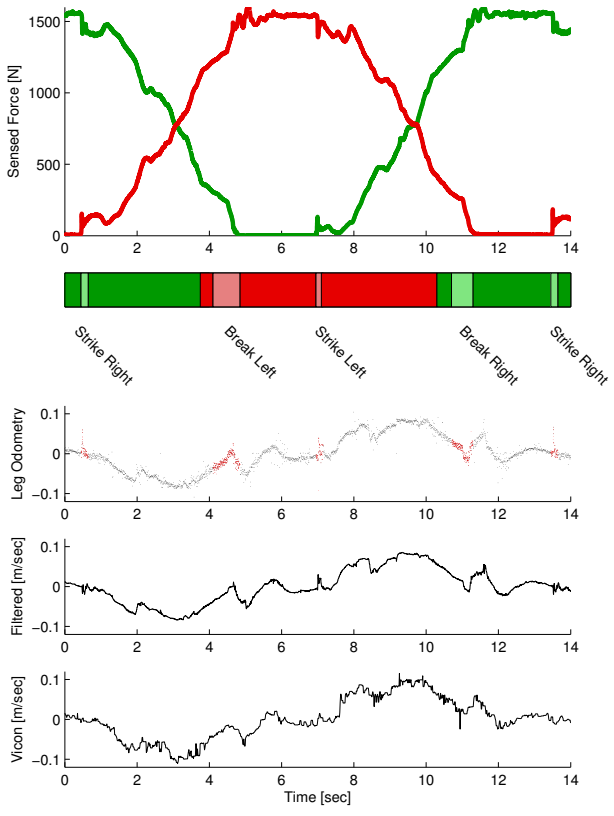


Fig. 5. Top: Evolution of foot force signals for the left (green) and right (red) foot during two steps. Bar Chart: classification of the primary standing foot (light color indicates a ground contact event). Lower plots: pelvis lateral (body Y-direction) velocity estimates for (1) raw leg odometry (the estimate during contact events is shown in red), (2) filter output and (3) VICON-based ground truth.

B. Kinematic Measurements

Given the accuracy of IMU orientation sensors, we choose to use joint sensing to measure the linear position and velocity of the pelvis only. Our input information is the current pelvis orientation estimate in the world frame $R^t_{b,w}$, the previous position of the stationary foot $\triangle^{t-1}_{f,w}$, and the position of the foot relative to the pelvis, as produced by forward kinematics $T^t_{f,b}$.

Dropping the time index, consider the pelvis located at the origin of a convience coordinate frame aligned with the true world frame, $T_{p,w'} = \begin{bmatrix} R_{b,w} & 0_{3,1} \\ 0 & 1 \end{bmatrix}$. Within this convience frame, the orientation of the foot is given by

$$T_{f,w'} = T_{b,w'} T_{f,b} = \begin{bmatrix} R_{f,w'} & \triangle_{f,w'} \\ 0 & 1 \end{bmatrix} \quad (1)$$

By effectively assuming that the pelvis orientation estimate has zero covariance, the foot orientation in the world frame is then simply $R_{f,w} = R_{f,w'}$.

Finally pairing this orientation with the fixed foot position $\triangle^{t-1}_{f,w}$, the position of the pelvis in the world frame and be recovered via

$$T_{p,w} = T_{f,w} T_{p,f} = T_{f,w} (T_{f,p})^{-1} \quad (2)$$

where $T_{f,w} = \begin{bmatrix} R_{f,w} & \triangle_{f,w} \\ 0 & 1 \end{bmatrix}$.

Two types of filter measurement could be formulated using this position estimate. The simplest approach would be directly apply this as a position measurement within the EKF. However, because of the inconsistencies in joint sensing and because the robot's foot does not make and maintain perfectly clean static contact with the ground we avoid this approach.

Alternatively we can use the difference between consecutive position estimates over a short period of time to create a velocity measurement of the pelvis frame

$$\hat{v}^t_b = \frac{\triangle^t_b - \triangle^{t-1}_b}{T_s} \quad (3)$$

where T_s is the integration period, typically 3 msec.

This approach is more attractive because each resultant observation is a discrete measurement of the robot's velocity and does not retain any accumulated history e.g. the effect of non-ideal ground contact, e.g., the footplate rolling or sliding. The influence of an erroneous velocity is transient and quickly corrected by subsequent observations.

Using both measurement types together would be comparable to the approach in [2], but we avoid doing so as it raises the possibility of creating inconsistencies, particularly when combined with position measurements derived from the LIDAR module (presented in the following section).

