2013 IEEE/RSJ International Conference on
Intelligent Robots and Systems (IROS)
November 3-7, 2013. Tokyo, Japan

RANSAC for Motion-Distorted 3D Visual Sensors

Sean Anderson and Timothy D. Barfoot

***Abstract*— Visual odometry (VO) is a highly efficient and powerful 6D motion estimation technique; state-of-the-art bundle adjustment algorithms now optimize over several frames of temporally tracked, appearance-based features in real time. It is well known that the temporal feature correspondence process is highly prone to mismatches. The standard technique used for outlier rejection in this process is *random sample consensus* (RANSAC), which is an iterative and non-deterministic process used to find the parameters of a mathematical model that best describe a likely set of inliers. The traditional model used for RANSAC in the visual odometry pipeline is a rigid transformation between two camera poses; this model has long assumed the use of an imaging sensor with a global shutter. In order to use imaging sensors that do not operate with a global shutter, it is proposed that the RANSAC algorithm be modified to use a constant-camera-velocity model. Specifically, this paper investigates the use of a two-axis scanning lidar in the visual-odometry pipeline. Images are formed using lidar intensity data, and due to the scanning-while-moving nature of the lidar, the behaviour of the sensor resembles that of a slow rolling-shutter camera. We formulate a Motion-Compensated RANSAC algorithm that uses a constant-velocity model and the individual timestamp of each extracted feature. The algorithm is validated using 6880 lidar frames with a resolution of 480 $\times$ 360, captured at 2Hz, over a 1.1km traversal. Our results show that the new algorithm results in far more inlying feature tracks for rolling-shutter-type images and ultimately higher-accuracy VO results.**

I. Introduction

Visual odometry (VO) is a powerful 6D motion estimation technique that is made efficient by temporally tracking points of interest, or *appearance-based features*, over a sequence of images. The technique is widely used due to the low cost and availability of passive camera technology. Although cameras using Complementary Metal-Oxide-Semiconductor (CMOS) technology are known to be cheaper than ones using a Charge-Coupled Device (CCD), they do not often employ a global shutter and are often avoided in robotics due to the added complexity of rolling-shutter-type distortions.

Our motivation in exploring the use of rolling-shutter-type imagery comes from the desire to use range/intensity data from a two-axis scanning lidar that is scanning while moving. In particular, we are interested in using the Autonosys LVC0702 [1], seen in Figure 1, which can generate 480×360 intensity imagery at a rate of 2Hz. Due to the slow vertical scan of the sensor, scanning while moving causes nonaffine image distortion (see Figure 2). The benefit in using intensity imagery, over passive camera imagery, is that it is an active measurement scheme and does not rely on external lighting conditions. Initial work with the sensor by McManus et al. [2] shows that Speeded-Up Robust Features (SURF) [3] can be both extracted from the intensity imagery and robustly matched over a full diurnal cycle.

The authors are members of the Autonomous Space Robotics Lab at the Institute for Aerospace Studies, University of Toronto, 4925 Dufferin Street, Toronto, Ontario, Canada. sean.anderson@mail.utoronto.ca, tim.barfoot@utoronto.ca

Fig. 1. This figure shows the ROC6 rover, equipped with an Autonosys LVC0702 lidar and a Thales DG-16 Differential GPS unit.

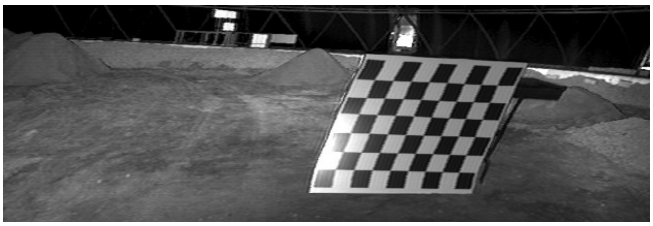

Fig. 2. This figure shows an example intensity image captured using the Autonosys LVC0702 two-axis scanning lidar. The nonaffine image distortion is caused by changing the viewpoint of the lidar during image acquisition.

Before considering a motion estimation problem that properly accounts for this type of data, we must first confront the issue of *feature correspondence* and *outlier rejection*. An initial guess of the temporal tracks between two sets of features is often generated using appearance-based feature descriptors; this alone does not take into account the geometry of the problem and is prone to generating mismatches. The standard outlier-rejection scheme for filtering the initial guess is random sample consensus (RANSAC) [4], which is a nondeterministic algorithm for robustly finding the parameters of a mathematical model that best describe a likely set of inliers. The model used in the RANSAC algorithm for the global-shutter, visual pipeline is a single rigid transformation (i.e., translation and rotation). In order to account for the temporal nature of rolling-shutter-type images, a RANSAC algorithm using a 6D constant-velocity model is proposed.

978-1-4673-6357-0/13/$31.00 ©2013 IEEE

The rest of this paper is as follows: in Section II we will provide a review of related work, in Section III we will formulate the problem and derive both a nonlinear and Euclidean RANSAC algorithm using a constant-velocity model, in Section IV the outlier-rejection scheme is validated experimentally using a 6880 frame sequence of lidar intensity imagery, and in Section V we provide final comments on the algorithm.

II. Related Work

The RANSAC algorithm [4] has been highly successful as an outlier-rejection scheme in the visual pipeline and has become popularized due to its speed and efficiency. Using global shutter cameras, both the monocular 5-point algorithm [5] and stereo-pair, 3-point algorithm [6] are widely used, most notably on the Mars Exploration Rovers [7]. Although the use of CMOS technology is desirable due to low cost and widespread availability, the addition of a temporal parameter to the standard camera model causes nontrivial and unhandled complexities in the mature visual pipeline. For this reason, the available literature concerning the use of rolling-shutter-type imagery is sparse in comparison to that of global-shutter imagery.

