Evaluation of heterogeneous measurement outlier rejection schemes for robotic planetary surface mapping

Chi Hay Tong*, Timothy D. Barfoot

University of Toronto Institute for Aerospace Studies, 4925 Dufferin Street, Toronto, Canada M3H 5T6

Article info

Article history:
Received 4 April 2012
Received in revised form 15 June 2012
Accepted 7 August 2012
Available online 17 September 2012

Keywords:
Surface mapping
Outlier rejection
SLAM
Mobile robotics

Abstract

In this paper, we describe the development and evaluation of a core algorithmic component for robust robotic planetary surface mapping. In particular, we consider the issue of outlier measurements when utilizing both odometry and sparse features for laser scan alignment. Due to the heterogeneity of the measurements and the relative scarcity of distinct geometric features in the planetary environment, we have found that the conventional outlier rejection methods in the current literature do not produce satisfactory classifications for accurate mapping performance. In light of these limitations, we develop a new approach capable of addressing these concerns. This includes a family of four outlier classification algorithms, which are incorporated through iterative reclassification into the batch alignment framework to provide robust surface mapping performance. Characterization of these outlier rejection schemes is presented using a combination of simulated data and real-world testing with an indoor rover.

1. Introduction

The goal of establishing a permanent presence on the lunar surface will be facilitated by robotic precursor missions [1]. In contrast to the exploratory traverses conducted by the Mars Exploration Rovers (MERs), rovers will perform extended operations in local worksite environments, revisiting the same locations multiple times. Much like a construction project on Earth, detailed maps will be required to conduct operations such as site selection, site preparation, and base construction. As a result, the creation of an accurate surface map will be the first step for many of these operations.

The desire for continuous robotic operations in a planetary environment comes with a number of implications. Without existing infrastructure such as Global Positioning System (GPS), on-board sensing must be used for navigation. In addition, operation in permanently shadowed regions restricts the choice of sensors. Finally, operations on planetary surfaces require a system that is able to handle the rugged, three-dimensional (3D) terrain, as depicted in Fig. 1. Therefore, we approach the joint task of rover localization and surface map construction in an unknown environment as a 3D Simultaneous Localization and Mapping (SLAM) problem.

For 3D SLAM in unstructured environments, many common implementations in the literature utilize passive cameras. In particular, impressive results have been demonstrated using a stereo camera, in both frame-to-frame Visual Odometry (VO) [2], and Bundle Adjustment (BA) [3] for loop closure. These methods are robust, providing accurate estimates even in the presence of numerous data association errors, or in dynamic environments where spurious features may arise. Unfortunately, the small field of view and reliance on external lighting make cameras unsuitable for large-scale mapping in varying lighting conditions.

Actively illuminated sensors such as a laser rangefinder are more appropriate for this scenario. Though
intensity information could be used in a similar manner as camera imagery \([4]\), we consider the more conventional data product of 3D point clouds. One approach to handling these dense point clouds involves aligning them in a pairwise fashion using Iterative Closest Point (ICP) \([5]\), which is applied sequentially to scans as they are obtained throughout the rover’s traverse \([6]\). However, the requirement of sufficiently diverse overlapping terrain limits this method to short motions between scans \([7]\). Due to power limitations, it is more likely that a planetary rover will survey the terrain by obtaining longer-range scans during infrequent stops.

Alternatively, a sparse-feature-based approach may be taken, where geometrically distinct interest points are extracted from each scan. These points can include peaks in the terrain \([7,8]\), or regions of high curvature \([9]\). Matching these points between scans provides tie-points for scan alignment.

Unlike the feature-rich data provided by camera imagery, geometric features in natural terrain tend to be very sparse. This could cause issues for pairwise scan matching, as a minimum of three common features are required for 3D alignment. In such situations, additional sensor measurements are required to align scan pairs. For example, a common source of sequential pose estimates for mobile rover platforms is wheel odometry; VO is another possibility. These odometry measurements can be used to supplement the alignment problem in case of measurement scarcity. Furthermore, the global consistency of the estimate can be further improved through loop closure, approaching the multi-scan alignment as a batch SLAM problem \([7,8,10,11]\).

While the measurement models used in these alignment algorithms are formulated to account for noise corruption, invariably, erroneous sensor measurements still occur. Outliers are defined as measurements that do not fit the noise model, which typically arise in mobile robot applications as incorrect data associations. This can be attributed to the fact that the majority of robotic applications utilize a single sensor type. However, outliers can also arise due to sensor failures or algorithmic shortcomings such as catastrophic wheel slippage in odometry. A robust alignment algorithm should be able to detect all types of measurement outliers, and reject them from being used in the alignment. In this paper, we will refer to the task of detecting and rejecting outliers in multiple measurement types as heterogeneous measurement outlier rejection (HMOR). While HMOR may repeat some tests already conducted in the data association phase, we have found in practice that the redundancy improves robustness towards real-world issues.

In short, HMOR is essential for robust surface mapping performance. This is supported by the images presented in Fig. 2, where the HMOR methods developed in this paper produced accurate maps of planetary analogue terrain. We compare these results to Fig. 3, where conventional approaches were used instead. As can be seen, the inclusion of the HMOR component provided significantly improved performance.

This paper describes the characterization performed to select a HMOR scheme appropriate for the robotic planetary surface mapping scenario. This characterization was conducted using a single high-performance laptop, in anticipation of faster flight-qualified systems in the future. This work predates the development of the entire robotic surface mapping framework presented in \([7]\), summarizing the experimental analysis conducted to inform the design decisions. Unit testing of the various HMOR methods was performed using controlled environments, where ground truth data for both the rover poses and the map were available. As a result, the analysis presented in this paper is based on experiments conducted in simulation and in an indoor lab environment. For overall performance of the mapping framework in outdoor terrain, see \([7]\).

In this paper, we will consider the situation where only odometry and sparse feature position measurements are available. However, a real-world scenario could include many other measurement types, such as inertial sensing, line features, or satellite imaging for periodic global localization updates. The algorithms presented in this paper are generalized for any measurement type where a probabilistic model is available.

This paper provides an extended description of the outlier rejection criteria presented in \([13]\), including more detailed analysis of the unique elements of our scenario,
an algorithm for inlier classification, a discussion on implementation issues, and additional validation conducted using a real robot.

The remainder of the paper is organized as follows. Section 2 details key related work, and Section 3 describes the conventional approach to outlier rejection, RANSAC [12], in further detail, identifying its limitations in applying it to our scenario. This is followed by Section 4, which introduces the proposed framework for HMOR. Finally, in Section 5, the framework and its algorithms are characterized through experiments, and concluding remarks are provided in Section 6.

2. Literature review

In this section, we present a brief review of related work on outlier rejection algorithms and robust estimation for SLAM. With demonstrated real-world success in utilizing camera-based systems, the main body of literature concerning these problems primarily comes from the field of computer vision.