We note that our approach neither assumes knowledge of the terrain surface normal that robot is standing upon nor attempts to maintain a consistent estimate of the foot orientation over time.

Figure 5 contains a number of plots comparing (1) the raw pelvis velocity measurements inferred from kinematics

with (2) the output of our integrating filter and (3) the velocity estimated from VICON motion capture. Typical pelvis velocity standard deviations, measured when standing still, are as follows:

- Raw incoming kinematics: 7.6 cm/sec
- After joint level filtering: 2.3 cm/sec
- After EKF integration: 1.4 cm/sec

V. LIDAR MEASUREMENTS

As illustrated in Figure 1, the robot is equipped with a Multisense SL sensor head designed by Carnegie Robotics which combines a fixed binocular stereo camera with a Hokuyo UTM-30LX-EW planar LIDAR sensor mounted on a spindle that can rotate up to 30 RPM. (We typically operate the device at 5 RPM to densely sample the terrain when walking). The LIDAR captures 40 scan lines of the environment per second, each containing 1081 range returns out to a maximum range of 30 meters. The entire head can pitch up and down (powered by a hydraulic actuator) but cannot yaw or roll.

Our projection of LIDAR range returns as points in the 3D workspace accounts for the robot's motion, and more importantly, the spindle rotation during the 1/40 second scanning period of the LIDAR's internal mirror. Neglecting this effect would result in mis-projections of returns to the side of the robot by as much at 2.5 m at the highest spindle rotation speed. Accurate projection also requires precise calibration of the LIDAR sensor, as discussed in [8].

A. Contribution to Estimation

Our strategy is to use the LIDAR to *continuously* infer the robot's position relative to a prior map while walking. We cannot assume that the sensor is oriented horizontally [13], nor can we afford time to stop moving and perform static 3D registration, e.g., using an Iterative Closest Point algorithm [14]. Instead we aim to incorporate information from each individual LIDAR scan into the state estimate using a Gaussian Particle Filter, as originally described in [12].

In typical operation, the robot is first commanded to stand still for between 10 and 30 seconds while it collects a full 3D point cloud of its environment (see Figure 6). This cloud is then converted into a probabilistic occupancy grid (OctoMap [15]) against which efficient localization comparisons are later performed. While the MAV experiments presented in [12] required offline mapping with a separate sensor, our legged humanoid and actuated LIDAR with 30 m range permit the map to be constructed immediately prior to operation and utilized during the entire task[2]. This makes our approach practical both for highly variable laboratory experiments and for field trials in which the environment is initially unknown. Furthermore, if the robot were to approach the map boundary, on-line construction of a new map could easily be performed in situ.

Since the LIDAR is fundamentally a planar 2D sensor, only a subset of the state vector (namely x, y and yaw in the current sensor plane) is observable at any given instant. We therefore partition the full state vector into observable and unobservable components, and use a GPF to incorporate each laser measurement over the observable variables. The particle filter samples are weighted according to the proposed sub-state likelihood, which is computed by comparing the LIDAR measurements, projected from the sub-state, to the prior map. From these weighted samples a mean and covariance, and in turn an equivalent Kalman measurement update for the full state vector are calculated resulting in a correction to the position $\triangle$ and the yaw.

[2]The DRC Trials terrain course was approximately 15 m in length.

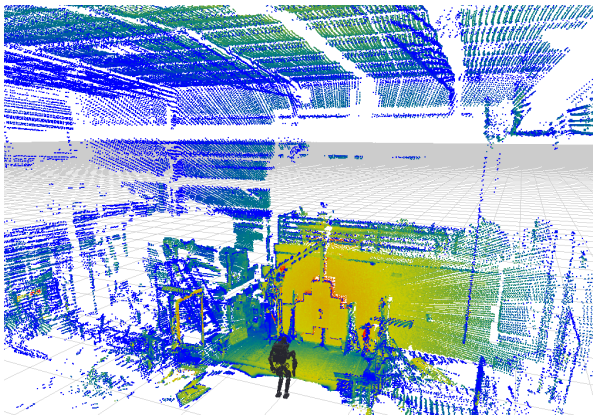


Fig. 6. The robot initially collects a static LIDAR point cloud of its environment, which is then converted into an occupancy map for subsequent localization.