Although much of the existing rolling-shutter literature makes use of special cases, there are still a few pieces of work that properly account for the 6D sensor motion. However, many of these methods stray from using RANSAC as an outlier-rejection scheme. Early work by Ait-Aider et al. [8] derives a nonlinear, least-squares estimator for the velocity of a monocular rolling-shutter camera over a single frame of data; in order to find this velocity over only a single image, a known geometric target is used, and the feature correspondence is supervised. In a separate work, Ait-Aider et al. [9] also derive the batch nonlinear optimization technique for a stereo rig that includes only one rolling-shutter camera, and one global-shutter camera. Again, this experiment uses a target with known geometry and easy-to-find feature correspondences. The possibility of using RANSAC is mentioned, but left as a future extension.

Jia and Evans [10] apply the extended Kalman filter (EKF) to a monocular rolling-shutter camera and use predictive measurements from a gyroscope and accelerometer. In a similar fashion to RANSAC, this algorithm handles outlier rejection by applying the EKF correction step for multiple feature correspondence hypotheses and checking to see which update generated the most likely set of inliers. Akin to the classic RANSAC algorithm, our proposed method does not require additional sensors, such as a gyroscope, to find feature correspondences.

The most prominent visual pipeline using a rolling-shutter camera is the full *bundle adjustment* algorithm formulated by Hedborg et al. [11], which uses a form of pose interpolation between nominal image times. The primary outlier-rejection scheme used in this work is a *cross-check* method, which makes no use of camera geometry and relies on the use of a feature tracker, such as Kanade-Lucas-Tomasi (KLT) [12]. After an initial bundle-adjustment step, a second stage of outlier rejection is applied by checking point triangulations from multiple viewpoints. In an earlier work, Hedborg et al. [13] note that the use of a *cross check* with KLT is ideal for sensors with a very high framerate. In contrast to the 30Hz camera used by Hedborg et al. [13], the proposed research aims to find feature correspondences between rolling-shutter-type images taken at 2Hz. Furthermore, Hedborg et al. [13] take advantage of a secondary rejection step that is tightly coupled with the estimation phase of the visual pipeline.

Using an identical two-axis, scanning lidar to this research, Dong et al. [14] perform a similar style of pose interpolation to the one seen by Hedborg et al. [11]. In this pipeline, the typical 3-point RANSAC algorithm is applied with a very loose threshold that allows for some outliers. In the estimation phase, a robust M-estimation scheme is then used to try to minimize the effect of incorrectly matched features. By using a constant-velocity model in the RANSAC algorithm, the proposed method aims to remove all outlier rejection burden from the estimation phase.

III. Methodology

In this section, we derive the Motion-Compensated RANSAC algorithm using a constant-velocity model. We show how our algorithm differs from the traditional rigid one and how using a velocity model impacts the algorithm. Furthermore, we propose two approximations to increase the computational efficiency of our algorithm.

A. Problem Formulation

To begin our derivation, we start by defining our inertial reference frame, $\boldsymbol{\mathcal{F}}_i$, and the instantaneous sensor frame, $\boldsymbol{\mathcal{F}}_s(t)$. The notation, $\mathbf{v}_c^{b,a}$, refers to a vector from a to b, expressed in frame c. For the feature correspondence problem, we define two sets of corresponding measurements, $\mathbf{y}_{m,1}$ and $\mathbf{y}_{m,2}$, where $m = 1 \dots M$. Each measurement pair, m, is extracted from sequential images 1 and 2 at times, $t_{m,1}$ and $t_{m,2}$, with a temporal difference of $\Delta t_m := t_{m,2} - t_{m,1}$. The sensor models for these measurements are

$$\mathbf{y}_{m,1} := \mathbf{f}(\mathbf{T}_s(t_{m,1})\mathbf{p}_m) + \mathbf{n}_{m,1}, \tag{1a}$$

$$\mathbf{y}_{m,2} := \mathbf{f}(\mathbf{T}_s(t_{m,2})\mathbf{p}_m) + \mathbf{n}_{m,2}, \tag{1b}$$

where $\mathbf{f}(\cdot)$ is the nonlinear camera model, $\mathbf{T}_s(t)$ is the 4×4 homogeneous transform matrix that specifies the pose of the sensor frame, $\boldsymbol{\mathcal{F}}_s(t)$, with respect to the inertial frame, $\boldsymbol{\mathcal{F}}_i$, at time t, and the measurement noises, $\mathbf{n}_{m,1} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_{m,1})$ and $\mathbf{n}_{m,2} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_{m,2})$, are assumed to be normally distributed with covariances $\mathbf{R}_{m,1}$ and $\mathbf{R}_{m,2}$. Note that each measurement pairing, m, is simply the hypothesis of a common landmark, l, typically based on the similarity of appearance-based feature descriptors, and may not actually be projected from the same 3D location, $\mathbf{p}_m := \begin{bmatrix} (\boldsymbol{\rho}_i^{l,i})^T & 1 \end{bmatrix}^T$. The goal is to determine the subset of all M measurement pairs, $\mathbf{y}_{m,1}$ and $\mathbf{y}_{m,2}$, that make up the most likely set of inliers.

The structure of the RANSAC algorithm consists of only a few main steps. It begins by selecting N random subsets of S measurement pairs, where S is the minimum number of

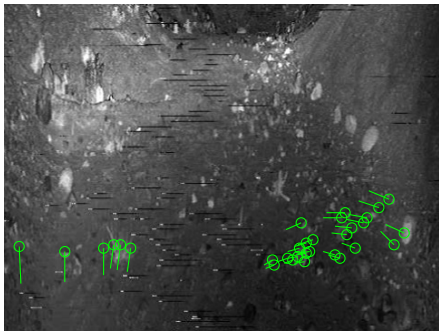

Fig. 3. This figure shows the inlying feature tracks after applying the classic RANSAC algorithm to our lidar intensity/range data. Due to fast motion and a slow vertical scan, only a small temporal band of features are matched.