For camera-based systems, the predominant method for outlier rejection is the RANdom SAmple and Consensus (RANSAC) [12] algorithm. RANSAC was originally developed for estimating image transforms in the presence of corruptive noisy data. Using the consensus heuristic and exploiting randomness to make the problem computationally tractable, RANSAC has been applied to numerous problems in computer vision to deal with the issue of feature-based outliers. In particular, RANSAC is often used for verifying frame-to-frame feature matches for the purposes of motion estimation [14]. Due to the large number of features extracted from the images, it is unnecessary to consider additional wheel odometry measurements. Unfortunately, the relative sparsity of features in the planetary mapping scenario conflicts with some of the basic assumptions required to use RANSAC. These issues will be analyzed in further detail in Section 3, which will prompt the need to develop a new approach.
An alternative approach to robust estimation involves absorbing the outlier classification problem into the estimation algorithm itself. M-estimators \[15\] are a generalization of the maximum-likelihood approach used for alignment, which also accounts for outlier measurements. The solution method provided by M-estimation consists of classifying and reweighting the influence of each measurement based on the current guess at each iteration. In effect, an additional noise model is introduced for the outlier measurements, separate from the inliers. The standard approach for M-estimation utilizes the measurement uncertainties directly for outlier classification.

Though scale estimation in Bundle Adjustment using M-estimators has been considered in the past \[16\], this paper offers a more principled approach using uncertainty propagation for use in the iterative reclassification stage. These two approaches comprise the majority of outlier rejection methods in the current literature. Efficient variations exist for these two methods, but the underlying assumptions are the same. A related body of work can be found within the research on the data association problem, which sits upstream of outlier rejection to identify which feature measurements correspond with previously observed features. As one of the roles of outlier rejection is to detect incorrect data associations, some relations can be identified from the algorithms that exist to form correct associations in the first place. Though this is not the only role of outlier rejection (e.g., it should also detect spurious measurements), some ideas can be drawn from the field.

The target-tracking literature offers some options for culling unlikely associations. Validation gating \[17\] is a method that applies a statistical test to potential measurements, verifying if the measurement is possible under the assumed noise model. The test is applied on individual measurements, via a threshold on the Gaussian noise distribution. This results in a reduction in the search space of potential associations. Similarly for navigation tasks, the gating can be extended to account for the uncertainty in the sensor pose, producing the Normalized Innovation Squared (NIS) test \[18\].

However, since it was found that individual gating is extremely unreliable if the vehicle pose is very uncertain \[19\], batch gating was introduced, which considers multiple associations simultaneously. Joint Compatibility Branch and Bound (JCBB) \[20\] provides a statistical test using an incrementally growing joint covariance structure between feature assignments, and Combined Constraint Data Association (CCDA) \[21\] seeks the largest set of assignments that are valid under individual and pairwise constraints. Active Matching \[22\] utilizes similar statistical tests to restrict feature detection to valid regions defined by the joint covariance structure.

In summary, there are few options in the current literature for robust estimation. The two main approaches, RANSAC and M-estimation, are typically used pairwise for computing the alignment between sequential camera images. In these algorithms, the statistical tests used to classify outliers are applied to measurements individually. While it will be shown in Section 4.2 that these tests are approximate, they are sufficient for frame-to-frame image matching due to the abundance of measurements. However, this does not transfer to our scenario, since the batch formulation results in a very high-dimensional state, and the relative scarcity of the terrain features.

More advanced statistical tests can be found in the work on data association, where algorithms that consider the uncertainty in the state hypothesis and the joint compatibility of batch associations may be found.
However, these methods operate at a single timestep, and are not easily extended to the multi-frame case since the estimation problem is not considered in the association process. Furthermore, all of the algorithms reviewed above are limited by the fact that they only detect outliers in a single measurement type.

To reiterate, existing data association methods cannot be applied to HMOR because they are not outlier rejection schemes; outliers can also arise due to spurious measurements, sensor failures, or algorithmic shortcomings. In light of this, no work was found that simultaneously considered outliers in multiple measurement types.

3. Conventional approach

Due to its status as the standard approach to outlier rejection, our initial approach involves attempting to apply RANSAC to address HMOR. We briefly present the conventional pairwise RANSAC algorithm, and in doing so, identify its limitations for our scenario. An attempt is then made to adapt RANSAC to the batch scenario, but it is found that fundamental assumptions are violated in this case, motivating the need for a new approach.

3.1. Pairwise RANSAC

The RANSAC framework typically serves as a preprocessing step, validating feature matches made between pairs of images prior to alignment. The primary assumption necessary for RANSAC is that all of the inlier measurements follow some underlying model, whereas the outlier measurements do not. In pairwise image alignment, the underlying model is the true transformation. Therefore, with this common model, the inlier measurements should reach a consensus for what the image transformation should be.

RANSAC is predicated on the belief that if an alignment hypothesis is computed based on a measurement subset composed only of inliers, it will accurately model the true state within the bounds of sensor noise. Furthermore, the largest consensus set will agree with that model, also within the bounds of measurement uncertainty. As a result, a minimal number of measurements are chosen for the subset, minimizing the chance of outlier corruption. By repeating this random selection a large number of times, it is expected that at least one minimal set will contain only inliers, and as a result, the true transformation will be recovered.

The number of trials required to obtain an outlier-free minimal set is computed by considering the random selection to be a Bernoulli trial, where the probability of failure is the corruption rate of the data. While this value is exponential in the number of measurements composing the minimal set, in the case of pairwise alignment of 3D points, this number is only three. However, this minimal requirement of three features also limits this outlier rejection approach to scenarios where four or more common features can be identified between frames. Since the sparsity of features in our mapping scenario can lead to numerous pairs of scans with fewer than four common features, this limitation alone is sufficient to discount the standard pairwise algorithm.

3.2. Batch RANSAC

To address the minimum-feature limitation as well as the presence of odometry outliers, let us consider extending RANSAC to the batch heterogeneous outlier case. By using a batch alignment method that incorporates odometry to compute the alignment hypothesis, such as GraphSLAM [10], we require the selection of a minimal initial set for the first step. However, the composition of a minimal initial set is not straightforward for heterogeneous measurements, because the dimensions and contributions to the estimated state are not equal for each measurement type. For example, two rover poses can be linked together by either a single 3D odometry measurement, or six feature measurements. As a result, the number of measurements in the initial set varies due to its composition. This presents complications when evaluating the quality-of-fit metric to identify the largest consensus set. Furthermore, since the number of trials required to obtain an outlier-free initial set is exponential in the number of measurements required, large alignment problems are computationally intractable.

In addition, odometry measurements have different noise properties than feature measurements. As a result, they cannot be handled in the same way. For example, since wheel odometry is an imprecise method of computing a rover pose transformation, even inlier odometry measurements tend to have very high uncertainty. If these methods are utilized for computing the alignment hypothesis, the resulting measurement errors computed based on this hypothesis model would also be large, reflecting the imprecision of the odometry measurements. These errors would not fall within the noise model for the individual measurements, resulting in a large number of inlier measurements misclassified as outliers. This is unacceptable, given the measurement scarcity in our scenario. A more appropriate statistical test should be used that classifies the measurements based on the uncertainty in the hypothesis model. A common currency is also required to compare and classify the heterogeneous measurement types.

4. Proposed framework

Since the RANSAC approach as a whole is unsuitable for our scenario, a different framework is required for heterogeneous outliers in batch alignment. To simplify the formulation of a new technique, we consider the problem where a hypothesis for the state of the world is given, and we must determine which measurements agree with the hypothesis. Three outlier classification algorithms are presented, which range in their level of approximation and computational requirements. An additional algorithm is also developed for the inverse problem of inlier classification, which may be more suitable for some problems. These statistical tests are then followed by a discussion on how the hypothesis models can be
4.1. Measurement models

Though the measurements come from different sources, their mathematical models typically share a common form. That is, if we define the true state as $x$, a vector composed of the true rover poses and feature positions, probabilistic measurements are of the form

$$z_i := h_i(x) + e_i, \quad e_i \sim N(0, P_i).$$

where $e_i$ is the noise variable, and $i$ is the index of the measurement, $z_i$. For simplicity, we make the standard assumption that the noise for the measurements is additive, zero-mean, and Gaussian with covariances, $P_i$. Since both odometry and feature measurements can be expressed in this common form of a nonlinear function of the true state and corruptive noise, other than the size of the vectors involved, there is no need to differentiate between the measurement types. This common form will be utilized as a method to compare all the measurement types in a common currency.

