Direct Visual SLAM Fusing Proprioception for a Humanoid Robot

Raluca Scona[1,2], Simona Nobili[1], Yvan R. Petillot[2], Maurice Fallon[3]

***Abstract*— In this paper we investigate the application of semi-dense visual Simultaneous Localisation and Mapping (SLAM) to the humanoid robotics domain. Challenges of visual SLAM applied to humanoids include the type of dynamic motion executed by the robot, a lack of features in man-made environments and the presence of dynamics in the scene. Previous research on humanoid SLAM focused mostly on feature-based methods which result in sparse environment reconstructions. Instead, we investigate the application of a modern direct method to obtain a semi-dense visually interpretable map which can be used for collision free motion planning. We tackle the challenge of using direct visual SLAM on a humanoid by proposing a more robust pose tracking method. This is formulated as an optimisation problem over a cost function which combines information from the stereo camera and a low-drift kinematic-inertial motion prior. Extensive experimental demonstrations characterise the performance of our method using the NASA Valkyrie humanoid robot in a laboratory environment equipped with a Vicon motion capture system. Our experiments demonstrate pose tracking robustness to challenges such as sudden view change, motion blur in the image, change in illumination and tracking through sequences of featureless areas in the environment. Finally, we provide a qualitative evaluation of our stereo reconstruction against a LIDAR map.**

I. Introduction

For a humanoid robot to carry out useful actions in real environments, it must have comprehensive and consistent situational awareness. For instance, visually tracking the camera pose while building a 3D map allows for localisation in and reasoning about the environment. During robot operation, the Simultaneous Localisation and Mapping (SLAM) system must be robust enough to handle disturbances caused by robot's executed motions and structure sparsity in the environment.

Despite the progress of direct visual SLAM, current methods still struggle in real-world situations. For example, when walking sharp accelerations cause motion blur in the images captured by the robot's camera. The robot may also simply point the camera towards blank walls which lack visual features – this is trivial but fatal as there is no useful structure to localise against. Further issues are changes in illumination and the presence of dynamic elements or people.

Where the performance of camera pose tracking fails, 3D reconstructions become heavily corrupted. Many standard datasets used to evaluate SLAM methods avoid the most challenging situations or use artificial clutter to improve reliability.

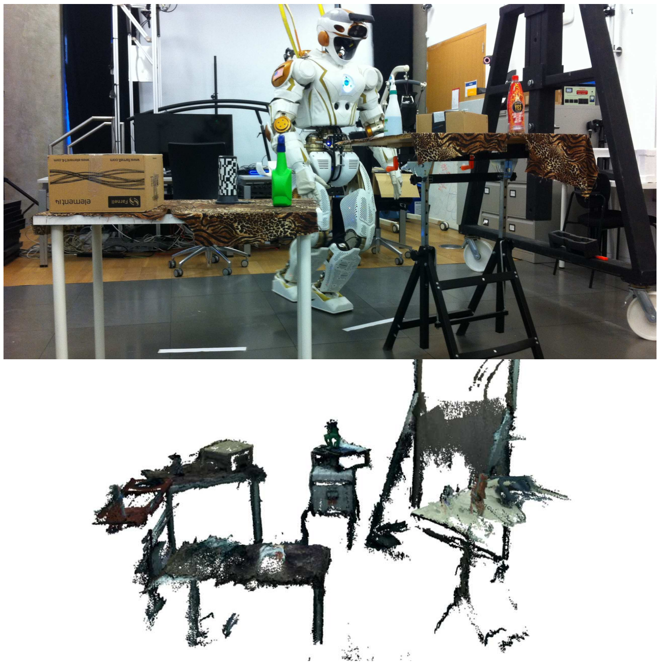


Fig. 1: Top: The NASA Valkyrie humanoid robot during operation in a laboratory environment. Bottom: stereo 3D reconstruction of a manipulation scene covering a 12 m^2 area

Instead, in this work we leverage proprioceptive sensing to aid the visual SLAM system to overcome the challenges stated above.

We extend ElasticFusion [1], which is a dense surfel-based RGB-D SLAM method. Our contribution is a camera pose tracking method which combines the frame-to-model visual tracking of ElasticFusion with a motion prior provided by a low-drift kinematic-inertial state estimator. We compute the pose of the camera by optimising over a cost function which fuses alignment over both geometric and photometric information as well as the kinematic-inertial motion prior.

We discuss current methods in humanoid state estimation and visual SLAM in Section II. Section III gives an overview of our system and describes the individual kinematic-inertial and visual tracking systems. Section IV states our approach. As our robot is equipped with a stereo camera, we first describe pre-processing methods for making this data suitable to be used within ElasticFusion. We then state the mathematical formulation of our robust pose tracking method.

Section V provides an extensive evaluation of our approach on the NASA Valkyrie humanoid robot (Figure 1). Section VI gives the main conclusions and future works of this research.