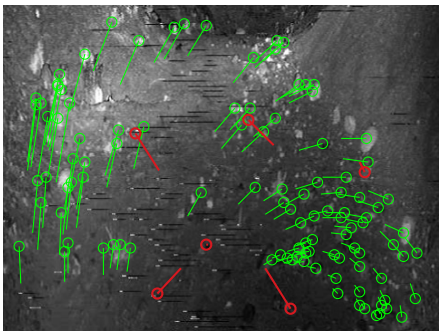

Fig. 4. This figure shows the how loosening the threshold on the tolerable measurement error, for the rigid RANSAC filter, allows for a larger number of inliers (green), but also introduces the possibility of outliers (red).

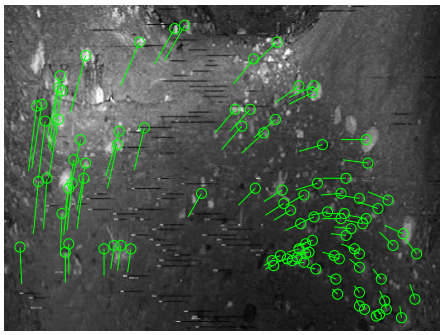

Fig. 5. This figure shows the inlying matches after applying the Motion-Compensated RANSAC algorithm; by using a constant velocity to model the motion of the sensor, the filter is able to more properly account for the distortion of the image.

pairs needed to solve the mathematical model. When using 3D point correspondences, such as in stereo camera or lidar data, S is typically 3. The number of required iterations, N, can be chosen using the equation

$$N = \frac{\ln(1 - p_{\text{succ}})}{\ln(1 - p_{\text{in}}^S)}, \tag{2}$$

where p_{succ} is the probability of choosing S measurement pairs that are all inliers and p_{in} is the probability of a single pair, m, being an inlier.

The first step, for each of the N subsets, is to solve the mathematical model that can be used to find the unknown pose change of the sensor frame, $\mathbf{T}_m$, between times $t_{m,1}$ and $t_{m,2}$,

$$\mathbf{T}_m := \mathbf{T}_s(t_{m,2})\mathbf{T}_s(t_{m,1})^{-1}. \tag{3}$$

The second step is to then apply the calculated transform, $\mathbf{T}_m$, to the measurements $\mathbf{y}_{m,1}$, using

$$\hat{\mathbf{p}}_{m,2} := \mathbf{T}_m \mathbf{p}_{m,1}, \quad \mathbf{p}_{m,1} := \mathbf{f}^{-1}(\mathbf{y}_{m,1}), \tag{4}$$

where the calculation of $\mathbf{p}_{m,1}$ is independent of the mathematical model and is performed only once for all M pairs. Note we take advantage of the invertible camera model available to 3D sensors. This assumption prevents us from directly applying our method to a monocular rolling-shutter camera; however, with some special care we believe it is possible to extend our technique for the monocular case. The third step is to then reproject the transformed measurements, $\hat{\mathbf{p}}_{m,2}$, back into measurement space:

$$\hat{\mathbf{y}}_{m,2} := \mathbf{f}(\hat{\mathbf{p}}_{m,2}) \tag{5}$$

Finally, each model is evaluated by finding the number of measurements pairs, $\mathbf{y}_{m,1}$ and $\mathbf{y}_{m,2}$, that satisfy

$$|\mathbf{y}_{m,2} - \hat{\mathbf{y}}_{m,2}| < \alpha \tag{6}$$

where α is a threshold in measurement space. For simplicity, we accept the model with the highest number of inliers.

B. Classic Rigid RANSAC

The mathematical model used in the classic RANSAC algorithm is a single rigid pose change between two static frames. It does not take into consideration the possibility of individual measurement times, and in order to use it, we must approximate our motion-distorted measurements as occurring at nominal image times, $\bar{t}_1$ and $\bar{t}_2$. The approximation being made is that

$$\Delta t = \bar{t}_2 - \bar{t}_1 \tag{7}$$

The transform, $\mathbf{T}_m$, is therefore the same for all pairs, m, and can be solved with only 3 noncollinear point pairs, in closed form, using the algorithms presented in [15] or [16]. This assumption is reasonable for rolling-shutter-type images that have little to no distortion. However, under significant motion it quickly becomes clear that the rigid algorithm is unfit for identifying a good set of inliers.

Taking into consideration the expected measurement noise, and setting the threshold on reprojection error, α, appropriately, the rigid RANSAC algorithm is limited to finding inliers that have a similar temporal difference, Δt. In practice, we tuned α to find as many inliers as possible (over the whole trajectory), without including outliers at standstill. Due to the slow vertical scan of our sensor, it is only possible to match a temporal band of features at high vehicle speeds (see Figure 3). Loosening α to allow for more feature matches at high vehicle speeds also allows for the inclusion of outliers, as seen in Figure 4.

C. Motion-Compensated RANSAC

To compensate for motion during image capture, it is proposed that we can approximate the motion of the sensor as a constant velocity,

$$\boldsymbol{\varpi} := \begin{bmatrix} \boldsymbol{\nu} \\ \boldsymbol{\omega} \end{bmatrix}, \tag{8}$$

where $\boldsymbol{\nu}$ and $\boldsymbol{\omega}$ are the linear and angular components, expressed in the sensor frame.

In order to calculate $\hat{\mathbf{y}}_{m,2}$, we derive the relationship between the sensor velocity, $\boldsymbol{\varpi}$, and the transform $\mathbf{T}_m$. We

begin with our transformation matrix definition,

$$\mathbf{T} := \begin{bmatrix} \mathbf{C} & \mathbf{r} \\ \mathbf{0}^T & 1 \end{bmatrix}, \tag{9}$$

where $\mathbf{C}$ is a 3×3 rotation matrix and $\mathbf{r}$ is a 3×1 translational component.