4.2. Measurement classification

For outlier classification, we will consider the problem of detecting outliers in a set of measurements, given a hypothesis model. The hypothesis model consists of some estimated values of the state, $\tilde{x}$, as well as the uncertainties associated to them. Since there is an uncertainty to the estimate, we model it as a random variable with the following normal distribution:

$$\tilde{x} := x + \Delta \tilde{x}, \quad \Delta \tilde{x} \sim N(0, \tilde{\Sigma}).$$

This relation states that the estimated state is assumed to model the true state, with an additive error that follows a zero-mean Gaussian distribution with a covariance, $\tilde{\Sigma}$, relating to our confidence in the estimate. In the following sections, three statistical tests are developed, with each progressively considering more of the noise properties involved. The first test uses the measurement models alone, the second incorporates the uncertainty in the hypothesis model, and the third considers the measurements as a batch. Finally, a summary is provided that compares all three tests.

4.2.1. Standard test

We begin with a simple test that we term the **Standard Test**, since it is the one that RANSAC, M-estimation, and validation gating use. As is common to all three methods, the first step is to compute the estimated error for each measurement, $\tilde{e}_i$, between the measurement obtained by the sensor and the expected measurement (computed using the hypothesis model):

$$\tilde{e}_i := z_i - h_i(\tilde{x}).$$

We term this the **estimated error** because the true state is unavailable, so the measurement error is **estimated** based on the hypothesis model. In the standard approach, the estimated error is assumed to follow the same probability distribution as the true error. That is,

$$\tilde{e}_i \sim N(0, P_i).$$

Using this property, the Mahalanobis distance (also known as the statistical distance) can be computed for the estimated error. This should follow a $\chi^2$ distribution with its degrees of freedom equal to the dimension of the measurement (if the property were true):

$$\tilde{e}_i^T P_i^{-1} \tilde{e}_i \sim \chi^2_{\text{dim}(e_i)}.$$  

With this distribution, if the value of the distance is above a predefined threshold, the measurement is classified as an outlier. Since the distribution and its associated thresholds are appropriately scaled by the dimension of the measurement, both measurement types are handled in the same way.

4.2.2. Independent Innovation Test

However, since noisy measurements were used to compute the hypothesis model, there is also an uncertainty to the estimate. The standard approach does not take this fact into account, which could result in many inliers misclassified as outliers. For example, if we are less certain about a rover pose and feature position, we should also be less certain about the exact distance between them. Therefore, we inflate the threshold for outlier classification by propagating the hypothesis model uncertainty through the measurement error function. This can be accomplished by linearizing the error function, and passing the covariance through the linear system. Let us define the linearized form of the estimated error (3) to be

$$\tilde{e}_i = z_i - h_i(\tilde{x}) \approx z_i - h_i(x) - H_i \Delta \tilde{x} = e_i - H_i \Delta \tilde{x},$$

where $H_i$ is the Jacobian of the $i$th measurement function evaluated at the true state, $x$. With the uncertainty model given by (2), we can analyze the statistical properties of $\tilde{e}_i$. The first moment is

$$E[\tilde{e}_i] \approx E[e_i - H_i \Delta \tilde{x}] = E[e_i] - H_i E[\Delta \tilde{x}] = 0,$$

and the second moment is

$$E[\tilde{e}_i \tilde{e}_i^T] \approx E[(e_i - H_i \Delta \tilde{x})(e_i - H_i \Delta \tilde{x})^T]$$

$$= E[e_i e_i^T] - E[e_i \Delta \tilde{x}^T H_i^T] - E[H_i \Delta \tilde{x} e_i^T] + E[H_i \Delta \tilde{x} \Delta \tilde{x}^T H_i^T].$$

Since the hypothesis model is separate from the measurements that we wish to classify, the hypothesis model error, $\Delta \tilde{x}$, and the measurement errors, $e_i$, are independent. Applying this fact, that the expected value of the cross terms should be zero, results in a simple expression for the propagated uncertainty of the estimated errors:

$$E[\tilde{e}_i \tilde{e}_i^T] \approx E[e_i e_i^T] + E[H_i \Delta \tilde{x} \Delta \tilde{x}^T H_i^T]$$

$$= P_i + H_i E[\Delta \tilde{x} \Delta \tilde{x}^T] H_i^T$$

$$= P_i + H_i \tilde{\Sigma} H_i^T.$$  

Finally, by maintaining the Gaussian noise assumption, we can summarize the noise properties of $\tilde{e}_i$ as

$$\tilde{e}_i \sim N(0, P_i + H_i \tilde{\Sigma} H_i^T).$$
As can be seen, the covariance of the estimated error is inflated by the uncertainty in the hypothesis model. Since each error is computed independently, and the estimated error is now of the same form as the innovation term in the Kalman Filter, we call this test the Independent Innovation Test. Since the true state is unavailable in practice, $H_i$ is approximated by evaluating it at the estimated state. This independently-computed Mahalanobis distance, also known as the Normalized Innovation Squared (NIS) [18], can be used to classify the measurements in a similar manner as the standard approach.

4.2.3. Batch Innovation Test

Although the hypothesis model uncertainty is considered in the Independent Innovation Test, it still remains an approximation of the true noise model. This is because it assumes independence between the estimated errors. While the measurements may have been obtained independently, the estimated errors are all based on a common hypothesis model. This common basis links the values together, providing correlations between the estimated errors. For example, if we had two feature measurements obtained from a very uncertain pose, we would expect the estimated errors to be large together. The estimated errors into a single vector, $\mathbf{e}$, which results in the same form as (11). That is, by making the following definitions,

$$
\mathbf{\dot{e}} = \begin{bmatrix} \mathbf{\dot{e}}_1 \\ \vdots \\ \mathbf{\dot{e}}_N \end{bmatrix}, \quad \mathbf{P} = \begin{bmatrix} \mathbf{P}_1 \\ \vdots \\ \mathbf{P}_N \end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix} \mathbf{H}_1 \\ \vdots \\ \mathbf{H}_N \end{bmatrix},
$$

(12)

we obtain

$$
\mathbf{\dot{e}} \sim N(0, \mathbf{P} + \mathbf{H} \Sigma \mathbf{H}^\top).
$$

In the Batch Innovation Test, we consider the result of this noise propagation, which implies that there are correlations between the estimated errors. Although (13) provides an expression for the distribution of $\mathbf{\dot{e}}$, we cannot simply apply a threshold on the Mahalanobis distance as in the previous sections. This would identify whether or not the entire set of measurements is an outlier, as opposed to classifying the individual measurements. An alternative test must be formulated that is able to recover the fact that regardless of the coupling of the estimated errors, the underlying measurements were still obtained independently. The JCBB [20] criterion considers the measurements as a group, and as a result, does not provide a means for identifying outliers in a group that fails the joint compatibility test.