[1] The authors are with the School of Informatics, University of Edinburgh, UK. {raluca.scona, simona.nobili}@ed.ac.uk

[2] The authors are with the School of Engineering & Physical Sciences, Heriot-Watt University, UK. y.r.petillot@hw.ac.uk

[3] The authors are with the Oxford Robotics Institute, University of Oxford, UK. mfallon@robots.ox.ac.uk

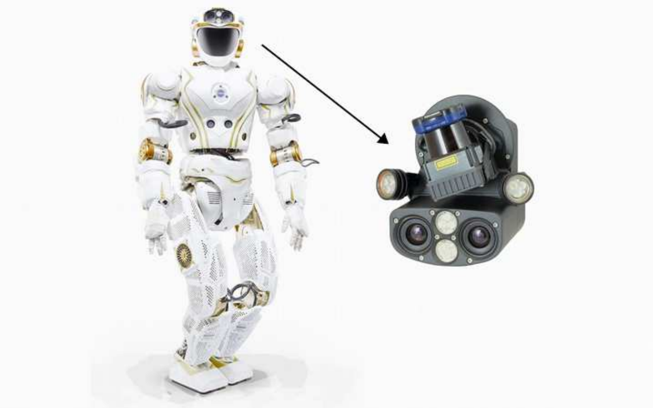


Fig. 2: The NASA Valkyrie is an 1.8 m tall electrically actuated humanoid robot. Within its head is a Carnegie Robotics Multisense SL which combines a rotating LIDAR sensor and a stereo camera. The sensor is inverted on the robot. (photo credits: NASA and CRL)

II. Background

A. Humanoid Kinematic-Inertial State Estimation

A humanoid robot can operate, to a degree, without exteroceptive sensors. Instead it can use a combination of inertial sensing (gyroscopes and accelerometers), kinematic sensing in the legs and force-torque sensing in the feet to estimate its state for control purposes. Using this information, the robot can estimate its position, orientation and velocity at a high-frequency (>200 Hz).

One group of approaches, including [2] and [3], use the inverted pendulum model to estimate the centre of mass (CoM) as this is the quantity of interest for control purposes. The approach has the benefit of explicitly measuring the deviation of the CoM value from its expected value. This allows for the detection of anomalies such as unexpected contact.

Other approaches estimate the motion of a specific link (typically the root link of the kinematic chain) by incorporating the individual sources of information within a filtering framework ([4], [5]). These approaches were successfully demonstrated on the Boston Dynamics Atlas humanoid robot during the DARPA Robotics Challenge.

B. Humanoid Visual Localisation and SLAM

There is a significant history of research in visual localisation and SLAM on humanoids. Initially, this focused on feature-based methods and Extended Kalman Filters (EKF). Stasse *et al*. [6] adapted MonoSLAM [7], a monocular EKF-based SLAM algorithm, to exploit knowledge about the HRP-2 robot's motion from its pattern generator and inertial sensing to improve the robustness of pose tracking. The work was a notable early example demonstrating loop closure on a humanoid.

The fusion of a visual tracking/SLAM method with proprioception was also used in the work of Ahn *et al*. [8] who also integrated a visual odometry module.

Oriolo *et al*. [9], [10] instead implemented a complementary strategy to fuse pose corrections from their sparse visual SLAM system with their EKF-based kinematic-inertial state estimator. Demonstrations were carried out on the Nao robot. Kwak *et al*. [11] proposed a particle filter-based SLAM method using a stereo camera. They attempted to build a 3D grid map for localisation but noise in the stereo data required them to only record camera data from stationary positions. They also mentioned that corruptions were introduced into their reconstructions by areas of the environment with no texture.

A common characteristic of these works is that they used sparse representations. These are useful for localisation but cannot be interpreted visually or be used for path planning. We investigate the application of a direct semi-dense SLAM method and aim to achieve sufficient robustness during the walking and turning motions of the robot.

C. Direct Visual SLAM

In recent years, advances have been made in the field of dense visual SLAM, supported by the arrival of low-cost RGB-D cameras such as the Microsoft Kinect or Asus Xtion. As a result, various methods of direct SLAM have been developed.

KinectFusion [12] was a seminal contribution to dense RGB-D SLAM. It was the first method to implement real-time dense tracking and fusion of depth data. Whelan *et al*. [13] extended this approach to large-scale environments. Kerl *et al*. [14] [15] improved pose estimation using a robust cost function during image alignment. ElasticFusion [1] implements deformation-based loop closures but avoid using a traditional pose-graph by instead performing relaxation on the surfaces mapped.

Direct methods for SLAM using passive stereo and monocular cameras have also been developed, for example [16], [17]. These methods are semi-dense as accurate disparity cannot be computed for low-texture image areas.

Many of the above methods perform well in structure-rich environments if there is a smooth camera trajectory. As described in Section I, this is rarely the case for locomoting robots, and consequently, their application has been limited. An example is the work of Wagner *et al*. [18] which fused robot wheel odometry (*i.e.*not a bipedal robot) with a dense SLAM solution based on a pose-graph extension of KinectFusion. Their work combines the two modalities but as the robot's motion is planar and smooth it avoids the complexities of true humanoid SLAM.

In our previous work [19] we integrated a dense SLAM approach on the Atlas humanoid robot. It provided a dense reconstruction as input to the robot's footstep planning system. However, it did not support loop closure for locally loopy trajectories, which has motivated this work.

We chose ElasticFusion for the current work as it is designed to handle locally loopy trajectories which are common in typical humanoid manipulation scenairos. Frame-by-frame fusion of 3D data results in an up-to-date environment model which can be used for collision-free motion planning.

III. System Overview

Our robot contains a Carnegie Robotics Multisense SL global-shutter stereo camera installed in its head (Figure 2). The sensor provides 1024×1024 image pairs of colour and corresponding disparity at a rate of 15 Hz. The lenses have a field of view of 80°×80°. Disparity is computed by an implementation of Semi Global Matching [20] running on an FPGA on board the device.

The robot is described by a kinematic tree with sensors attached to different links. An illustration of these coordinate frames and their corresponding transforms can be seen in Figure 3.