Using the exponential map, a transformation matrix can be calculated with the following tools:

$$\mathbf{T} = e^{-\boldsymbol{\pi}^{\boxplus}} = \begin{bmatrix} \mathbf{1} - \boldsymbol{\phi}^{\times}\mathbf{S} & \mathbf{S}\boldsymbol{\rho} \\ \mathbf{0}^T & 1 \end{bmatrix}, \qquad \boldsymbol{\pi} := \begin{bmatrix} \boldsymbol{\rho} \\ \boldsymbol{\phi} \end{bmatrix}, \tag{10}$$

$$\mathbf{S} = \frac{\sin\phi}{\phi}\mathbf{1} + \left(1 - \frac{\sin\phi}{\phi}\right)\mathbf{a}\mathbf{a}^T - \frac{1-\cos\phi}{\phi}\mathbf{a}^{\times}, \tag{11}$$

where $\boldsymbol{\rho}$ is a translation, $\boldsymbol{\phi}$ uses the axis-angle parameterization, $\boldsymbol{\phi} := \phi\mathbf{a}$, such that ϕ is the angle and $\mathbf{a}$ is a unit-length axis vector, and $(\cdot)^{\times}$ is the skew-symmetric operator:

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}^{\times} := \begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix}.$$

Note that for small pose changes, the exponential map can be approximated using

$$e^{-\boldsymbol{\pi}^{\boxplus}} \approx \mathbf{1} - \boldsymbol{\pi}^{\boxplus}, \tag{12}$$

where $(\cdot)^{\boxplus}$ is the SE(3) operator,

$$\mathbf{w}^{\boxplus} = \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix}^{\boxplus} := \begin{bmatrix} \mathbf{v}^{\times} & -\mathbf{u} \\ \mathbf{0}^T & 0 \end{bmatrix}.$$

Drawing from previous work with this motion-distorted sensor scheme [17], the general kinematic relationship is

$$\dot{\mathbf{T}} = -\boldsymbol{\varpi}(t)^{\boxplus}\mathbf{T}. \tag{13}$$

For a constant-velocity assumption, $\boldsymbol{\varpi}$, we avoid integrating (13) and note the special relationship:

$$\boldsymbol{\pi} = \Delta t \cdot \boldsymbol{\varpi} \tag{14}$$

Using (14), the desired transform is simply

$$\mathbf{T}_m = e^{-\Delta t_m \cdot \boldsymbol{\varpi}^{\boxplus}}. \tag{15}$$

1) Nonlinear Least-Squares Estimator: For each of the N subsets, the first major RANSAC step is to solve for the optimal constant velocity, $\boldsymbol{\varpi}$, that minimizes the S paired measurements' reprojection error. To do this, we setup a typical nonlinear least-squares estimation scheme, similar to that of a *bundle adjustment* problem. The objective function we wish to minimize is simply

$$J(\boldsymbol{\varpi}) := \frac{1}{2}\sum_{m=1}^{M} \mathbf{e}_m^T \mathbf{R}_{m,2}^{-1} \mathbf{e}_m, \tag{16}$$

where the error term is defined as

$$\mathbf{e}_m := \mathbf{y}_{m,2} - \hat{\mathbf{y}}_{m,2} = \mathbf{y}_{m,2} - \mathbf{f}(\mathbf{T}_m \mathbf{p}_{m,1}). \tag{17}$$

Note that we avoid estimating the landmark positions by assuming perfect knowledge of $\mathbf{p}_{m,1}$ (i.e., no noise on the measurement $\mathbf{y}_{m,1}$).

In order to linearize the measurement error, $\mathbf{e}_m$, we begin by noting the composition of two nonlinearities; the nonlinear camera model, and the transform $\mathbf{T}_m$. Starting with the transformation nonlinearity, we define

$$\mathbf{g}_m(\boldsymbol{\varpi}) := \mathbf{T}_m \mathbf{p}_{m,1}. \tag{18}$$

Consider the perturbation to the velocity,

$$\mathbf{T}_m = e^{-\Delta t_m \cdot \boldsymbol{\varpi}^{\boxplus}} = e^{-\Delta t_m \cdot (\bar{\boldsymbol{\varpi}} + \delta\boldsymbol{\varpi})^{\boxplus}}, \tag{19}$$

where $\bar{\boldsymbol{\varpi}}$ is the nominal solution and $\delta\boldsymbol{\varpi}$ is the perturbation. Given that $\delta\boldsymbol{\varpi}$ is small, we have

$$\mathbf{T}_m \approx e^{-\Delta t_m \cdot (\bar{\boldsymbol{\mathcal{S}}}_m \delta\boldsymbol{\varpi})^{\boxplus}} e^{-\Delta t_m \cdot \bar{\boldsymbol{\varpi}}^{\boxplus}} \tag{20a}$$

$$= e^{-\Delta t_m \cdot (\bar{\boldsymbol{\mathcal{S}}}_m \delta\boldsymbol{\varpi})^{\boxplus}} \bar{\mathbf{T}}_m, \tag{20b}$$

where $\bar{\boldsymbol{\mathcal{S}}}_m := \boldsymbol{\mathcal{S}}(\Delta t_m \bar{\boldsymbol{\varpi}})$, using the SE(3) operator [18],

$$\boldsymbol{\mathcal{S}}(\boldsymbol{\pi}) := \int_0^1 \boldsymbol{\mathcal{T}}^{\alpha}\, d\alpha \equiv \sum_{n=0}^{\infty} (-1)^n \frac{1}{(n+1)!}\left(\boldsymbol{\pi}^{\boxtimes}\right)^n,$$

where $(\cdot)^{\boxtimes}$ is the SE(3) operator [19],

$$\mathbf{w}^{\boxtimes} = \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix}^{\boxtimes} := \begin{bmatrix} \mathbf{v}^{\times} & \mathbf{u}^{\times} \\ \mathbf{0} & \mathbf{v}^{\times} \end{bmatrix},$$

and $\boldsymbol{\mathcal{T}}$, is the SE(3) operator given by

$$\boldsymbol{\mathcal{T}}(\boldsymbol{\pi}) := \mathbf{T}^{\boxdot} = \begin{bmatrix} \mathbf{C} & \mathbf{r} \\ \mathbf{0}^T & 1 \end{bmatrix}^{\boxdot} := \begin{bmatrix} \mathbf{C} & -\mathbf{r}^{\times}\mathbf{C} \\ \mathbf{0} & \mathbf{C} \end{bmatrix} \equiv e^{-\boldsymbol{\pi}^{\boxtimes}}.$$