To regain the independence of the measurements, we use the properties of the $\chi^2$ distribution. The $\chi^2$ distribution is defined as the sum of $K$ squared standard normal Gaussians. That is, if we define a random variable

$$
q = \sum_{i}^{K} s_i^2, \quad s_i \sim N(0, 1),
$$

(14)

its values would be distributed according to the $\chi^2_k$ distribution:

$$
q \sim \chi^2_k.
$$

(15)

Using this fact, if we had two random variables drawn from $\chi^2$ distributions,

$$
q_{K} \sim \chi^2_{k},
$$

(16)

$$
q_{K-d} \sim \chi^2_{k-d},
$$

(17)

we would expect that the following relation also holds true:

$$
(q_{K} - q_{K-d}) \sim \chi^2_{d}.
$$

(18)

Applying this property to outlier classification, we remove one measurement from the set, and compute the Mahalanobis distance of the remaining estimated errors as a batch. Repeating this for all of the measurements and subtracting the values from the Mahalanobis distance of all of the estimated errors together, we can utilize a threshold on the $\chi^2_d$ distribution as before. This leave-one-out approach regains the independence of the measurements, as each measurement is tested independently.

However, we cannot simply classify the measurements that fail the $\chi^2_d$ test as outliers, since considering the measurements as a group resulted in an assumption that would produce numerous misclassifications. In computing the Mahalanobis distance, we made the assumption that all of the estimated errors follow the computed Gaussian distribution. As outliers are defined as measurements that do not follow the noise model, incorporating them into the computation leads to the corruption of the distance metric as a whole. Therefore, if any one measurement fails the $\chi^2_d$ test, the only conclusion that can be reached is that at least one outlier is present in the set.

At this point, we apply a heuristic that the measurement that performed the worst on the test is the least likely to be an inlier. Accordingly, it should be classified as an outlier and removed from the set. The test is then repeated until all of the measurements pass, resulting in an inlier-only set.

This approach can also be adapted to address the inverse problem of inlier classification. Starting with an empty set of measurements, we can incrementally build up a set of inlier by adding individual measurements to the set, and computing the increase in the Mahalanobis distance. If any of the measurements pass the $\chi^2_d$ test, then at least one inlier is present in the set. Again, we apply a heuristic that the measurement with the lowest corresponding Mahalanobis distance is most likely to be an inlier. This test is exhaustively repeated until the remaining candidate measurements are all outliers, resulting in an inlier-only set. As will be evidenced in the following section, this alternative approach may prove to be more computationally efficient in certain scenarios.

In both the outlier and inlier classification approaches, numerous Mahalanobis distance computations are required. As a result, naively inverting the covariance matrix for each computation can prove to be very costly. This is because in the Batch Innovation Test, the covariance matrix is the size of the set of measurements combined.

To address this, an incremental method of computing the inverse of a matrix given the inverse of a submatrix should be used for inlier classification, and a decremental
method which computes the inverse of a submatrix given the inverse of the whole matrix should be used for outlier classification. These methods can be obtained by rearranging the results of the incremental method presented in [20]. These efficient algorithms result in matrix inversions that are reduced to the same dimension as the individual measurements, which is the same size as the inversions required for the previous two tests.

Unfortunately, due to the number of iterations required, the Batch Innovation Test is still computationally expensive compared to the previous two. In the Standard and Independent Innovation Tests, Mahalanobis distances are computed only once for each measurement in the set. That is, if we let \( N \) be the number of measurements in the classification set, both algorithms have a computational complexity of \( O(N) \), where the matrix inversion computation is considered to be a constant-time process because it is fixed by the number of dimensions of each measurement. In contrast, the Batch Innovation Test repeats this process for each outlier detected, which, if we define \( M \) as the number of outliers in the set, results in a complexity of \( O(MN) \) for outlier classification, and \( O(N-MN) \) for inlier classification. While the computational requirements are larger, the test is significantly more accurate. Some efficient approximations are presented in the following section.

4.2.4. Summary

In summary, we have presented three statistical tests for classifying measurements given a hypothesis model. Each of these tests strikes a different balance between accuracy and computational requirements. The choice should be made based on the problem at hand. To help visualize the differences between the algorithms, we consider a simple classification example.

Since the relations in an actual rover alignment problem are too difficult to visualize, we utilize a 2D problem instead. In Fig. 4, we depict samples drawn from a jointly coupled Gaussian, similar to the estimated error relations derived in (13), and the classifications made by applying the presented tests. The labels OC-S, OC-I, and OC-B/IC-B are used to identify the thresholds related to the Standard, Independent Innovation, and the outlier and inlier variants of the Batch Innovation Tests, respectively.

The small box in Fig. 4(a) represents the threshold for OC-S, as it simply uses the given measurement uncertainty and assumes independence between the two dimensions. This threshold correctly classifies all of the outliers, but due to its size, it also misclassifies many inlier measurements as well. The larger box in Fig. 4(b) corresponds to the inflated uncertainty as a result of incorporating the uncertainty in the hypothesis model by OC-I, but the independence assumption remains. As a result, some outliers are incorrectly classified as inliers.

Finally, the ellipse in Fig. 4(c) illustrates a threshold where both the noise propagation and the correlations are considered. The majority of the measurements are correctly classified, with a few inliers incorrectly classified on the edge of the threshold. These misclassifications occur as a result of using a hard threshold, which does not reflect the actual tail of a Gaussian distribution. In OC-B/IC-B, a similar threshold is used, but the measurements are tested independently. However, this ellipse best illustrates how OC-B/IC-B compares to the other two.

To confirm the computational complexity analysis presented in the previous section, a small simulation was run to illustrate how these algorithms scale with the number of outliers in the set of measurements. In this simulation, test cases were generated with 100 3D feature position measurements each, where the number of outliers in each set varied from 0 to 50. Then, each of the classification algorithms were run on the test cases, recording the average processing time over 10 trials. The results of these simulations\(^1\) are depicted in Fig. 5.

We see that both the OC-S and OC-I methods remain constant in their processing time given any number of outliers, with very low computational requirements (note the difference in scale between the two figures). This corresponds well with the presented computational complexity of \( O(N) \).

In comparison, OC-B requires more computation as the number of outliers increases, since the test is repeated for each outlier identified. Though a complexity of \( O(MN) \) should result in a straight line, the slope of the curve

---

\(^1\) Produced using a MacBook Pro with a 2.66 GHz Core 2 Duo and 4 GB of 1067 MHz DDR3 RAM in Matlab.
flattens out due to the reduction in matrix dimensionality as outliers are removed. IC-B scales in a similar manner, but with the number of inlier measurements. As expected, there is an advantage to either batch method, depending on the ratio of inliers to outliers. While we would expect the tradeoff between the two algorithms to be where the number of inliers matches the number of outliers, it can be seen in the plot that the crossover point is much sooner. This can be attributed to the fact that though the number of iterations in each algorithm scales with the number of classifications required, the OC-B algorithm simply starts with larger matrices. This translates into more costly matrix multiplication and memory allocation operations.

However, the processing time required for the batch methods is still significantly larger than for the independent methods. We can address this issue by exploiting the interrelated nature of the tests to make some approximations to speed up the classification task. Looking at Fig. 4, we see that the majority of the points that fall within the OC-S threshold are also contained within the OC-B/IC-B ellipse, and that points beyond the OC-I threshold tend to be also outside the OC-B/IC-B ellipse. Though some inliers are misclassified, the large measurement errors tend to not provide a positive contribution to the alignment accuracy anyway. Therefore, a rough pre-classification step can be utilized prior to applying the OC-B/IC-B criterion, by classifying all estimated errors that pass the OC-S test as inliers, and those that fail the OC-I test as outliers. Due to the simplicity of the tests, this rough classification is inexpensive to compute, and reduces the classification problem size. Similarly, we can approximate the tests by classifying measurements with very low error values as definitely inliers, because they would contribute a negligible amount to the Mahalanobis distances. The use of these rough classifications ensures that the actual OC-B/IC-B test is only applied to identify the more subtle outliers.