We define a pose $\mathbf{T}$ as a transformation matrix restricted to the class of rigid body motions forming the special Euclidean group $\mathbb{SE}(3)$ composed of a rotation matrix $\mathbf{R} \in \mathbb{SO}(3)$ and a translation vector $\mathbf{t} \in \mathbb{R}^3$. We refer to ${}^{(est)}\mathbf{T}_{A_{t_i}\to B_{t_j}}$ as the transformation measured by the estimator est of frame B at time t_j relative to frame A at time t_i.

Our kinematic-inertial state estimator tracks the pose of the pelvis in the world frame, ${}^{(ki)}\mathbf{T}_{W\to P_t}$. The visual SLAM system tracks the pose of the camera in the world, ${}^{(vt)}\mathbf{T}_{W\to C_t}$ using consecutive pairs of images from the stereo camera.

The pelvis frame P and the camera frame C are connected through a non-rigid kinematic chain containing 3 back joints and 3 neck joints. Forward kinematics is used to relate measurements between the pelvis and camera frame at each time step t: ${}^{(fk)}\mathbf{T}_{P_t\to C_t}$.

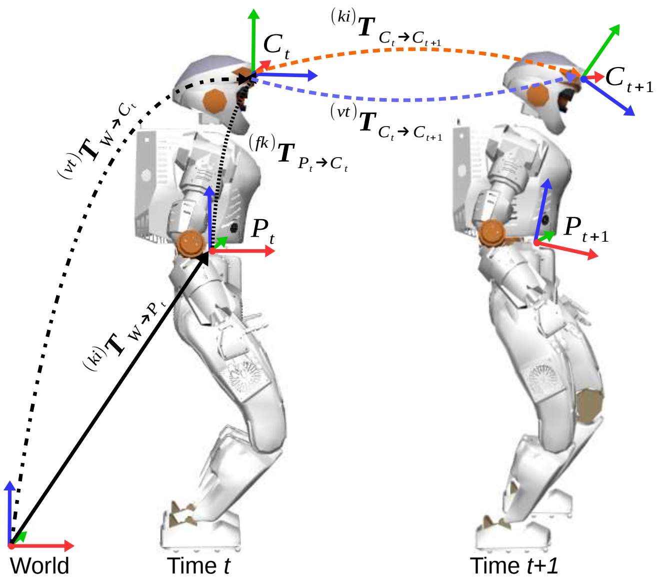

${}^{(ki)}\boldsymbol{T}_{C_t\to C_{t+1}}$
C_t
C_{t+1}
${}^{(vt)}\boldsymbol{T}_{C_t\to C_{t+1}}$
${}^{(fk)}\boldsymbol{T}_{P_t\to C_t}$
${}^{(vt)}\boldsymbol{T}_{W\to C_t}$
P_t
P_{t+1}
${}^{(ki)}\boldsymbol{T}_{W\to P_t}$
World
Time t
Time t+1

Fig. 3: The major co-ordinate frames of this SLAM system. These frames are connected via a time-varying kinematic tree.

Our system is based on the fusion of the kinematic-inertial state estimator and visual SLAM. We now describe these individual sources of information.

A. Kinematic-Inertial State Estimation

The kinematic-inertial state estimator uses sensor measurements from the 6 joints of each leg, force-torque sensors in each foot and an IMU rigidly attached to the robot's pelvis. Estimates of position, orientation and velocity are produced concurrently and incorporate constraints derived from the contact state of the feet. The estimate is computed at high frequency (250 Hz), low latency (2-3 msec) and remains aligned to gravity. It is the direct input to the low-level control system.

While our previous research developed a low drift EKF approach, [5], the estimator integrated with the Valkyrie control system, [4], performs similarly and is used here.

Motion prior computation: The estimator produces a running estimate of the pelvis pose ${}^{(ki)}\mathbf{T}_{W\to P_t}$. Using the estimate corresponding to the timestamps of consecutive images, t to $t+1$, and the pelvis-to-camera forward kinematics, the incremental motion of the camera can be computed as follows:

$$
{}^{(ki)}\mathbf{T}_{C_t\to C_{t+1}} = \left({}^{(ki)}\mathbf{T}_{W\to P_t}\,{}^{(fk)}\mathbf{T}_{P_t\to C_t}\right)^{-1} \left({}^{(ki)}\mathbf{T}_{W\to P_{t+1}}\,{}^{(fk)}\mathbf{T}_{P_{t+1}\to C_{t+1}}\right) \quad (1)
$$

One particular challenge when moving from the Boston Dynamics Atlas, used in the above works, to the Valkyrie is the quality of the gyroscope sensing. Atlas contains a Fibre Optic Gyroscope while Valkyrie relies on a MEMS Microstrain GX4-25 which required online gyro bias estimation to suppress orientation drift. With this in place, the approach produces a low drift dead-reckoning estimate. Its performance is evaluated in Section V.

B. Visual Tracking in ElasticFusion

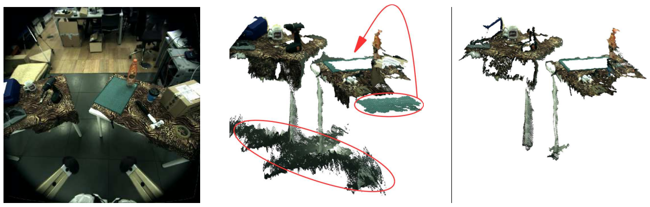
Transformation matrices are over-parametrised representations. For pose tracking optimisation the minimal representation $\boldsymbol{\xi} \in \mathbb{R}^6$ expressed in the associated Lie algebra $\mathfrak{se}(3)$ is used instead. Correspondences between $\mathbf{T} \in \mathbb{SE}(3)$ and $\boldsymbol{\xi} \in \mathfrak{se}(3)$ are computed through the matrix logarithm and exponential functions respectively [21].