B. Latency and Computation

To utilize this sensor modality in a real-time system requires careful consideration of latency. The LIDAR range measurements require significantly more time to be sensed and processed, which introduces significant latency relative to the 1kHz kinematic and inertial information. These latencies are as follows, for a 3.3GHz 12-core desktop PC:

Component	Latency	Frequency
Lower joint Kalman Filters	0.16 msec	1 kHz
Pose Extended Kalman Filter	0.54 msec	333 Hz
LIDAR data transmission	7 msec	40 Hz
GPF processing time	11.4 msec	40 Hz
Overall LIDAR latency	18.4 msec	40 Hz

The values and the experiments presented in Section VII use 1000 GPF samples, although reliable performance (and reduced latency) is possible with just 300 samples.

We use a multi-process messaging architecture to parallelize computation, with the GPF algorithm requiring a single CPU core. Within the estimator, the EKF retains a 1 sec history of measurements to accommodate the LIDAR/GPF latency with the corrections made to the appropriate filter state followed by re-filtering of all newer kinematic and inertial measurements.

C. Reliability and Practicality

We also considered the reliability of each modality within the DRC competition context. A practical issue with the

inertial sensor is that the robot must be completely stationary during initialization. Boston Dynamics' estimator requires initialization as soon as the robot is first powered on, typically with its feet solidly contacting the ground. Our proposed estimator provides greater flexibility, as it can be initialized at any point when the robot is standing and deemed to be stationary.

While the time taken to construct the LIDAR map (10–30 sec) is a minor inconvenience, in all datasets and task scenarios available to us there was sufficient stationary structure within sensor range for localization to operate reliably. Since the algorithm uses measurements of the entire environment, movement of a few objects or people near the robot has no noticeable effect on performance. However, in scenarios containing substantial background motion (such as the crowds attending the DRC Trials), special consideration may be required to disregard portions of the map with significant activity.

Finally, while our preference has been to build a map from a single location and use it continuously while operating, in Section VII-B we demonstrate that the GPF localization algorithm is robust to non-encompassing maps as well.

VI. Stereo Vision

While the Multisense SL contains a stereo camera capturing 1-MPix images at up to 15 Hz with an 80° x 80° field of view, in this paper we do not focus on the use of vision within our state estimation. Figure 8 illustrates challenging but realistic scenes from the DRC Trials — containing strong shadows, false feature detections, and abrupt lighting changes — all of which are detrimental to vision algorithms. Computational requirements and latency are also not insignificant, so we have focused efforts to date on exteroceptive measurements from LIDAR alone.

Nonetheless, we have also experimented with stereoscopic sensing as a potential complementary modality. We benchmarked the performance of FOVIS [16], a state-of-the-art visual odometry system, by post-processing color and disparity images generated at 10Hz during a typical walking experiment (Experiment 1 in Figure 9) to produce the plot in Figure 7. In a laboratory setting and with a smooth walking motion, FOVIS produces negligible drift and retains alignment to key-frames for several steps at a time, with $\sim$ 150 msec typical latency. Further description of the use of vision in our system is left for future work.

VII. Experiments

In this section we demonstrate the performance of the state estimator through a variety of experiments.

In each case our team's footstep planner (described in [8] and further developed in [17]) creates a kinematically-feasible footstep sequence that reaches a goal position and orientation while minimizing the number of steps taken. The experiments involving uneven terrain traversal require human input to ensure correct initial footstep placement.

In our first set of experiments, the system sequentially feeds footsteps to the Boston Dynamics walking controller

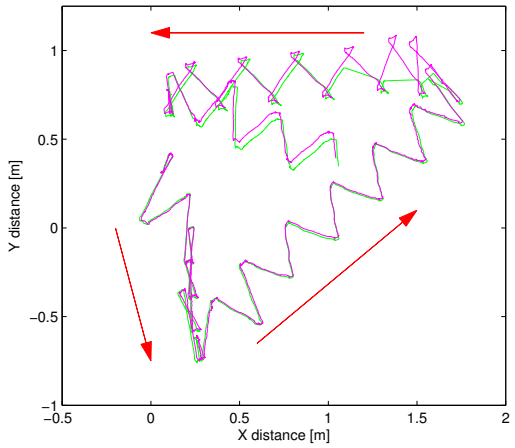


Fig. 7. Overhead view of open-loop visual odometry (magenta) and VICON-based ground truth (green)—both estimating the position of the pelvis link—for a typical walking gait. The robot starts at (0, 0.4) and faces predominantly to the right throughout. After 40 steps, the position estimate drifted by only 3 cm. Note that the robot was outside of the VICON field of view for a short time in the upper right.