4.3. Hypothesis generation

With statistical tests suitable for classifying heterogeneous measurements in the batch scenario given a hypothesis model, we now consider some approaches for generating the hypothesis models themselves. For accurate classification, we desire a rough estimate that is also robust to the presence of outliers in the measurement set.

One option is to look to the established results of RANSAC. Though the algorithm is inapplicable to our scenario as a whole, if we conduct RANSAC in a pairwise fashion between frames, some feature measurement outliers can be identified and removed from the set. Though some misclassifications may be made, any outlier removals will only serve to improve the estimate. Though it is known in robust estimation theory that the presence of a single outlier can corrupt the entire estimate, the severity of outliers in robotic applications tends to be limited in practice. We can then use the hypothesis model generated by RANSAC as a seed for HMOR using the previously developed statistical tests. In effect, we form a hierarchal method to identify heterogeneous outliers, where pairwise RANSAC provides some initial classification, and our statistical tests complete the task.

4.4. Iterative reclassification

Since the classification performance of the statistical tests presented depends highly on the quality of the hypothesis model, the imperfect alignment produced by applying RANSAC is still likely to result in some incorrect classifications. If a method were available to improve the hypothesis models, a better classification performance would result.

Inspired by M-estimators [15], incorporating the outlier classification into the alignment process itself will produce a robust estimate. Batch estimation problems are typically framed as nonlinear optimization problems, which seek the maximum likelihood estimate of the state given the measurement data. The predominant approach [7,8,10,11,23] involves utilizing the Gauss–Newton algorithm [24] to iteratively improve an estimate, given a weighted nonlinear least-squares objective function, until convergence at a locally optimal point. Therefore, the improved estimate at each iteration can be used to obtain
an improved set of classifications for the next iteration. This approach is illustrated in Fig. 6.

This is a simple and modular framework, where the components are easily replaced depending on the requirements and limitations of the scenario at hand. For measurement classification, any of the presented statistical tests can be used, or even interchangeably applied. For conventional M-estimation, a threshold similar to the OC-S test is applied, and the weights associated to the measurements are readjusted based on their classifications. Our approach offers more advanced statistical tests, which can be combined with various reweighting options to create a smooth optimization surface.

However, without additional prior knowledge about the characteristics of the outlier measurements, more assumptions are required to select an appropriate reweighting scheme. Furthermore, due to the interrelated nature of the measurements for the batch tests, reweighting would need to take the coupling into consideration. As a result, we elect to simply remove the classified outlier measurements at each iteration, and give them a chance to be reclassified as inliers later on. This is justified by the fact that outliers in our scenario tend to be catastrophic wheel slippage and feature misassociations, which are not useful for the alignment.

4.4.1. Implementation issues

In practice, two main issues arise if the framework is simply implemented as depicted in Fig. 6. First, if the current estimate at an iteration is poor, too many measurement may be misclassified as outliers. This results in a numerically unstable system of equations. This instability can be addressed by using the Levenberg–Marquardt [25,26] augmentation to the Gauss–Newton method to maintain numerical stability in certain regions of the objective space.

Second, when reclassifying the measurements at each iteration, cyclical classifications may result, preventing convergence. Options for forcing the convergence include cooling schedules for classification, or locking classifications after they have been made. In our implementation, the classification for a measurement is considered locked if the number of times it has been classified an outlier exceeds a preconfigured threshold. This approach was chosen to account for poor initial guesses, but also to save on processing time for definite outliers.

5. Experiments

In this section, we detail the experiments conducted to characterize the HMOR framework presented in this paper. These experiments were essential to inform the design decisions necessary in constructing the entire surface mapping framework described in [7]. To unit test the outlier rejection component apart from the rest of the mapping framework, we required ground truth data for the rover poses, feature positions, and measurement classifications. Unfortunately, this could not be conducted in natural, unstructured terrain because accurate feature positions were not available. As a result, the datasets utilized for characterization were collected from simulation and experiments conducted in a controlled indoor lab environment where this type of data could be easily extracted. Overall mapping performance in outdoor terrain can be found in [7].

We begin an overview of our controlled mapping scenario, including the measurement models employed, and a summary of the classification approaches compared in this section. This is followed by a brief discussion of the metrics used to evaluate and compare performance. With these metrics, we then introduce the large number of simulations conducted to analyze statistical performance, and finally, we validate our framework by demonstrating its utility using real-world data obtained through hardware experiments.

5.1. Overview

As presented in the introduction, the planetary mapping task presents a scenario where feature measurements are scarce, and odometry measurements are available. For these experiments, we considered odometry measurements to be composed of translations and rotations between rover poses, and feature measurements to be 3D positions of observed point features. Outliers were present in both measurement types, with wheel slippage for odometry, and incorrect data associations for the features. These measurement models are common in the feature-based SLAM literature, and reflects the measurement heterogeneity of the planetary mapping scenario.

The batch alignment was computed using a variation of MOGA [8], where the dependence on a prior orbital map was removed [7]. This is similar to the GraphSLAM [10] approach of incorporating odometry and feature measurements into a Bundle Adjustment problem, but extended to 3D with our rotation parameterization [23].

The classification approaches compared are listed as follows:

1. No outlier rejection.
2. Exhaustive pairwise RANSAC for the feature measurements only.
4. Iterative reclassification using the Independent Innovation Test (OC-I).
5. Iterative reclassification using the Batch Innovation Test (OC-B).
6. Iterative reclassification using Batch Inlier Classification (IC-B).
7. Perfect classification based on ground truth.

The first approach conducted no outlier rejection, and simply computed the alignment based on all of the data available. This provided a worst-case bound on the performances. The state-of-the-art was represented by conducting pairwise RANSAC between frames to reject measurement outliers. To avoid the stochasticity of the algorithm and due to the low number of measurements, we performed RANSAC exhaustively.

For our proposed framework, all four classification methods were implemented in the iterative reclassification scheme using the RANSAC alignment as our initial hypothesis. For simplicity, the probability threshold employed in the statistical tests was set to the value corresponding to $3\sigma$ of a one-dimensional Gaussian distribution. Though performance can likely be improved through tuning, retaining the commonly used value allowed for easy and direct comparison.

Finally, as the performance of an estimator largely depends on the measurements available, the severity of measurement noise, and the models used, the alignment obtained through perfect measurement classification was also computed. This served as a benchmark for the comparisons, as it represented the best performance achievable for the given set of measurements and alignment algorithm.

The classification ground truth was obtained by computing the measurement errors based on the ground truth rover poses and feature positions. Errors that exceeded the $3\sigma$ threshold of the measurement noise model were classified as outliers. While this hard threshold does not reflect the actual tail of a Gaussian distribution, very few inliers should exceed its limits.

### 5.2. Metrics

To compare the relative performance of the algorithms, a set of common metrics must be defined to allow for quantitative analysis. Since we are concerned with the outlier classification task, one good measure of performance is classification accuracy. We define false positives as inliers misclassified as outliers, and false negatives as outliers misclassified as inliers. However, simply counting and comparing these values do not provide a complete picture of the scenario at hand. It can be argued that false negatives are considerably worse than false positives, since the presence of outliers tends to have a catastrophic effect on the alignment. However, depending on the severity of the outlier, a sufficient number of inliers can counteract its effects.