This can be simplified again using the small-pose approximation found in (12),

$$\mathbf{T}_m \approx (\mathbf{1} - \Delta t_m \cdot (\bar{\boldsymbol{\mathcal{S}}}_m \delta\boldsymbol{\varpi})^{\boxplus})\bar{\mathbf{T}}_m. \tag{21}$$

Applying this perturbation scheme to (18),

$$\mathbf{g}_m(\bar{\boldsymbol{\varpi}} + \delta\boldsymbol{\varpi})$$

$$\approx \left(\mathbf{1} - \Delta t_m (\bar{\boldsymbol{\mathcal{S}}}_m \delta\boldsymbol{\varpi})^{\boxplus}\right)\bar{\mathbf{T}}_m \mathbf{p}_{m,1} \tag{22a}$$

$$= \bar{\mathbf{T}}_m \mathbf{p}_{m,1} - \Delta t_m (\bar{\boldsymbol{\mathcal{S}}}_m \delta\boldsymbol{\varpi})^{\boxplus} \bar{\mathbf{T}}_m \mathbf{p}_{m,1}. \tag{22b}$$

Using the homogeneous-coordinate operator,

$$\mathbf{y}^{\boxminus} := \begin{bmatrix} \boldsymbol{\epsilon} \\ \eta \end{bmatrix}^{\boxminus} = \begin{bmatrix} \eta\mathbf{1} & \boldsymbol{\epsilon}^{\times} \\ \mathbf{0}^T & \mathbf{0}^T \end{bmatrix},$$

and the identities,

$$\mathbf{w}^{\boxplus}\mathbf{y} \equiv -\mathbf{y}^{\boxminus}\mathbf{w}, \quad (\mathbf{T}\mathbf{y})^{\boxminus} \equiv \mathbf{T}\,\mathbf{y}^{\boxminus}\boldsymbol{\mathcal{T}}^{-1},$$

it follows that,

$$\mathbf{g}_m(\bar{\boldsymbol{\varpi}} + \delta\boldsymbol{\varpi}) = \bar{\mathbf{g}}_m + \mathbf{G}_m \delta\boldsymbol{\varpi}, \tag{23}$$

correct to first order, where,

$$\bar{\mathbf{g}}_m := \bar{\mathbf{T}}_m \mathbf{p}_{m,1}, \tag{24}$$

$$\mathbf{G}_m := \Delta t_m \bar{\mathbf{T}}_m (\mathbf{p}_{m,1})^{\boxminus} \bar{\boldsymbol{\mathcal{T}}}_m^{-1} \bar{\boldsymbol{\mathcal{S}}}_m. \tag{25}$$

Returning to our measurement error term,

$$\mathbf{e}_m \approx \mathbf{y}_{m,2} - \mathbf{f}(\bar{\mathbf{g}}_m + \mathbf{G}_m \delta\boldsymbol{\varpi}) \tag{26a}$$

$$\approx \bar{\mathbf{e}}_m - \mathbf{H}_m \delta\boldsymbol{\varpi}, \tag{26b}$$

correct to first order, where

$$\bar{\mathbf{e}}_m := \mathbf{y}_{m,2} - \mathbf{f}(\bar{\mathbf{g}}_m), \tag{27}$$

$$\mathbf{H}_m := \mathbf{F}_m \mathbf{G}_m, \qquad \mathbf{F}_m := \left.\frac{\partial \mathbf{f}}{\partial \mathbf{g}}\right|_{\bar{\mathbf{g}}_m}. \tag{28}$$

Setting $\frac{\partial J}{\partial \delta \boldsymbol{\varpi}^T} = \mathbf{0}$, we find the optimal state update equation for a single iteration of Gauss-Newton:

$$\delta\boldsymbol{\varpi}^* = \left(\sum_m \mathbf{H}_m^T \mathbf{R}_{m,2}^{-1} \mathbf{H}_m\right)^{-1} \left(\sum_m \mathbf{H}_m^T \mathbf{R}_{m,2}^{-1} \bar{\mathbf{e}}_m\right) \tag{29}$$

Using the normal iterative Gauss-Newton approach, the solution of $\delta\boldsymbol{\varpi}^*$ can be used to update $\bar{\boldsymbol{\varpi}}$. The estimator requires a minimum of 3 well-spaced point correspondences. Note that 3 is only the minimum number of correspondences required; this estimator can be used to improve the $\boldsymbol{\varpi}$ estimate after finding a set of likely inliers.

2) Point Transformation: In order to compare each of the N constant-velocity models, we must evaluate the number of inliers. This is done by transforming each of the measurements, $\mathbf{y}_{m,1}$, into the corresponding frame, $\mathcal{F}_s(t_{m,2})$. The required transform, $\mathbf{T}_m$, can be calculated using (15).

In contrast to the rigid RANSAC algorithm, which calculates only one transform per model, our Motion-Compensated RANSAC algorithm requires that $\mathbf{T}_m$ be evaluated for each pair, m, using each of the N constant-velocity models. This process adds a significant amount of overhead to the computational cost.

D. Fast Motion-Compensated RANSAC

Although the nonlinear Motion-Compensated RANSAC algorithm is robust, the iterative estimator has to run for each of the N randomly seeded measurement subsets. In order to improve the performance of the algorithm, we propose an Euclidean least-squares estimator and a heuristic for the point transformation step.