Visual tracking in ElasticFusion is implemented as optimising a joint energy function. This function is composed of two terms which perform photometric (RGB) and geometric (ICP - Iterative Closest Point) frame-to-model alignment:

$$
E(\boldsymbol{\xi}) = wE_{rgb}(\boldsymbol{\xi}) + E_{icp}(\boldsymbol{\xi}) \quad (2)
$$

The weight w is empirically set to 0.1 reflecting the difference in units between the two error terms: metres as used in E_{icp} and pixel intensity values as used in E_{rgb}.

Our contribution modifies this energy term to incorporate information from the robot's proprioceptive sensors.



(a) Colour Image (b) Raw Stereo Point Cloud (c) Filtered Stereo Point Cloud

Fig. 4: The raw disparity images are filtered to remove unreliable data. (a) is the original colour image. (b) is the corresponding raw stereo point cloud - red circles highlight erroneous depth from areas of low texture, such as the floor and a green sheet reconstructed appart from its actual location (indicated with an arrow). (c) is the result of our filtering procedure.

IV. Robust Pose Tracking

The fusion of a kinematic-inertial state estimator with visual SLAM is desirable as the modalities are complementary: the former can handle degenerate cases where the vision system fails entirely, such as a lack of visual features or changes in illumination, while at the same time it provides information about global roll and pitch through the IMU. Through force-torque and joint encoder sensing, we can reliably know when the robot is stationary. Our goal is to bound the typical drift of the kinematic-inertial estimate through frame-to-model alignment and loop closures as performed by ElasticFusion.

A. *Disparity Pre-filtering*

ElasticFusion was developed for active RGB-D cameras and assumes a Gaussian error model associated with the depth data. This model is not suitable for stereo, where the error grows quadratically with depth. In order to mitigate this issue, we carry a pre-processing step on the stereo data. The procedure is computed in 6.5 msec per frame.

Disparity is not reliably computed for areas in the image with low texture. Therefore, we filter out data from these areas by computing for each pixel over a 5×5 window the gradient in the vertical, horizontal and diagonal directions. If these gradients are small, the pixel is considered to be an area of low texture and is dropped.

Our cameras are set-up in a horizontal configuration, which makes estimating the disparity of horizontal edges unreliable. We discard data originating from edges oriented at an angle of less than 10 degrees from horizontal.

Finally, we remove small unconnected groups of points which could be due to specular effects.

Figure 4 gives a qualitative impression of the effect of this filtering procedure.

B. *Proprioceptive ElasticFusion*

Given the cumulative rigid body motion as sensed through kinematic-inertial measurements between two consecutive image frames $\boldsymbol{\xi}_{ki} = \log({}^{(ki)}\mathbf{T}_{C_t \to C_{t+1}})$ taken from Equation 1, we define an additional residual $\mathbf{r}_{ki}$ which computes the error between the fused estimate $\boldsymbol{\xi}$ and the kinematic-inertial estimate $\boldsymbol{\xi}_{ki}$:

$$\mathbf{r}_{ki}(\boldsymbol{\xi}) = \log(\exp(\boldsymbol{\xi}_{ki})\exp(\boldsymbol{\xi})^{-1}) \tag{3}$$

With the corresponding energy term:

$$E_{ki} = \mathbf{r}_{ki}^\top \mathbf{r}_{ki} \tag{4}$$

In our system, this term is added to the global energy function in Equation 2 with a corresponding weight q:

$$E(\boldsymbol{\xi}) = wE_{rgb}(\boldsymbol{\xi}) + E_{icp}(\boldsymbol{\xi}) + qE_{ki}(\boldsymbol{\xi}) \tag{5}$$

The rigid body motion $\boldsymbol{\xi}$ is then solved by Gauss-Newton non-linear least squares minimisation using ElasticFusion's three-level coarse-to-fine pyramid scheme:

$$\begin{aligned}(w\mathbf{J}_{rgb}^\top \mathbf{J}_{rgb} + \mathbf{J}_{icp}^\top \mathbf{J}_{icp} + q\mathbf{J}_{ki}^\top \mathbf{J}_{ki})\hat{\boldsymbol{\xi}} =& \\ -(w\mathbf{J}_{rgb}^\top \mathbf{r}_{rgb} + \mathbf{J}_{icp}^\top \mathbf{r}_{icp} + q\mathbf{J}_{ki}^\top \mathbf{r}_{ki})&\end{aligned} \tag{6}$$

Where $\hat{\boldsymbol{\xi}}$ is the increment computed at each iteration which is used to update the pose:

$$\boldsymbol{\xi} = \log(\exp(\boldsymbol{\xi})\exp(\hat{\boldsymbol{\xi}})) \tag{7}$$

After the optimisation has converged, we update the global pose of the camera:

$$\mathbf{T}_{W \to C_{t+1}} = \mathbf{T}_{W \to C_t}\exp(\boldsymbol{\xi}) \tag{8}$$

Weighting Terms in the Tracking Cost Function: A particular issue is choosing how to balance the numerical contribution of each error term within the tracking cost function (Equation 5).

The kinematic-inertial term provides a single constraint between the previous pose estimate and the kinematic-inertial pose (Equation 3).

However, the ICP and RGB alignment procedures impose one constraint per pair of matched 3D points/pixels. This results in an imbalanced number of constraints, and, if not considered, the kinematic-inertial term would have inconsequential influence.