Since incorrect classifications have varied effects on the alignment, we also consider the alignment accuracy itself. By analyzing the end result, the effects of the misclassifications are incorporated into the metric. For estimate accuracy, rather than computing the difference in alignment between the estimated and ground truth values in a single global reference frame, we utilized a relative measure instead. This is motivated by the fact that the SLAM problem is unobservable; there is an unknown translation/rotation with respect to a global reference frame. Without any measurements shared between the ground truth system and rover measurements, it is difficult to justify the use of a single privileged frame for evaluating the accuracy [27,28].

As a result, we define the distance mismatch as the translation error, and utilize the axis-angle representation of the rotation difference for the rotational error. That is, the relative rover position estimate error, $\hat{\mathbf{e}}_{p_{k_2 \rightarrow k_1}}$, and the relative rover orientation estimate error, $\hat{\mathbf{e}}_{q_{k_2 \rightarrow k_1}}$, between poses $k_1$ and $k_2$ can be expressed as

$$\hat{\mathbf{e}}_{p_{k_2 \rightarrow k_1}} := \mathbf{P}_{k_1} - \mathbf{P}_{k_2},$$

$$\hat{\mathbf{e}}_{q_{k_2 \rightarrow k_1}} := \mathbf{C}_{k_2 \rightarrow k_1} - \mathbf{C}_{k_1 \rightarrow k_1},$$

where $\mathbf{P}_{k_1}$ is the translation between frame $k_1$ to $k_2$ expressed in frame $k_1$, $\mathbf{C}_{k_2 \rightarrow k_1}$ is the rotation from frame $k_1$ to $k_2$, and $(\cdot)$ denotes the estimated values. The transformation from a rotation matrix to the axis-angle representation is indicated by the $\leftarrow$ symbol [29]. Similarly, the estimation error for each feature position is computed from its relative position for the poses from which it is observed. That is, for each feature $i$ observed in frame $k$, the relative feature position estimate error is

$$\hat{\mathbf{e}}_{p_i} := \mathbf{p}_k - \hat{\mathbf{p}}_k,$$

where $\mathbf{p}_k$ is the position of the $i$th landmark expressed in frame $k$. With these definitions, the metric we use for alignment accuracy is the root-mean-squared (RMS) error over all dimensions of the relative transformations between all pairs of poses, and for all poses that observe a feature. These metrics can be expressed as

$$\hat{E}_{\text{trans}} := \sqrt{\frac{\sum_{k_1=1}^{K} \sum_{k_2=k_1+1}^{K} \| \hat{\mathbf{e}}_{p_{k_2 \rightarrow k_1}} \|^2}{3M_p}},$$

$$\hat{E}_{\text{rot}} := \sqrt{\frac{\sum_{k_1=1}^{K} \sum_{k_2=k_1+1}^{K} \| \hat{\mathbf{e}}_{q_{k_2 \rightarrow k_1}} \|^2}{3M_p}},$$

$$\hat{E}_{\text{feat}} := \sqrt{\frac{\sum_{k=1}^{K} \sum_{i=1}^{N_k} \| \mathbf{e}_{p_i} \|^2}{\sum_{i=1}^{N_k} N_i}},$$

where $K$ is the number of rover poses, $M_p := \binom{K}{2}$, and $N_i$ is the number of times feature $i$ was observed. It should be noted that we divide by 3 because there are three dimensions each for translation, rotation, and position. This metric identifies localization and mapping consistency, rather than alignment accuracy to an unknown global frame.
5.3. Simulations

We began with a large number of 3D simulated rover trials to provide preliminary validation and statistical analysis. The 10,000 trials consisted of ten rover poses each in simple loop traverses, with a low feature count of ten to ensure measurement scarcity. A typical layout of a simulation trial is depicted in Fig. 7.

Table 1 provides a summary of the classification performance of the algorithms. False positive and false negative rates were also computed by dividing the number of false classifications by the number of outliers. These values give an indication of the algorithmic performance in relation to the corruption rates. Of the 10,000 odometry measurements, 977 were outliers, and for the feature measurements, there were 2,343 outliers in 45,542 measurements. These resulted in an overall corruption percentage of 9.8% and 5.1% for the odometry and feature observations, respectively. While the overall corruption rates were low, due to the scarcity of measurements, the influence of each outlier was very high.

We see that there was a marked improvement in using the iterative methods over standard RANSAC, not only in that odometry measurement classifications were considered, but also in that there were more correct classifications over all. Due to the nonlinearities of the system and the fact that a perfect hypothesis model cannot be obtained from noisy data, the classifications were still imperfect. However, the results reflected the spectrum of accuracy as suggested by Fig. 4.

OC-S, with the smallest threshold, correctly rejected the most outliers, but also misclassified many inliers. OC-I included many more measurements, which resulted in a reduced number of inlier misclassifications, but then outliers were included as well. Finally, OC-B and IC-B had the best classification performance, with a reduced number in both categories. The difference in performance between the two batch methods can be attributed to the accumulation of linearization errors when building up the classification sets from either direction.

The effects of the misclassifications are summarized by the average alignment accuracies presented in Table 2. Though the alignment performance in these cases was fairly poor, when compared to the accuracy obtained by the perfect classification, we see that the data available did not lead to much better performance. The pairwise RANSAC algorithm provides a small improvement over no outlier rejection, but the iterative framework greatly outperformed RANSAC alone. Once again, the spectrum of accuracy is reflected in the results, with OC-S performing the worst of the three approaches, and OC-B the best. Interestingly, only a minor improvement was offered in using the batch methods compared to OC-I, and thus may only be necessary in certain applications.

![Fig. 7. A typical layout for a simulation trial, with the rover poses represented by the red triangles, landmarks by the green diamonds, and odometry connectivity by the red arrows. Two loops were taken, with measurements obtained at infrequent stops. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)](attachment:image.png)

<table>
<thead>
<tr>
<th>Table 1</th>
<th>Classification accuracy for the simulated traverses.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Odometry</td>
</tr>
<tr>
<td></td>
<td>False positive count/rate (%)</td>
</tr>
<tr>
<td></td>
<td>False negative count/rate (%)</td>
</tr>
<tr>
<td>No rejection</td>
<td>0/0</td>
</tr>
<tr>
<td>RANSAC</td>
<td>0/0</td>
</tr>
<tr>
<td>OC-S</td>
<td>631/64.6</td>
</tr>
<tr>
<td>OC-I</td>
<td>416/42.6</td>
</tr>
<tr>
<td>OC-B</td>
<td>407/41.7</td>
</tr>
<tr>
<td>IC-B</td>
<td>418/42.8</td>
</tr>
<tr>
<td>Perfect</td>
<td>0/0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 2</th>
<th>Average relative RMS alignment accuracy for the simulated traverses.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Rover position error (m)</td>
</tr>
<tr>
<td></td>
<td>Mean</td>
</tr>
<tr>
<td>No rejection</td>
<td>0.91</td>
</tr>
<tr>
<td>RANSAC</td>
<td>0.86</td>
</tr>
<tr>
<td>OC-S</td>
<td>0.64</td>
</tr>
<tr>
<td>OC-I</td>
<td>0.59</td>
</tr>
<tr>
<td>OC-B</td>
<td>0.55</td>
</tr>
<tr>
<td>IC-B</td>
<td>0.55</td>
</tr>
<tr>
<td>Perfect</td>
<td>0.35</td>
</tr>
</tbody>
</table>
iterations. Additionally, if we consider the fact that the average time required to compute the initial hypothesis models was 4.6 s, it appears that the majority of the computation time for the iterative methods came from the alignment algorithm itself. Therefore, we can conclude that the additional computational costs required were not as severe as expected.

