Robust Sensor Characterization via Max-Mixture Models: GPS Sensors

Ryan Morton and Edwin Olson

Abstract— Large position errors plague GNSS-based sensors (e.g., GPS) due to poor satellite configuration and multipath effects, resulting in frequent outliers. Due to quadratic cost functions when optimizing SLAM via nonlinear least square methods, a single such outlier can cause severe map distortions. Following in the footsteps of recent improvements in the robustness of SLAM optimization process, this work presents a framework for improving sensor noise characterizations by combining a machine learning approach with max-mixture error models. By using max-mixtures, the sensor's noise distribution can be modeled to a desired accuracy, with robustness to outliers. We apply the framework to the task of accurately modeling the uncertainties of consumer-grade GPS sensors. Our method estimates the observation covariances using only weighted feature vectors and a single max operator, learning parameters off-line for efficient on-line calculation.

I. INTRODUCTION

Most state-of-the-art Simultaneous Localization and Mapping (SLAM) algorithms require sensor uncertainties to be characterized probabilistically. Such characterization is a straightforward task for many common robot sensors, e.g., LIDARs. However, the noise distribution of other sensors provide a much more challenging characterization task. The cause of these erroneous observations stems from the inability of the sensor to observe the complete state of the environment. For example, ground robots' wheels lose traction resulting in a slip-or-grip problem and vertical structures block Global Positioning System (GPS) signals, leading to multipath effects. Generally, the sensor return is assumed to reflect the most likely state of the quantity being measured, e.g., latitude and longitude for a GPS sensor[1]. The task of defining the uncertainty of these individual observations is termed sensor characterization.

The typical approach to sensor characterization uses Maximum Likelihood (ML) optimization to find the parameters that best describe the training data. This off-line process returns parameters that maximize the average performance over ground-truthed training data. These parameters can then be used on-line to calculate observation noise estimates for use within SLAM.

A single outlier can cause a severe, even irrecoverable, map distortion due to the quadratic cost surface used by the nonlinear least squares optimization within many SLAM algorithms. In this work, we take the view that outliers arise from overly-simple and optimistic noise models; a better noise model would assign a higher probability to
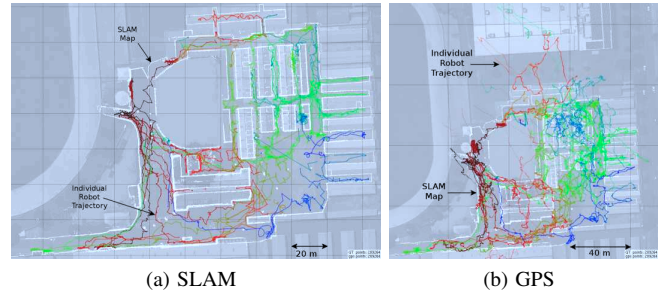
The authors are with the Computer Science and Engineering, University of Michigan, 2260 Hayward Street, Ann Arbor, Michigan, 48109, USA. {rmorton, ebolson}@umich.edu.

[1]In this work GPS means any Global Navigation Satellite System (GNSS)

(a) SLAM   (b) GPS

Fig. 1: Robot Trajectories for 12 Exploration Robots. Colored to reflect elapsing time (red-green-blue as t increases).

an "outlier", thus limiting its effect. Simple Gaussian error models often suffer from this problem, since the tails drop off exponentially fast. We use max-mixtures of Gaussians to improve the richness of the sensor characterization models.

To highlight our framework, we show noise models for GPS sensors, which are particularly prone to large errors; it is common to receive observations from consumer grade sensors that are tens to hundreds of meters from the true location. Specifically, the contributions of this paper include:

- A machine-learning approach for sensor uncertainty estimation.
- An extension of sensor uncertainty estimation using max-mixtures.
- A new metric for evaluating the impact of outliers on a SLAM system.
- Evaluation of a variety of models for consumer-grade GPS sensor characterization.

In the next section, we give a short overview of related advancements in SLAM, robust estimation, and GPS error characterization. We describe the mathematical formulation of sensor characterization in Section III and show the robustness improvements of the max-mixture models in Section IV. Then in Section V, we describe features from GPS sensor data and how to combine them to robustly estimate GPS uncertainty. In Section VI, we show empirical results for the noise models on a real-world dataset comprised of 45 hours of GPS data from a 14-robot team.