We implement a simple heuristic for scaling the contribution of the kinematic-inertial term to have a sufficient influence on the combined motion estimate. Given the proportion of inliers for the ICP and RGB alignment procedures relative to the size of the point cloud (*i.e.* the percentages ICP_p, RGB_p), we define a corresponding proportional term for the kinematic-inertial measurement:

1) In a degenerate situation where the proportion of inliers for both ICP and RGB alignment procedures is low ($ICP_p, RGB_p < 5\%$), we trust the kinematic-inertial estimate fully ($KI_p = 100\%$).
2) We observed that in well structured environments, the kinematic-inertial term should contribute slightly more than one third to the total pose error minimisation. Therefore, we set $KI_p = max(ICP_p, RGB_p) + \alpha$, where $\alpha = 10\%$ was a suitable value in our evaluation.

The contributions from the ICP, RGB and kinematic-inertial components are evenly balanced within a well structured scene. However, for sequences with almost no overlap between consecutive frames the kinematic-inertial component dominates. The resulting weight q is computed as:

$$q = \frac{KI_p}{100} \times points \tag{9}$$

Although not addressed here, an additional strategy could be formulated to handle failures in the state estimator. For example, foot slippage could be detected from unexpected spikes in velocity and the influence of the kinematic-inertial term can be reduced during these sequences.

Finally, this approach is implicitly robust to dynamic elements in the scene. This is achieved by exploiting the data fusion strategy of ElasticFusion, where continuously dynamic points are assigned a low confidence and do not become part of the map.

V. Experimental Results

In this section we will present an evaluation of our method with a series of experiments on the Valkyrie humanoid robot. The test environment consists of a manipulation scene containing several tables with objects on them surrounding the robot. Our laboratory is equipped with a Vicon motion capture system which provides ground truth trajectory measurements.

The dataset used as part of this evaluation is described in Table I. Log1 is a short walking sequence of a single loop trajectory within a static feature-rich environment *i.e* the ideal operating scenario. During this log we also took care not to perform any fast motions of the neck. Log2 is a longer and locally loopy trajectory containing several visual challenges.

We analyse 4 aspects:

A. We assess the tracking performance of the proposed method against the original ElasticFusion system and the robot's kinematic-inertial state estimator for two different experiments.
B. We then demonstrate that our approach overcomes the typical limitations of visual tracking during challenging situations.
C. We perform an evaluation of the accuracy of the stereo reconstruction against LIDAR point clouds as produced by the Hokuyo UTM-30LX-EW spinning planar LIDAR contained within the MultiSense SL.
D. Finally, we demonstrate the integration of our algorithm within the closed-loop walking controller of the Valkyrie humanoid robot.

In our evaluation we refer to three different estimators:

- EF: ElasticFusion running on stereo data (12 Hz).
- KI: The open-loop kinematic-inertial state estimator which is also used in the control loop (250 Hz).
- PEF: Proprioceptive ElasticFusion - our proposed system which fuses the visual and kinematic-inertial systems (12 Hz).

We evaluate the performance by comparing trajectories against ground truth measurements using the metrics proposed by Sturm *et al.* [22]:

a) Absolute Trajectory Error: We compute the absolute error between two trajectories that are aligned in the least-squares sense. At time t, the error between corresponding poses is:

$$\text{ATE}_t = {}^{(gt)}\mathbf{T}^{-1}_{W \to C_t} \, {}^{(est)}\mathbf{T}_{W \to C_t} \tag{10}$$

b) Relative Pose Error: To measure drift between two corresponding trajectories, we compute the relative pose error over a time interval Δ at each timestep t:

$$\text{RPE}_t = {}^{(gt)}\mathbf{T}^{-1}_{C_t \to C_t + \Delta} \, {}^{(est)}\mathbf{T}_{C_t \to C_t + \Delta} \tag{11}$$

c) Drift per Distance Travelled: We divide the relative pose error by the length of the path travelled to obtain the drift per metre travelled:

$$\text{DDT}_t = \frac{\text{RPE}_t}{\sum_{k=t}^{t+\Delta} ||{}^{(gt)}\mathbf{t}_{C_k \to C_{k+1}}||} \tag{12}$$

A typical robot walking gait involves oscillatory motion. As a result, it is difficult to estimate the total distance travelled. We take the simple approach of integrating the length of path travelled by the camera for each time sample. This can overstate what one thinks of as 'distance travelled'.

We compute these metrics over all timestamps and the calculate the root mean square (RMS) for each trajectory.

A. Evaluation of Individual Systems

We first explored the performance of ElasticFusion on stereo data pre-processed as described in Section IV-A. The robot was commanded to walk clockwise up to the point of completing a loop in Log1.

In Figure 5, one can see the typical walking gait of the Valkyrie which involves oscillatory motion as the robot switches support between its left and right feet. Despite vibrations due to foot impacts and some sharp rotations, the motion estimated closely matches the Vicon trajectory — in large part because the environment contained structure in all directions. This indicates that stereo-only ElasticFusion can achieve acceptable tracking performance in feature rich environments assuming smooth camera motion.

Quantitative tracking performance for each of the estimators is presented in Table II for Log1 and for the more challenging Log2.

As ElasticFusion uses drift-free frame-to-model tracking, we evaluate the drift by limiting the size of its local tracking model to the previous 200 frames only.

For the kinematic-inertial state estimator, the main directions of drift are in the linear Z-axis and yaw rotation due to estimator unobservability in those directions. While roll and pitch are globally observable through the accelerometer, yaw is computed by integrating the rate gyroscope estimates and it drifts over time.

PEF achieves the best performance for each portion of the state with an average translational drift of 0.54 cm/m for Log2. The baseline ElasticFusion system is unable to operate in this environment and fails.