The standard deviation of the error over all of the traverses remained much the same, indicating an overall positive shift in the performance. In the cases where many outliers were present, it was observed that the OC-I algorithm performed slightly better than the batch methods. This can be attributed to the fact that the batch methods consider groups of measurements in their classification criteria, and as a result, are susceptible to high corruption rates. It was observed that the weaker assumptions made by OC-I provided more robustness to large numbers of outliers. However, the overall best performance was displayed by the OC-B algorithm. Compared to RANSAC, the rover position error was reduced on average by 36%, the orientation error by 33%, and the feature position error by 36%.

5.4. Hardware experiments

Further characterization was also conducted through real-world hardware experimentation, to support the statistical analysis obtained through simulation. The planetary mapping scenario was emulated by manually driving a modified P3-AT rover in our controlled indoor lab environment, where easily-identified plastic tubes served as the sparse features composing the map. Images depicting our experimental setup are provided in Fig. 8.

The 3D odometry measurements were produced by a combination of the wheel encoders and an inclinometer. Though the workspace was fairly open, the traverses were varied to provide a large range of measurements, and to emulate real-world usage. Sand was also introduced in specific locations to encourage catastrophic wheel slip--page outliers when desired.

The 3D feature position measurements were obtained by identifying the plastic tubes in camera imagery. A panning unit provided bearing information, and a known length on the tubes was used to determine scale for computing the range values. The unique barcodes on each landmark provided feature identification, which simplified the data association problem. Though we employed a camera for the feature measurements, it was utilized infrequently to emulate the scenario of interest. Feature measurement outliers were generated by conducting random misassociations.

Ground truth data was provided by a Vicon motion capture system, which was used to track retroreflective markers placed on the rover and the landmarks. Since the motion capture system was able to provide the position of each marker to within a few millimetres, it served as an appropriate ground truth reference for the hardware experiments. Laser scans were also obtained at each rover pose for visualization purposes.

The use of this indoor setup ensured that we had accurate ground truth for the rover poses, landmark positions, and feature measurement data associations. This allowed us to unit test the outlier rejection methods comprehensively, controlling not only the environmental conditions, but also the occurrence and severity of the outlier measurements.

Ten different traverses were conducted using this hardware, where we varied the number of landmarks in the scene, the number of rover stops, and the severity of the outliers. For each of the ten traverses, sets of random feature misassociations were imposed, producing 20 variations per hardware trial. In total, this resulted in 200 runs for this hardware dataset. With true non-Gaussian noise and possible unmodelled nonlinearities, this dataset provided validation on the robustness of the algorithms, and their real-world applicability. To highlight and compare the performances of the algorithms, we will first provide analysis of a typical hardware traverse, and then provide some overall statistics for the 200 runs.

5.4.1. Typical traverse

Fig. 9 illustrates one of the hardware traverses conducted, with 10 rover poses and 8 features in the scene. Though the experiments were conducted in 3D, the plots depict an overhead view for ease of visualization.

![Fig. 8. The equipment used for the hardware trials. Odometry measurements were provided by the rover base, and a camera was used to produce feature measurements to plastic tubes with barcode markings. Vicon markers placed on the landmarks and rover served as ground truth for the experiments. (a) Modified P3-AT base and (b) a sample experimental setup.](image-url)
Furthermore, for additional clarity, the odometry measurements and feature observations are illustrated separately, in Fig. 9(a) and (b), respectively.

Five outliers were experienced during this traverse. Three odometry outliers were present due to travel over the sand pile, with one significantly less severe than the other two. The remaining two feature measurement outliers were caused by incorrect data associations. As can be seen, a complex network of measurements can easily arise even in simple scenarios.

The classification performance for this traverse is summarized in Table 3. As can be seen, none of the outliers were detected by RANSAC due to the minimum overlap constraint. In comparison, all of the iterative algorithms were able to correctly identify two of the three odometry outliers, with the false negative corresponding to the outlier with the low severity. The differences in feature outlier classification performance reflect Fig. 4, where we visually compared our proposed statistical tests. OC-S correctly identified all of the outliers, but produced one false positive due to its smaller classification threshold. On the other hand, the OC-I algorithm utilized too large of a threshold, which resulted in a false negative. Finally, both batch methods correctly classified all of the feature measurements.

The alignment accuracy for this traverse is summarized in Table 4, where we see the same trend in accuracy as suggested by the classification performance. The iterative methods provided a large reduction in the error values, and the minor difference between the batch methods and the perfect classification alignment confirmed the expectation that the odometry outlier misclassification would not have a significant effect.

![Fig. 9. Illustrations of the measurements for the typical hardware traverse, where the rover poses, landmarks, and measurements are represented by red triangles, green diamonds, and arrows, respectively. Outlier measurements are indicated by the dashed lines. The odometry outliers were caused by travelling over the sand pile, and the feature measurement outliers resulted of incorrect data associations. These outliers are undetectable by standard pairwise methods due to the insufficient feature observation overlap. (a) Odometry measurements and (b) feature observations.](image)

### Table 3

<table>
<thead>
<tr>
<th></th>
<th>Odometry</th>
<th>Feature observations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>False positive count/rate (%)</td>
<td>False negative count/rate (%)</td>
</tr>
<tr>
<td>No rejection</td>
<td>0/0</td>
<td>3/100</td>
</tr>
<tr>
<td>RANSAC</td>
<td>0/0</td>
<td>3/100</td>
</tr>
<tr>
<td>OC-S</td>
<td>0/0</td>
<td>1/33.3</td>
</tr>
<tr>
<td>OC-I</td>
<td>0/0</td>
<td>1/33.3</td>
</tr>
<tr>
<td>OC-B</td>
<td>0/0</td>
<td>1/33.3</td>
</tr>
<tr>
<td>IC-B</td>
<td>0/0</td>
<td>1/33.3</td>
</tr>
<tr>
<td>Perfect</td>
<td>0/0</td>
<td>0/0</td>
</tr>
</tbody>
</table>

Finally, qualitative analysis is provided by plotting the results of the alignment algorithms. Since there was an unknown transformation between the estimate and ground truth reference frames, we performed an ordinary-least-squares alignment between the ground truth and estimated feature positions. This additional alignment step allowed the maps to be visually compared. Fig. 10 depicts the mean estimates of the rover positions and feature locations against the ground truth values.

These plots illustrate the value of the proposed algorithms over the conventional approach. Due to the inability of RANSAC to identify outliers in our scenario, the resulting estimate was poor. In comparison, the iterative algorithms performed significantly better, with OC-S resembling the true locations, and further refinement demonstrated by OC-I and OC-B/IC-B as the classification accuracies improved.
In addition to these overhead diagrams, we utilized the point cloud data gathered at each rover pose during the traverse to generate 3D renders of the indoor lab environment for additional qualitative analysis. Fig. 11 provides a comparison between the resulting maps based on the estimates provided by RANSAC and OC-B/IC-B. These maps were produced by overlaying the point clouds according to the rover pose estimates. As can be seen, the outliers present in the RANSAC estimates resulted in very poor mapping performance. On the other hand, the map associated to the OC-B/IC-B estimates is significantly better, capturing distinctive visual elements such as the straight walls and flat floor of the indoor workspace.