II. BACKGROUND

Many modern approaches formulate SLAM in terms of two components: a front-end that builds a factor graph and a back-end that optimizes it [1]. For example, the front-end might add an odometry constraint (edge) representing a rigid-body-transformation between two sequential robot positions

(nodes); the constraint's mean represents the raw encoder observation while the edge weight is a function of the a priori uncertainty of the encoder. The back-end periodically optimizes the graph, producing a ML estimate for all nodes in the graph, i.e., the best position of the nodes given all the constraints. When errors are Gaussian this optimization can be viewed as a least squares method with each constraint incurring a quadratic cost.

A known problem in SLAM and other nonlinear least squares optimization problems stems from these quadratic costs, namely that a single outlier can have significant and detrimental impact on the solution [2], [3], [4]. The quadratic costs come directly from assuming constraints have Gaussian error, which brings computational efficiency.

Improving the robustness of SLAM to these outliers is currently an active area of research. Some approaches use switching variables, which can 'turn off' constraints and, thus, handle outliers by modifying the graph structure in the back-end [2], [5]. Another related approach detects and rejects outliers via loop consistency checks on small clusters of nodes [6]. Both of these approaches show promise at *dealing with outliers* in the back-end of SLAM. In this paper, we take the view that outliers arise from modeling errors and that richer error models can result in systems that are robust to ouliers and do not require specialized "outlier rejection" methods.

While mixture models have long been used to approximate complex distributions, they typically result in high computational costs. Our system uses max-mixtures, which allows both flexible error models and fast inference [4]. Some sensors could be characterized with robust cost functions, [7], [8], but these have been previously shown to be subsumed by max-mixtures [4].

This paper focuses on models for GPS data, which can be used both to improve the accuracy of the map and to register the relative frame to Earth. Even with many loop closures, SLAM maps can have significant distortion compared to ground truth, e.g., long hallways may erroneously bend and GPS sensors can offer the needed constraints even when global registration is not necessarily needed [9]. Unmanned Aerial Vehicles (UAVs) and agricultural robots, which generally have unobstructed satellite line-of-sight, can rely heavily upon GPS sensors for positioning. In environments with an unobstructed view of the sky, GPS sensors offer cost-effective and computationally efficient position estimates that are, when supplemented with other sensors, sufficient for many systems [10]. Additional sensors, such as inertial measurement units or visual odometry have been shown to reduce errors caused by short blackouts in GPS data, i.e., obstruction of satellite view [11], [12], [13].

Early error models for GPS assumed a constant variance for all GPS returns [11], [14]. Many GPS sensors provide their own uncertainty estimates, but they can be misleading. For example, they may report erroneous readings for a period of time before detecting the loss of satellite locks. This problem stems from the discrete nature of individual satellite locks that can cause position estimate discontinuities [15], a

problem that is blamed for a crash during the DARPA Grand Challenge [16].

Rather than accurately modeling the noise of GPS sensors, a common approach has been binary classification of (in)valid sensor returns [13], [17], [16]. These systems stop adding GPS data to SLAM upon detection of a GPS dropout and/or multipath effects. Some approaches use additional sensors to improve multipath detection via satellite line-of-sight calculations by building 3D-models of the local structure [18], [19].

Some approaches circumvent the GPS sensor's position estimate and calculate positions directly from the trilateration of pseudo-ranges to individual satellites [20]. The most closely related work uses robust cost function and switch variables to modify the graph, but their method also requires a GPS sensor with pseudo-range capabilities [21]. Although these approaches using sensors equipped with pseudo-ranges and/or sensor utilizing the local structure have shown signs of success, we desire a robust approach that uses observations directly from standard consumer-grade GPS sensors.

III. Sensor Characterization

Sensor characterization consists of estimating the distribution of observations, $\mathbf{z}_i$, returned from the sensor around the true location, $\mathbf{x}_i$. The goal of sensor characterization is a distribution best defining the zero-mean $P(\mathbf{z}_i|\mathbf{x}_i)$. Designers generally choose the form of the distribution, e.g., Gaussian, and estimate the parameters of the model using training data, or by hand-tuning, to maximize log-likelihood. The output of the characterization is a covariance matrix $\mathbf{\Sigma}_i$, which may be constant or a function of the observation.