Dataset Description								Trans. ATE RMSE (m)	
Log	Dist. (m)	Steps	Time (sec)	Lack of Features	Lights Off	Motion Blur	Continuous Dynamics	EF	PEF
Log1	18.5	34	485	✗	✗	✗	✗	0.048	**0.020**
Log2	62.96	102	1204	✓	✓	✓	✓	FAIL	**0.025**

TABLE I: Description of the dataset used in this evaluation. ✓/✗ indicates the presence/absence of a certain challenge. ElasticFusion (EF) fails on Log2 because of these challenges.

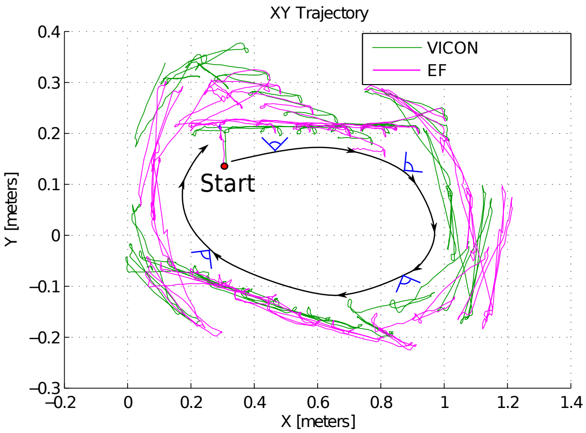


Fig. 5: Overhead view of the camera trajectory as estimated by our ground truth Vicon system (green) and ElasticFusion using stereo data only (magenta) for Log1. The direction of motion is indicated by the black line and the blue frames indicate the point of view of the robot (facing outside).

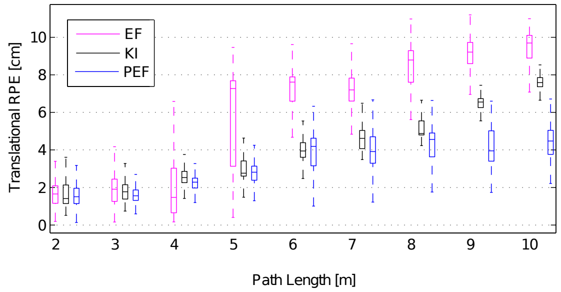


Fig. 6: Translational RPE (cm) for increasing path lengths (m) for Log1, showing PEF achieving the smallest drift rate.

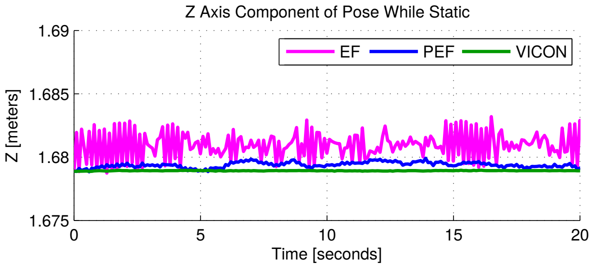


Fig. 7: Z-component of the robot's pose for a sequence in Log1 when it is stationary: fusion of kinematic-inertial within visual tracking (PEF) results in more stable pose than using only vision (EF).

	Log1			Log2		
DDT	EF	KI	PEF	EF	KI	PEF
XYZ (cm/m)	1.06	0.72	**0.61**	18.55	0.78	**0.54**
XY (cm/m)	1.00	0.54	**0.53**	17.06	0.56	**0.45**
Z (cm/m)	0.35	0.48	**0.30**	7.28	0.54	**0.29**
Yaw (deg/m)	0.33	0.53	**0.16**	6.51	0.38	**0.19**

TABLE II: Drift per distance travelled averaged over 2 m to 10 m trajectory intervals. The proposed fusion approach, PEF, outperforms the individual sub-systems.

A more detailed view of the rate of translational drift as a function of the path length for Log1 is shown in Figure 6. In this case, the relative pose error for PEF grows at the slowest rate.

Another benefit of performing vision and kinematic-inertial fusion is an increase in stability of the PEF estimate when compared to EF. Figure 7 shows that when tracking against the model, the EF estimate contains high frequency jitter because of its repeated geometric/photometric optimisations, which is common in visual tracking systems.

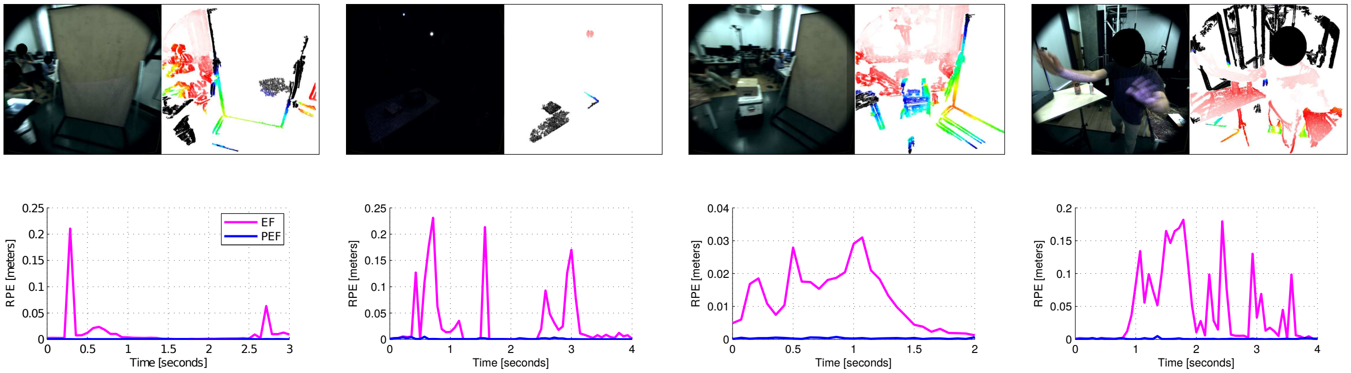
B. *Evaluation in Challenging Settings*

Following on from the previous section, here we focus on the particularly challenging parts of Log2 which cause the baseline EF system to fail.