1) Euclidean Least-Squares Estimator: Revisiting the estimation problem, we begin by reformulating the error in Euclidean space:

$$\mathbf{e}_m := \mathbf{P}\left(\mathbf{p}_{m,2} - \hat{\mathbf{p}}_{m,2}\right), \qquad \hat{\mathbf{p}}_{m,2} = \mathbf{T}_m \mathbf{p}_{m,1}, \tag{30}$$

where

$$\mathbf{P} := \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{31}$$

The advantage of this is that it eliminates the need to linearize the camera model. The disadvantage is that the algorithm is no longer optimizing with respect to the error that is evaluated in the inlier criterion (6).

Given that the two images are sequential, and relatively close in time, we propose the assumption that $\mathbf{T}_m$ is 'small', and therefore can be approximated using (12):

$$\mathbf{e}_m \approx \mathbf{P}\left(\mathbf{p}_{m,2} - (\mathbf{1} - \Delta t_m \boldsymbol{\varpi}^{\boxplus})\mathbf{p}_{m,1}\right) \tag{32a}$$

$$= \mathbf{P}\left(\mathbf{p}_{m,2} - \mathbf{p}_{m,1} - \Delta t_m (\mathbf{p}_{m,1})^{\boxminus} \boldsymbol{\varpi}\right) \tag{32b}$$

$$= \mathbf{q}_m - \Delta t_m \mathbf{Q}_m \boldsymbol{\varpi}, \tag{32c}$$

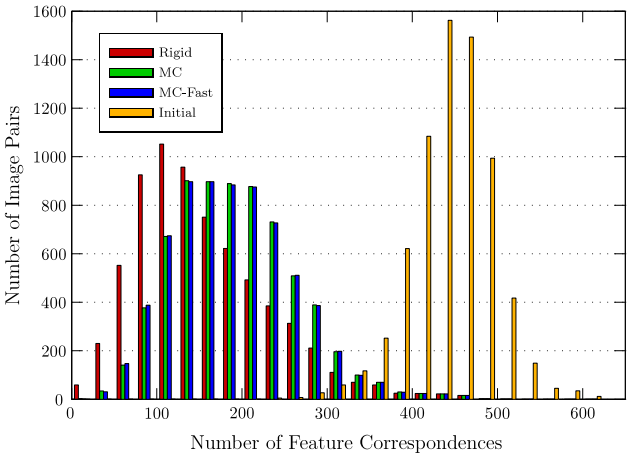

Fig. 6. This figure shows the distribution of sequential image pairs over the number of successfully matched feature measurements. The distribution of 'initial' matches indicates the total number of hypothesized matches before filtering (including outliers). The rigid, Motion-Compensated, and Fast Motion-Compensated filters are then applied to generate the plotted distribution of qualified inliers.

where $\mathbf{1}$ is the identity matrix and

$$\mathbf{q}_m := \mathbf{P}\left(\mathbf{p}_{m,2} - \mathbf{p}_{m,1}\right), \tag{33}$$

$$\mathbf{Q}_m := \mathbf{P}(\mathbf{p}_{m,1})^{\boxminus}. \tag{34}$$

The simplified objective function that we minimize is

$$J(\boldsymbol{\varpi}) := \frac{1}{2} \sum_m \mathbf{e}_m^T \mathbf{e}_m. \tag{35}$$

Inserting $\mathbf{e}_m$ into J and setting $\frac{\partial J}{\partial \delta \boldsymbol{\varpi}^T} = \mathbf{0}$,

$$\boldsymbol{\varpi} = \left(\sum_m \Delta t_m{}^2 \, \mathbf{Q}_m^T \mathbf{Q}_m\right)^{-1} \left(\sum_m \Delta t_m \mathbf{Q}_m^T \mathbf{q}_m\right), \tag{36}$$

which is solved in one step.

2) Discretization of Required Transforms: The Motion-Compensated problem formulation requires the calculation of the transform, $\mathbf{T}_m$, for each measurement pair, m, using each of the N proposed constant-velocity models. In order to improve computational performance, it is proposed that for each of the N models, finding only a discretized subset of the transforms is admissible; these transforms are uniformly sampled between the minimum and maximum values of Δt_m. When evaluating each measurement pair, m, the sampled transform with the closest Δt_m is used.

IV. Experimental Results

In this section, we test the two Motion-Compensated algorithms using data acquired from a real two-axis scanning lidar. Specifically, we make use of a 1.1km and 6880 frame traversal from *The Gravel Pit Lidar-Intensity Imagery Dataset* [20], which was acquired during a visual-teach-and-repeat experiment in Sudbury, Ontario, Canada. This experiment used an Autonosys LVC0702 lidar to capture azimuth, elevation, range, time, and intensity images, with a resolution of 480×360, at 2Hz, in unstructured, natural

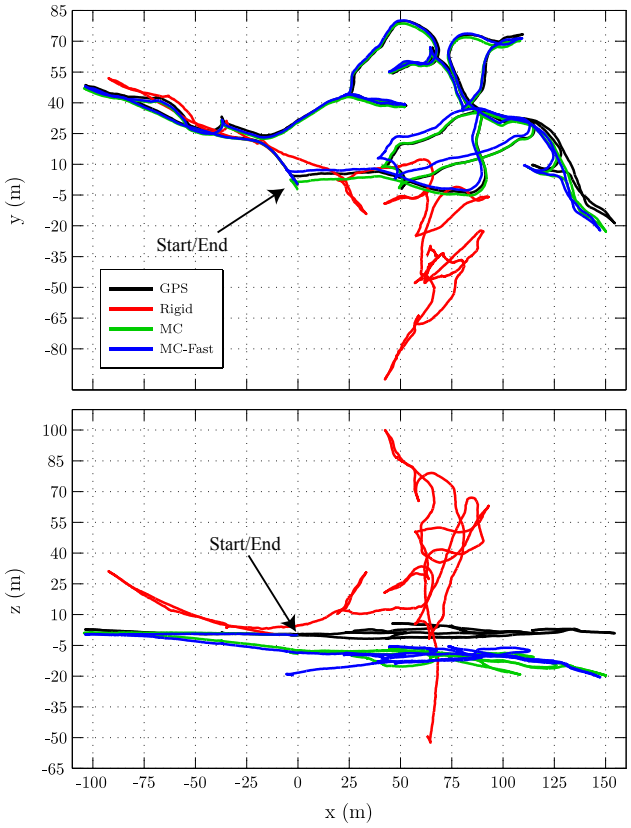

Fig. 7. Top and side views of the 1.1km trajectory. Feature matches were generated using the rigid, Motion-Compensated and Fast Motion-Compensated RANSAC algorithms. These matches were then used as inputs to a continuous-time SLAM algorithm. Each estimate can be compared to the available GPS tracks (in black).