A. Simple Gaussian Models

Let $\mathbf{e}_i$ be the observation error, i.e., $\mathbf{e}_i = (\mathbf{z}_i - \mathbf{x}_i)$. Assuming a zero-mean uni-modal Gaussian distribution, parametrized by $\mathbf{\Sigma}_i$, the distribution becomes:

$$P(\mathbf{e}_i) = \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_i) = \frac{1}{(2\pi)^{\frac{d}{2}}|\mathbf{\Sigma}_i|^{\frac{1}{2}}} e^{-\frac{1}{2}\mathbf{e}_i^T\mathbf{\Sigma}_i^{-1}\mathbf{e}_i} \quad (1)$$

with d equal to the degree-of-freedom (DOF) of the observation. The log-likelihood, assuming n independent observations, can be written:

$$L = \log \prod_{i=0}^{n-1} P(\mathbf{e}_i) = \sum_{i=0}^{n-1} \log P(\mathbf{e}_i)$$

$$= -\frac{nd}{2}\log(2\pi) - \frac{1}{2}\sum_{i=0}^{n-1} \log(|\mathbf{\Sigma}_i|) + \mathbf{e}_i^T\mathbf{\Sigma}_i^{-1}\mathbf{e}_i \quad (2)$$

The last term in (2) explicitly shows the quadratic costs associated with deviations from the mean.

The sensor characterization task is the definition of $\mathbf{\Sigma}_i$, a function of individual observations, to maximize L. Each such function can take many forms, but we use a parametrization where each element is a linear combination of features of the observation. For example, assuming a 2 DOF sensor

with independent noise in each DOF, the covariance may be defined as:

$$\mathbf{\Sigma}_i = \begin{bmatrix} \sigma_{i,x}^2 & 0 \\ 0 & \sigma_{i,y}^2 \end{bmatrix} = \begin{bmatrix} (f_{i,x}^T w_x)^2 & 0 \\ 0 & (f_{i,y}^T w_y)^2 \end{bmatrix} \quad (3)$$

With feature vectors $f_{i,x}$ and $f_{i,y}$ as functions of each observation (presumably informing about the error in x and y respectively) and global weight vectors w_x and w_y. In this context, the characterization task would simply be the definition of the weight vectors.

IV. ROBUST SENSOR CHARACTERIZATION USING MAX-MIXTURES OF GAUSSIANS

Gaussian mixture models offer a richer representation by combining multiple Gaussian components, allowing modeling of arbitrarily complex distributions. Max-mixture models define the distribution as the max of k weighted components:

$$P(\mathbf{e}_i) = \eta_i \max(\alpha_1 P_1(\mathbf{e}_i), \ldots, \alpha_k P_k(\mathbf{e}_i))$$
$$= \eta_i \max_{j=1}^{k} \left(\frac{\alpha_j}{(2\pi)^{\frac{d}{2}} |\mathbf{\Sigma}_{i,j}|^{\frac{1}{2}}} e^{-\frac{1}{2}\mathbf{e}_i^T \mathbf{\Sigma}_{i,j}^{-1} \mathbf{e}_i} \right) \quad (4)$$

Each components, P_j, is of the form in (1). The log-likelihood of the data becomes:

$$L = \log \prod_i P(\mathbf{e}_i) = \sum_i \log P(\mathbf{e}_i)$$
$$= \sum_i \log \left(\frac{\eta_i}{(2\pi)^{\frac{d}{2}}} \right) + \max_{j=1}^{k} \left(\log \left(\frac{\alpha_j e^{-\frac{1}{2}\mathbf{e}_i^T \mathbf{\Sigma}_{i,j}^{-1}\mathbf{e}_i}}{|\mathbf{\Sigma}_{i,j}|^{\frac{1}{2}}} \right) \right) \quad (5)$$

Note that η_i becomes an additive constant that does not effect the minimization and does not generally need to be computed. The sensor characterization task for max-mixtures is to define the covariance estimates $\mathbf{\Sigma}_{i,j}$ and mixing terms α_j that maximize L.

The most common mixture models are sum-mixtures, i.e., $P(\mathbf{e}_i) = \sum_{j=1}^{k} \alpha_j P_j(\mathbf{e}_i)$, but our choice of max-mixtures is motivated by the use of logarithms in (5). The ability to push the logarithm inside the max operator brings significant computational advantages [4].

V. GPS SENSOR CHARACTERIZATION

We consider two families of GPS uncertainty predictors based on uni-modal Gaussians and max-mixtures of Gaussians using non-linear optimization for parameter fitting. The primary difference between these families is that, for each observation, uni-modal models have a single covariance estimate, $\mathbf{\Sigma}_i$, and max-mixture models have k of them; plus k weight vectors.