Fig. 8. In realistic field scenarios such as the DRC Trials, strong shadows and self observations result in false visual features (shown in red).

(with its own internal state estimate) for execution[3] while continually modifying their positions during execution so that the controller achieves the intended motion. Figure 9 summarizes our results for a variety of walking patterns totaling 57 minutes of operation and 155 meters traveled.

The kinematic-only estimates (both our own and the manufacturer's) are seen to continuously drift, typically at 1.2–1.5 cm per step. This drift rate generally disimproves when the locomotion is atypical: up inclines, more dynamic, or with extended length steps. Orientation estimation performance is comparable across the different estimators. (Note that the precision of the ground truth orientation, via VICON, is on the order of 1°, so a precise comparison is not possible).

As the reader can see in Figure 9 the value of fusing LIDAR-based corrections becomes evident after just a few steps [4]. Walking for just 10 mins, the kinematic-only esti-

[3]The controller provides both a quasi-static step mode and a dynamic walking mode; we almost exclusively use the former.

[4]In the manipulation experiment, the LIDAR contribution actually degrades performance slightly; for this reason, and for general stability, we now discard LIDAR data when standing still.

mators drift by as much as a meter while the LIDAR aided approach remains accurate to within a 2cm throughout.

A. Continuous Terrain Traversal

In an experiment of specific note, depicted in Figure 10, the robot traverses a set of cinder blocks that matches the terrain course from the December 2013 DRC Trials. The LIDAR-aided localization estimate remains accurate to within 2 cm at all times, enabling the robot to continuously walk over the blocks. It can even execute this sequence in reverse to return to its exact starting position, despite having no rear-facing sensors, because the forward-facing sensors keep the robot so precisely localized. This forward-backward traversal was repeated 4 times in 12 minutes. By comparison, the kinematic-only estimates drift continuously, culminating in a total of 0.8 m accumulated error, and would have caused the robot to fall after its first few steps.

During the December 2013 DRC Trials, most teams executed the terrain course two steps at a time because of this drift (see [8] for our analysis of the task). Extrapolating from this experiment, the proposed algorithm can enable execution about four times more quickly than was achieved in the DRC Trials.

B. Localization in Partial Maps

In a second noteworthy experiment, captured in a video accompanying this paper, we explored the performance of the estimator when the robot is not fully surrounded by the prior map. The field of view of the LIDAR sensor roughly accommodates mapping the hemisphere in front of the robot from a single standing position. In this experiment the robot first created a map from its starting pose and then turned 90 degrees before walking for 12 minutes, such that a significant portion of the LIDAR data fell outside of the map. Localization performance (Experiment 2 in Figure 9) was much the same as in the other experiments, further demonstrating the robustness of the proposed approach.

The likelihood function underlying the GPF is computed as the product of the likelihood of each LIDAR return in a scan. When a return falls far from an occupied cell of the OctoMap, an unobserved likelihood is applied to this return, resulting in stable performance despite a high percentage of potentially corruptive outlier measurements.

VIII. Conclusions

We have presented a probabilistic fusion algorithm for humanoid state estimation and characterized the contribution of inertial, kinematic and LIDAR sensing to that estimate. Of particular note is that the approach supports continuous *drift free* localization within a 3D LIDAR map, which we aim to use for enabling longer-term semi-autonomous operation of the BDI Atlas humanoid robot. A key result is that our approach allows the robot to continuously walk over complex terrain while precisely achieving all of its footstep placements, which was demonstrated in a variety of extended experiments and would not have been possible without the proposed algorithm.