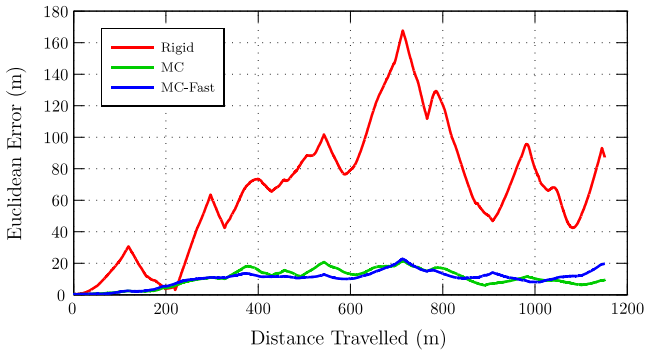

Fig. 8. This figure shows the Euclidean error growth between GPS and the different continuous-time SLAM outputs in Figure 7. Using the Motion-Compensated feature tracks obviously produces a much more accurate result that the rigid ones. Note that the drastic reductions in error are likely due to direction switches in the trajectory that tend to cancel existing pose drift.

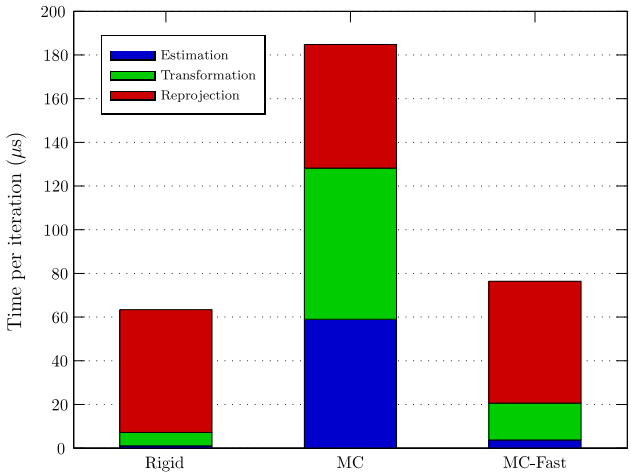

Fig. 9. This figure shows the required computational effort of each RANSAC iteration. The iteration time is broken down into the three most demanding parts of the algorithm: the time taken to estimate the mathematical model (given 3 points), the time it takes to generate and apply the transform provided by the model, and the time it takes to reproject the Euclidean points through the spherical camera model. All computations were performed in C++ and timed with a 2.4GHz processor.

terrain. During the traversal, the robot travelled between 0.0 and 0.5 m/s. A GPU-accelerated SURF implementation was used to extract features from the 8-bit intensity imagery; azimuth, elevation, range, and time measurements are then interpolated for each feature and initial matches are determined using SURF feature descriptor similarity. Outside the scope of this paper, a separate study is being conducted on the use of alternative feature detector and descriptors for motion distorted imagery.

A. *Quality*

The typical output of our Motion-Compensated RANSAC algorithm can be seen in Figure 5. In this section, we take two approaches to measuring the quality of the outlier-rejection schemes. First, we wish to evaluate the quantity of inliers that the algorithm is able to identify. Although we cannot know the true number of inliers between each pair of images, we can contrast algorithms by comparing the distribution of filtered feature correspondences against the initial distribution of possible feature matches. Assuming a relatively constant ratio between inliers and outliers, it is expected that the distributions should have a similar shape to the initial set, but a lower mean of feature correspondences. The results of this distribution comparison can be seen in Figure 6. Note that both the Motion-Compensated and Fast Motion-Compensated algorithms produce a near identical result, which, as expected, is distinctly better than the rigid RANSAC algorithm's output.

The second method used to evaluate the quality of outlier rejection is a relative, continuous-time SLAM algorithm [17]. The output of each RANSAC algorithm was used to initialize a set of open-loop feature correspondences; the window-style batch estimator was then run over each set of matches to produce an odometry estimate that can be compared to GPS ground truth. The output of the SLAM algorithm can be seen in Figure 7. Due to the 'slow' vertical scan of the lidar, the rigid RANSAC algorithm tends to find horizontal bands of temporally similar features; without a strong distribution of features in the vertical direction, it is expected that the pitch parameter of the estimation will suffer. The outputs of the algorithm fed by Motion-Compensated filters are clearly superior, as made obvious by the error plot in Figure 8.

B. Computational Efficiency

In order to characterize the computational efficiency of each of these algorithms, the time spent completing each major RANSAC step was recorded over 1500 iterations, for each of the 6880 frames and averaged, as seen in Figure 9. The iteration time is broken down into three major sections: the time taken to estimate the mathematical model, the time it takes to generate and apply the transform, $\mathbf{T}_m$, to each measurement, and the time it takes to reproject the Euclidean points into spherical coordinates.