We tested the proposed fusion method with the following specific challenges:

- Lack of features in the camera view: the camera points at a blank wall while the robot turns (∼ 20 sec per sequence). For PEF, the lack of suitable depth causes the kinematic-inertial tracking to dominate Equation 5.
- Changes in illumination: by turning the room's lighting on and off (∼ 15 sec per sequence). Behaviour is similar to previous case.
- Motion blur: by performing fast head motion (∼ 5 sec per sequence). While depth can be estimated in this case, the set of inliers is much smaller, meaning again kinematic-inertial tracking is preferred.
- Continuous dynamics in the scene: by introducing moving objects and people covering more than 50% of the field of view of the camera (∼ 5 sec per sequence). Depth measurements to moving objects are present, but the data fusion strategy of ElasticFusion integrates several frames before inserting the dynamic objects within the map.

The absolute trajectory error is shown in Table I. Examples of the successful operation of the PEF algorithm while the challenges mentioned above are occurring are shown in Figure 8. In each case the baseline EF system fails.



(a) Featureless Scene (b) Turning off Lights (c) Motion Blur (d) Dynamics in the Scene

Fig. 8: Top: Examples of colour-disparity image pairs during challenging sequences. Bottom: Corresponding frame-to-frame translational RPE for PEF and the baseline EF.

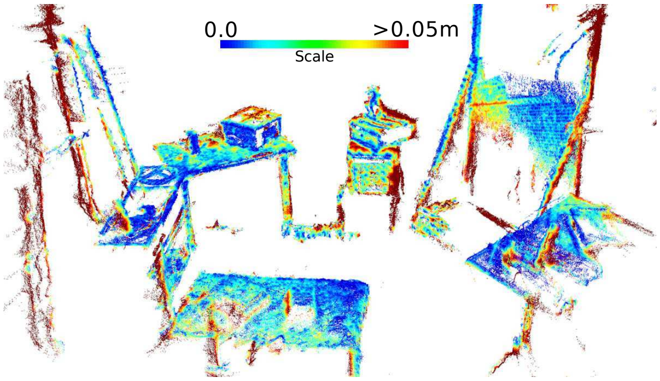


Fig. 9: Heat map of stereo per-point error.

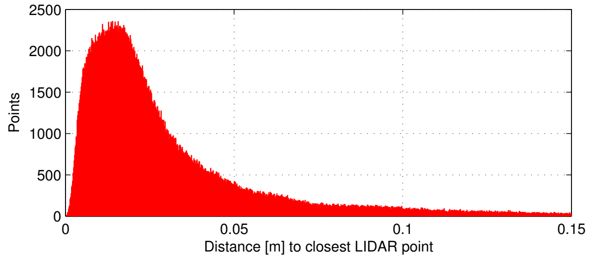


Fig. 10: Distribution of stereo points error.

C. Evaluation of 3D Reconstruction

In Figure 1 (bottom) we present a 3D model showing the reconstruction obtained during Log2. We evaluate its accuracy against a model created using the spinning Hokuyo LIDAR sensor contained within the MultiSense SL. The accuracy of the LIDAR sensor is +/-10 mm within the range of 0.1 m - 10 m, which makes it appropriate for coarsely evaluating the stereo reconstruction. We manually align several LIDAR point clouds from Log2 to create a 3D model of the test environment. Due to imperfections in how the LIDAR map is produced, we point out that these results are indicative of reconstruction quality rather than fully quantitative.

Visual comparison between the two reconstructions is shown in more detail in the attached video.

For each point in the visual model, we compute the distance to the closest LIDAR point as a per-point error. A heat-map of this error is shown in Figure 9. The scale and structure of the stereo model can be seen to closely match the LIDAR model. Of particular note is that the stereo model is aligned with gravity (by design) which is essential for it to be used in practical applications.

Figure 10 shows the distribution of per-point errors, with a median error value of 0.02 m. We conclude the reconstruction is of sufficient accuracy for tasks such as collision free motion planning.

D. Closed Loop Integration

In our final experiment we demonstrate the integration of PEF within the closed loop walking controller of the Valkyrie robot. This experiment can be seen in the accompanying video.

The environment used for this experiment is depicted in Figure 1 (top). It consists of two tables with objects and corresponding white goal positions on the floor. The robot walks to each table in turn to reach these goal positions.

The kinematic-inertial state estimator (KI) represents the direct input to the walking control system of the robot. In order to prevent this state estimator from drifting, we transmit pose corrections on a regular interval from our method (PEF) which enables the robot to successfully reach the goal targets repeatedly.

VI. Conclusions and Future Work

In this paper we investigated the challenges of implementing direct visual SLAM on a humanoid robot. These include motion blur in the image, lack of visual features in the scene, change in illumination and fast motion resulting in dramatic view change. They typically effect the pose tracking components of visual SLAM systems, causing them to fail and in turn leading to corrupted reconstructions of the scene.

In order to handle these challenges, we extended the direct visual SLAM method ElasticFusion to integrate information from our high-rate low-drift kinematic-inertial state estimator. We use the state estimator to provide a camera motion prior which is integrated within the pose tracking component of ElasticFusion to handle for the described degenerate cases. As many previous approaches made use of sparse point-based SLAM methods, our direct approach can produce a semi-dense reconstruction which can also be interpreted visually and used for tasks such as collision free motion planning.