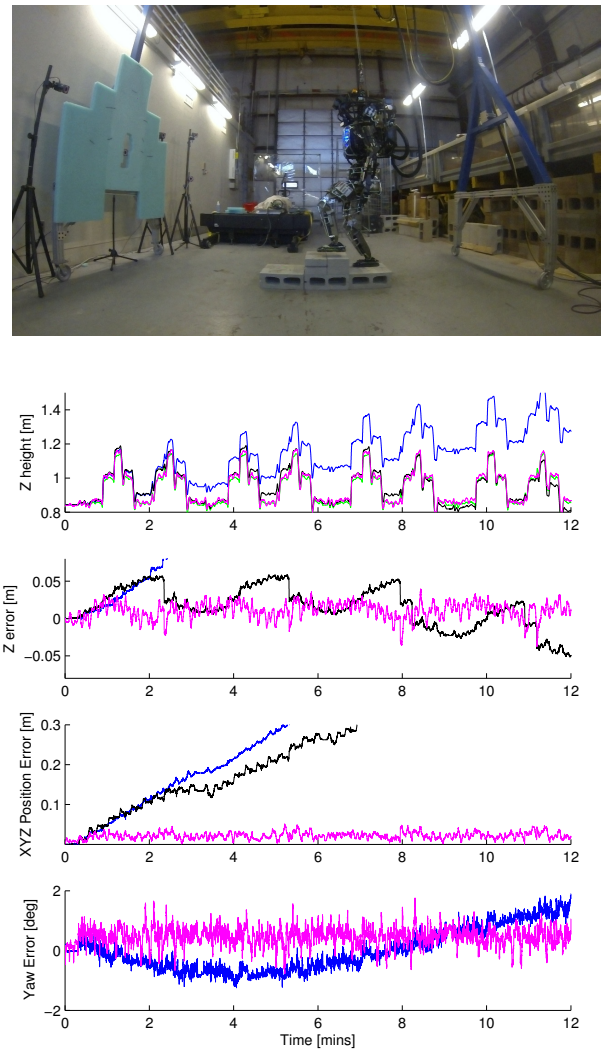


Fig. 10. Analysis of continuous and repeated traversal of the terrain illustrated in the upper image. In the upper plot, green is the VICON ground truth; blue is the estimate provided by BDI; black incorporates inertial and kinematic data and behaves similarly; magenta additionally incorporates LIDAR and as a result remains accurately localized. A longer 5-block traversal sequence (both forward and backward) can be seen in the supplementary video accompanying this paper.

In future work we will focus on better characterization of the joint angle biases so as to reduce the rate of kinematic drift as well as incorporating vision as a contributing component (Section VI).

A. Adaptations for Closed Loop Control

In parallel with these efforts, we have now adapted our quadratic program-based walking controller [9] for use in place of the manufacturer-provided controller. The state estimator described here is being used directly in the 200Hz control loop without modification. We also carry out independent Kalman filtering of the 12 lower body joints measurements, which was mentioned briefly above. As yet, we have not moved to integrate the LIDAR-based position correction module into our real-time control.

A selection of videos showing the state estima-

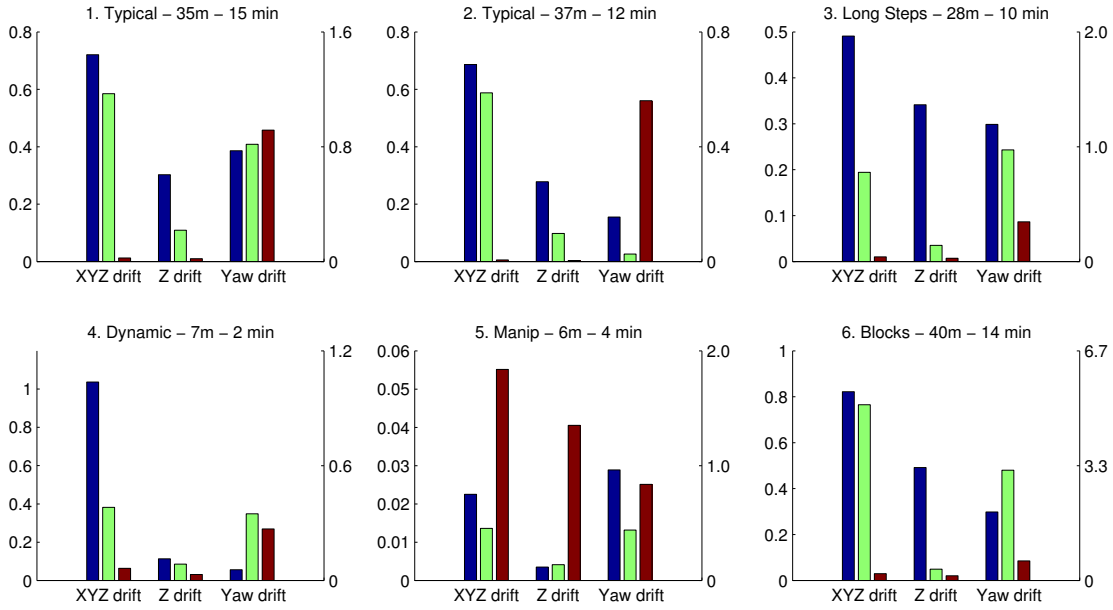


Fig. 9. Summary of localization accuracy for a variety of walking experiments. Position drift is measured with the left scale (in meters) and yaw drift with the right scale (in degrees). Error (versus ground truth) of the BDI state estimator (blue), MIT's kinematic-only (green) and closed-loop LIDAR (red) estimators are shown. Clockwise from upper left: 1: typical gait (15 cm forward steps), 2: typical gait with a partial map (see Section VII-B), 3: long steps (36 cm forward steps), 4: dynamic walking, 5: carrying out manipulation, 6: traversing the cinder block course in Figure 10.