The fastest algorithm, as expected, is the classic rigid one; however, the quality analysis suggests that the rigid algorithm is completely unfit for use with this sensor. Moving to the Motion-Compensated RANSAC algorithm, the large increase in estimation time is due to the addition of the iterative Gauss-Newton process. The increase in transformation time is due to the cost of calculating a transformation matrix for each measurement pair. The Fast Motion-Compensated RANSAC algorithm shows a drastic reduction in estimation time, although still not as fast as the rigid algorithm. For the transformation step, eight discrete transformations were used to approximate the constant-velocity model, which provided a significant speed up, with minimal cost to quality.

Note that the reprojection time is constant between all algorithms and is included purely to provide context to the total iteration time. Furthermore, note that the reprojection process can be ignored if the measurements are compared in Euclidean space. However, setting the threshold value appropriately is not as straightforward.

V. Conclusion and Future Work

This paper has derived a novel adaptation of the traditional RANSAC algorithm for use in the visual pipeline with rolling-shutter-type sensors. Using both the geometric and temporal measurement data, a batch estimator was derived to solve for the 6D constant velocity of the sensor. In order to find a likely set of true measurement pairs between sequential images, the estimator is applied iteratively for many random minimal (i.e., three) point sets. To improve computational performance, this paper also proposes an Euclidean least-squares estimator and a heuristic for the point transformation step. To validate the approach, each algorithm was applied to a sequence of 6880 lidar intensity/range scans acquired over a 1.1km trajectory. The next step in developing this technology is to test with different types of sensors. In particular, a stereo pair of synchronized rolling-shutter cameras, or an even slower lidar-style camera.

ACKNOWLEDGMENT

We would like to extend our deepest thanks to the staff of the Ethier Sand and Gravel in Sudbury, Ontario, Canada for allowing us to conduct our field tests on their grounds. We also wish to thank Dr. James O'Neill from Autonosys for his help in preparing the lidar sensor for our field tests. In addition, we would also like to acknowledge (from the Autonomous Space Robotics Laboratory) Colin McManus for being instrumental in gathering the data used in this paper, Andrew Lambert for his help in preparing the GPS payload, Paul Furgale and Chi Hay Tong for their work on the GPU SURF algorithm and Hang Dong for his work on the Autonosys calibration routine. Lastly, we also wish to thank the NSERC and the Canada Foundation for Innovation, DRDC Suffield, the Canadian Space Agency, and MDA Space Missions for providing us with the financial and in-kind support necessary to conduct this research.

References

[1] J. O'Neill, W. T. Moore, K. Williams, and R. Bruce, "Scanning system for lidar," 2010, U.S. Patent US 0053715 A1.

[2] C. McManus, P. T. Furgale, and T. D. Barfoot, "Towards appearance-based methods for lidar sensors," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, 9-13 May 2011, pp. 1930–1935.

[3] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," *Computer Vision–ECCV 2006*, pp. 404–417, 2006.

[4] M. Fischler and R. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Comm. of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[5] D. Nistér, "An efficient solution to the five-point relative pose problem," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, no. 6, pp. 756–770, 2004.

[6] B. M. Haralick, C.-N. Lee, K. Ottenberg, and M. Nölle, "Review and analysis of solutions of the three point perspective pose estimation problem," *International Journal of Computer Vision*, vol. 13, no. 3, pp. 331–356, 1994.

[7] M. Maimone, Y. Cheng, and L. Matthies, "Two years of visual odometry on the mars exploration rovers," *Journal of Field Robotics*, vol. 24, no. 3, pp. 169–186, 2007.

[8] O. Ait-Aider, N. Andreff, J. Lavest, and P. Martinet, "Simultaneous object pose and velocity computation using a single view from a rolling shutter camera," *Comp. Vision–ECCV 2006*, pp. 56–68, 2006.

[9] O. Ait-Aider and F. Berry, "Structure and kinematics triangulation with a rolling shutter stereo rig," in *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 1835–1840.

[10] C. Jia and B. L. Evans, "Probabilistic 3-d motion estimation for rolling shutter video rectification from visual and inertial measurements," in *Proc. IEEE Int. Workshop on Multimedia Signal Processing*, 2012.

[11] J. Hedborg, P. Forssén, M. Felsberg, and E. Ringaby, "Rolling shutter bundle adjustment," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 1434–1441.

[12] B. D. Lucas, T. Kanade, *et al.*, "An iterative image registration technique with an application to stereo vision," in *Proceedings of the 7th International Joint Conference on Artificial Intelligence*, 1981.

[13] J. Hedborg, E. Ringaby, P.-E. Forssén, and M. Felsberg, "Structure and motion estimation from rolling shutter video," in *IEEE International Conference on Computer Vision Workshops*, pp. 17–23, 2011.

[14] H. Dong and T. D. Barfoot, "Lighting-invariant visual odometry using lidar intensity imagery and pose interpolation," in *Proceedings of the International Conference on Field and Service Robotics (FSR)*, Matsushima, Japan, 16-19 July 2012.

[15] B. K. Horn, "Closed-form solution of absolute orientation using unit quaternions," *JOSA A*, vol. 4, no. 4, pp. 629–642, 1987.

[16] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-d point sets," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, no. 5, pp. 698–700, 1987.

[17] S. Anderson and T. D. Barfoot, "Towards relative continuous-time slam," in *Proceedings of the IEEE Intl. Conference on Robotics and Automation (ICRA)*, Karlsruhe, Germany, 6-10 May 2013.

[18] P. T. Furgale and T. D. Barfoot, "Associating uncertainty with three-dimensional poses for use in estimation problems," submitted to the *IEEE Transactions on Robotics* on February 5, 2013. Manuscript # TRO-13-0050.

[19] G. M. T. D'Eleuterio, "Dynamics of an elastic multibody chain: Part c – recursive dynamics," *Dynamics and Stability of Systems*, vol. 7, no. 2, pp. 61–89, 1992.

[20] S. Anderson, C. McManus, H. Dong, E. Beerepoot, and T. D. Barfoot, "The gravel pit lidar-intensity imagery dataset," submitted to the *International Journal of Robotics Research* on March 6, 2013. Manuscript # IJR-13-1695.