5.4.2. Overall performance

The overall classification performance on all of the hardware runs was very similar to the simulated results. The dataset consisted of 280 outliers in the 2000 odometry measurements, and 482 outliers in the 8540 feature measurements. This resulted in an overall corruption rate of 14% and 5.6% for the odometry and features, respectively. The classification performance is summarized in Table 5.

Once again, the iterative methods outperformed RANSAC, and the batch methods produced the best classification performance. The spectrum of accuracy suggested by Fig. 4 was maintained, and the alignment accuracy, summarized in Table 6, provided similar results as well.

The average computation time required (using the same computer as the time complexity simulations) for RANSAC was 2.2 s, 3.5 s for OC-S, 5.0 s for OC-I, 4.0 s for OC-B, and 4.1 s for IC-B. Interestingly, these timing results do not follow the expectation that the batch methods require the most processing time. However, due to the

| Table 4
| Relative RMS alignment accuracy for the typical hardware traverse. |
|---|---|---|
| | \( \hat{\epsilon}_{\text{trans}} \) (m) | \( \hat{\epsilon}_{\text{rot}} \) (rads) | \( \hat{\epsilon}_{\text{feat}} \) (m) |
| No rejection | 1.08 | 0.67 | 0.27 |
| RANSAC | 1.08 | 0.67 | 0.27 |
| OC-S | 0.51 | 0.27 | 0.14 |
| OC-I | 0.31 | 0.11 | 0.11 |
| OC-B | 0.23 | 0.09 | 0.08 |
| IC-B | 0.23 | 0.09 | 0.08 |
| Perfect | 0.21 | 0.08 | 0.06 |

Fig. 10. Alignment results for the typical hardware traverse, where feature positions are marked by crosses, rover poses by dots, and straight lines to connect successive poses. Black is used for ground truth values, and red for estimates. Since no outliers were identified by RANSAC, the alignment result was the same as no-outlier-rejection. In addition, the batch methods are paired because they produced the same classifications for this traverse. These plots illustrate the range in alignment accuracy for the proposed classification algorithms incorporated into the iterative reclassification framework. (a) RANSAC, (b) OC-S, (c) OC-I, and (d) OC-B/IC-B. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)

Fig. 11. 3D renders of the indoor lab environment generated using the data gathered during the typical hardware traverse. Point clouds obtained at each rover pose were overlaid according to their estimated values, where each colour corresponds to a different rover pose. As can be seen, the RANSAC estimates resulted in poor mapping performance, while OC-B/IC-B accurately captured environmental elements such as the straight walls. (a) RANSAC and (b) OC-B/IC-B. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)
A number of novel contributions have been made in this paper to address this problem, which are summarized as follows:

1. The identification of a common probabilistic form for the measurements (1), which allows for the use of the Mahalanobis distance (5) as a metric to facilitate comparison and classification of heterogeneous measurement types in the HMOR problem.

2. The formulation of a batch measurement classification criterion (18) that retains the underlying measurement independence. This pairs naturally with batch alignment.

3. The incorporation of the proposed OC-I, OC-B, and IC-B classification tests into an iterative framework, creating a robust estimator.

In addition, it should be worth restating that in the SLAM scenario, this is the first algorithm that is able to handle outliers in odometry measurements.

This work was conducted to inform design decisions during the development of a complete robotic surface mapping framework [7]. Characterization was performed through simulation and hardware experimentation, which demonstrated that our proposed approach provided a considerable improvement in performance over the current state-of-the-art. The four classification algorithms developed in this paper present a tradeoff between computational speed and accuracy, which can be chosen appropriately for the scenario at hand.

As shown in Fig. 2, the HMOR methods provided robustness towards outlier measurements, producing an accurate surface map of the terrain [7]. As a result of the experimental characterization presented in this paper, we decided to utilize the OC-I algorithm during iterative reclassification in our implementation. While the batch methods produced more accurate alignments in the experiments, the improvement in performance was minor. This was outweighed by the increased computational requirements in the full-scale mapping scenario, and the reduced robustness of the batch methods to large numbers of outliers.

Some possibilities for future work include improving the performance by utilizing a more accurate noise propagation method, using additional information to generate better hypothesis models, and the formulation of a more efficient batch classification test that can reject more than one outlier per iteration and that is able to handle larger outlier corruption rates.

### Acknowledgements

The authors would like to thank Trevor Campbell, Konstantine Tsotsos, Yoni Halpern, and Keith Leung for their assistance in developing and testing the middleware used to control and track the rover in the hardware experiments. This work was supported by a Natural Sciences and Engineering Research Council of Canada Collaborative Research and Development (NSERC CRD)/

### Table 5

<table>
<thead>
<tr>
<th>Odometry</th>
<th>Feature observations</th>
</tr>
</thead>
<tbody>
<tr>
<td>False positive count/rate (%)</td>
<td>False negative count/rate (%)</td>
</tr>
<tr>
<td>False positive count/rate (%)</td>
<td>False negative count/rate (%)</td>
</tr>
<tr>
<td><strong>No rejection</strong></td>
<td><strong>RANSAC</strong></td>
</tr>
<tr>
<td>0/0</td>
<td>0/0</td>
</tr>
<tr>
<td>280/100</td>
<td>280/100</td>
</tr>
<tr>
<td><strong>OC-S</strong></td>
<td><strong>OC-I</strong></td>
</tr>
<tr>
<td>93/33.2</td>
<td>39/13.9</td>
</tr>
<tr>
<td>91/32.5</td>
<td>106/37.9</td>
</tr>
<tr>
<td><strong>OC-B</strong></td>
<td><strong>IC-B</strong></td>
</tr>
<tr>
<td>16/5.7</td>
<td>13/4.6</td>
</tr>
<tr>
<td>98/35.0</td>
<td>101/36.1</td>
</tr>
<tr>
<td><strong>Perfect</strong></td>
<td><strong>Perfect</strong></td>
</tr>
<tr>
<td>0/0</td>
<td>0/0</td>
</tr>
<tr>
<td>0/0</td>
<td>0/0</td>
</tr>
</tbody>
</table>

### Table 6

Average relative RMS alignment accuracy for the hardware traverses.

<table>
<thead>
<tr>
<th>Rover position</th>
<th>Feature position</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\ell_{\text{trans}}$</td>
<td>$\ell_{\text{rot}}$</td>
</tr>
<tr>
<td>Mean &amp; Std Dev</td>
<td>Mean &amp; Std Dev</td>
</tr>
<tr>
<td><strong>No rejection</strong></td>
<td><strong>RANSAC</strong></td>
</tr>
<tr>
<td>0.69 &amp; 0.44</td>
<td>0.54 &amp; 0.34</td>
</tr>
<tr>
<td>0.29 &amp; 0.21</td>
<td>0.23 &amp; 0.19</td>
</tr>
<tr>
<td>0.32 &amp; 0.13</td>
<td>0.36 &amp; 0.34</td>
</tr>
</tbody>
</table>

stability of classifications, the batch methods required less iterations than OC-I. The overall mean error improvement for OC-B when compared to RANSAC, was 57% for the rover positions, 65% for the orientations, and 55% for the feature positions.
Canadian Space Agency Partnership Support Program (CSA PSP) grant in cooperation with MDA Space Missions.

References