We evaluated our approach through a series of experiments in our laboratory. Our fusion method achieves lower drift rates than the tracking of the kinematic-inertial state estimator and ElasticFusion's visual tracking individually but more importantly it is robust to sequences containing the aforementioned visual challenges. We provided a qualitative evaluation of our stereo-produced reconstruction against LIDAR and described an online integration experiment of our method within the walking controller of Valkyrie.

Currently, our method is implicitly robust to dynamics in the scene by exploiting the fact that dynamic objects are assigned low confidence and do not become a part of the model as long as these are continuously moving. In the future, we are interested in making use of our motion prior to actively detect dynamic objects in the scene and to segment them out.

Another observation is that the size of the reconstruction continues to grow in time even as we continuously explore a single static scene. This is due to noisy sensor readings which result in redundant surfels being added to the map. This represents another aspect we are interested in improving.

Acknowledgements

This research is supported by the Engineering and Physical Sciences Research Council (EPSRC), as part of the CDT in Robotics and Autonomous Systems at Heriot-Watt University and The University of Edinburgh. We would like to acknowledge the University of Edinburgh Humanoid Team as well as our collaborators at NASA and IHMC.

References

[1] T. Whelan, S. Leutenegger, R. F. Salas-Moreno, B. Glocker, and A. J. Davison, "ElasticFusion: Dense SLAM without a pose graph," *Robotics: Science and Systems (RSS)*, 2015.

[2] X. Xinjilefu, S. Feng, and C. Atkeson, "Center of mass estimator for humanoids and its application in modelling error compensation, fall detection and prevention," in *IEEE/RSJ Int. Conf. on Humanoid Robots*, Seoul, Korea, November 2015.

[3] 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.

[4] T. Koolen, S. Bertrand, G. Thomas, T. de Boer, T. Wu, J. Smith, J. Englsberger, and J. Pratt, "Design of a momentum-based control framework and application to the humanoid robot Atlas," *Intl. J. of Humanoid Robotics*, vol. 13, 2016.

[5] M. F. Fallon, M. Antone, N. Roy, and S. Teller, "Drift-free humanoid state estimation fusing kinematic, inertial and LIDAR sensing," in *IEEE/RSJ Int. Conf. on Humanoid Robots*, Madrid, Spain, November 2014.

[6] O. Stasse, D. Andrew J, R. Sellaouti, and K. Yokoi, "Real-time 3D SLAM for humanoid robot considering pattern generator information," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2006, pp. 348–355.

[7] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "MonoSLAM: Real-time single camera SLAM," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 29, no. 6, pp. 1052–1067, 2007.

[8] S. Ahn, S. Yoon, S. Hyung, N. Kwak, and K. S. Roh, "On-board odometry estimation for 3D vision-based SLAM of humanoid robot," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2012, pp. 4006–4012.

[9] G. Oriolo, A. Paolillo, L. Rosa, and M. Vendittelli, "Vision-based odometric localization for humanoids using a kinematic EKF," in *IEEE/RSJ Int. Conf. on Humanoid Robots*, 2012, pp. 153–158.

[10] ——, "Humanoid odometric localization integrating kinematic, inertial and visual information," *Autonomous Robots*, vol. 40, no. 5, pp. 867–879, 2016.

[11] N. Kwak, O. Stasse, T. Foissotte, and K. Yokoi, "3D grid and particle based SLAM for a humanoid robot," in *IEEE/RSJ Int. Conf. on Humanoid Robots*, 2009, pp. 62–67.

[12] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "KinectFusion: Real-time dense surface mapping and tracking," in *IEEE/ACM Intl. Sym. on Mixed and Augmented Reality (ISMAR)*, 2011, pp. 127–136.

[13] T. Whelan, J. McDonald, M. Kaess, M. Fallon, H. Johannsson, and J. Leonard, "Kintinuous: Spatially extended KinectFusion," in *RSS Workshop on RGB-D: Advanced Reasoning with Depth Cameras*, Sydney, Australia, Jul 2012.

[14] C. Kerl, J. Sturm, and D. Cremers, "Robust odometry estimation for RGB-D cameras," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2013, pp. 3748–3754.

[15] ——, "Dense visual SLAM for RGB-D cameras," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2013, pp. 2100–2106.

[16] J. Engel, J. Sturm, and D. Cremers, "Semi-dense visual odometry for a monocular camera," in *Intl. Conf. on Computer Vision (ICCV)*, 2013, pp. 1449–1456.

[17] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," in *Eur. Conf. on Computer Vision (ECCV)*, 2014, pp. 834–849.

[18] R. Wagner, U. Frese, and B. Bäuml, "Graph SLAM with signed distance function maps on a humanoid robot," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2014, pp. 2691–2698.

[19] M. F. Fallon, P. Marion, R. Deits, T. Whelan, M. Antone, J. McDonald, and R. Tedrake, "Continuous humanoid locomotion over uneven terrain using stereo fusion," in *IEEE/RSJ Int. Conf. on Humanoid Robots*, Seoul, Korea, November 2015.

[20] H. Hirschmüller, "Stereo processing by semi-global matching and mutual information," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 30, no. 2, pp. 328–341, Feb 2008.

[21] J.-L. Blanco, "A tutorial on SE(3) transformation parameterizations and on-manifold optimization," *University of Malaga, Tech. Rep*, vol. 3, 2010.

[22] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2012, pp. 573–580.