tion combined with the walking controller, as well as all the developed source code, can be accessed at `people.csail.mit.edu/mfallon/state-est`

ACKNOWLEDGMENT

This work was supported by the Defense Advanced Research Projects Agency (via Air Force Research Laboratory award FA8750-12-1-0321). We thank the MIT DRC Team for their work on the overall humanoid system and Adam Bry, Abe Bachrach and Dehann Fourie for development of the core estimation framework.

REFERENCES

[1] B. J. Stephens, "State estimation for force-controlled humanoid balance using simple models in the presence of modeling error," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, May 2011, pp. 3994–3999.

[2] X. Xinjilefu, S. Feng, W. Huang, and C. Atkeson, "Decoupled state estimation for humanoids using full-body dynamics," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, Hong Kong, China, 2014.

[3] M. Bloesch, M. Hutter, M. A. Hoepflinger, S. Leutenegger, C. Gehring, C. D. Remy, and R. Siegwart, "State estimation for legged robots - consistent fusion of leg kinematics and IMU," in *Robotics: Science and Systems (RSS)*, 2012.

[4] N. Rotella, M. Bloesch, L. Righetti, and S. Schaal, "State estimation for a humanoid robot," *CoRR*, vol. 1402.5450, 2014, IROS Submission.

[5] S. Chitta, P. Vernaza, R. Geykhman, and D. Lee, "Proprioceptive localization for a quadrupedal robot on known terrain," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, April 2007, pp. 4582–4587.

[6] O. Stasse, A. J. Davison, R. Sellaouti, and K. Yokoi, "Real-time 3D SLAM for a humanoid robot considering pattern generator information," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2006.

[7] A. Hornung, K. M. Wurm, and M. Bennewitz, "Humanoid robot localization in complex indoor environments," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, Taipei, Taiwan, October 2010.

[8] M. Fallon, S. Kuindersma, S. Karumanchi, M. Antone, T. Schneider, H. Dai, C. P. D'Arpino, R. Deits, M. DiCicco, D. Fourie, T. T. Koolen, P. Marion, M. Posa, A. Valenzuela, K.-T. Yu, J. Shah, K. Iagnemma, R. Tedrake, and S. Teller, "An architecture for online affordance-based perception and whole-body planning," *J. of Field Robotics*, 2014.

[9] S. Kuindersma, F. Permenter, and R. Tedrake, "An efficiently solvable quadratic program for stabilizing dynamic locomotion," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, Hong Kong, China, June 2014.

[10] D. C. Moore, A. S. Huang, M. Walter, E. Olson, L. Fletcher, J. Leonard, and S. Teller, "Simultaneous local and global state estimation for robotic navigation," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2009.

[11] E. Marder-Eppstein, E. Berger, T. Foote, B. Gerkey, and K. Konolige, "The Office Marathon: Robust navigation in an indoor office environment," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, May 2010.

[12] A. Bry, A. Bachrach, and N. Roy, "State estimation for aggressive flight in GPS-denied environments using onboard sensing," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2012, pp. 1–8.

[13] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, "Monte Carlo localization for mobile robots," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, May 1999.

[14] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 14, no. 2, pp. 239–256, 1992.

[15] K. M. Wurm, A. Hornung, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: A probabilistic, flexible, and compact 3D map representation for robotic systems," in *Proc. of the ICRA 2010 Workshop on Best Practice in 3D Perception and Modeling for Mobile Manipulation*, Anchorage, AK, USA, May 2010.

[16] A. Huang, A. Bachrach, P. Henry, M. Krainin, D. Maturana, D. Fox, and N. Roy, "Visual odometry and mapping for autonomous flight using an RGB-D camera," in *Proc. of the Intl. Symp. of Robotics Research (ISRR)*, Flagstaff, USA, Aug. 2011.

[17] R. L. Deits and R. Tedrake, "Computing large convex regions of obstacle-free space through semi-definite programming," in *Workshop on the Algorithmic Fundamentals of Robotics (WAFR)*, Istanbul, Turkey, Aug. 2014.