For mobile ground robots, we assume radial GPS errors, with x and y errors independent and identically distributed, i.e., $e_i = ||\mathbf{e}_i||$ represents the total translation error. Thus, $f_i = f_{i,x} = f_{i,y}$, $w = w_x = w_y$, and each covariance function becomes:

$$\mathbf{\Sigma}_i = \begin{bmatrix} \sigma_i^2 & 0 \\ 0 & \sigma_i^2 \end{bmatrix} = \begin{bmatrix} (f_i^T w)^2 & 0 \\ 0 & (f_i^T w)^2 \end{bmatrix} \quad (6)$$

The global weights are learned off-line and we next describe some possible GPS feature vectors f_i for each observation z_i.

A. GPS Observation Features

An interesting property of GPS receivers is that they produce a wealth of data that can be used to generate features. In this section, we explore a variety of features, beginning with trivial (but still useful) ones and moving to more complex features.

1) Constant Noise Model: While few real-world systems today would attempt to characterize all GPS observations as having the same uncertainty, such a model can serve as a baseline method for evaluation. Using our framework, we simply let

$$f_i^T = [\ 1\ ]$$

Using this model, the magnitude of the learned weight would represent the standard deviation for all observations. Since all observations have equal feature vectors, f_i, all observations will have the same noise estimate $f_i^T w$.

2) Number of Satellites Noise Model: One strategy for determining the reliability of GPS data is to observe the number of visible satellites, $n_{sat,i}$. A simple feature vector expressing this assumption is given by:

$$f_i^T = [\ n_{sat,i}\ ]$$

We expect to learn a negative weight for this feature since more satellites should reduce the observation error. However, this feature cannot be used alone, because negative σs are prohibited. Later we'll see how these simple feature can be combined with others.

A more effective use of the number of satellites is to construct the feature vector f_i such that the $n_{sat,i}^{th}$ element of f_i is set to 1, i.e., a "one-hot" encoding. For example, with 12 possible simultaneous satellite observations and a reported observation of 5 satellites, the feature vector would be given by:

$$f_i^T = [\ 0,\ 0,\ 0,\ 0,\ 0,\ 1,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0,\ 0\ ]$$

This representation allows a different covariance model to be fit to each number of satellites, but requires more training data through the whole range of possible $n_{sat,i}$. We expect to learn large weights for lower indices and small weights at the larger indices as the number of satellites should decrease the error.

3) Dilution of Precision Noise Model: A few standard outputs from GPS sensors are intended to represent the positional uncertainty in terms of geometric dilution of precision. Three such values are typically reported: horizontal dilution of precision (hdop), positional dilution of precision (pdop), and time dilution of precision (tdop). Each represents a multiplicative scaling of the uncertainty as a function of the geometric configuration of satellites, relative to the sensor, and should be informative of the true uncertainty. We can incorporate these values into our feature vector by letting (for example) $f_i^T = [\ hdop_i\ ]$. We expect a positive correlation between these values and true uncertainty.

4) Vendor-Provided Noise Model: Many GPS units provide an uncertainty estimate computed by the sensor. To evaluate the quality of this estimate, we set our feature vector to contain just this estimate:

$$f_i^T = [\ \sigma_{vendor,i}\ ]$$

If the vendor supplied estimates are correct, we would expect to learn a weight of 1 for this feature.

5) Combination Noise Model: The uncertainty estimate σ_i is a linear combination of features and, thus, the aforementioned features can be combined, with the individual feature and weight components retaining their original meaning. For example, combining the constant model, the simplified number of satellites model, and the vendor-provided estimate would produce the feature vector:

$$f_i^T = [\ 1,\ n_{sat,i},\ \sigma_{vendor,i}\ ]$$

We expect this feature to outperform the individual features, which are subsumed by the combination model, so long as over-fitting is avoided.

B. Additional Consideration for Max-Mixture Models

The aforementioned models can be used directly as uni-modal models of GPS uncertainty. However, simple Gaussian error models for GPS tend to perform poorly, as both earlier work and our experiments show. Yet, a $k = 2$ max mixture model provides significant improvements. With $k = 2$, characterization of GPS sensors using max-mixtures requires maximizing L over a single α value and two w vectors.

Let w_j be the dataset-wide weight vector associated with the jth feature. For observation i, let $f_{i,j}$ be feature j and standard deviation now given by $\sigma_{i,j} = f_{i,j}^T w_j$. For example, the features of a 2 component mixture of the constant model and the combination model (from previous section) are given by:

$$f_i = \begin{bmatrix} f_{i,1}^T \\ f_{i,2}^T \end{bmatrix} = \begin{bmatrix} [\ 1\ ] \\ [\ 1,\ n_{sat,i},\ \sigma_{vendor,i}\ ] \end{bmatrix} \quad (7)$$

With w now defining the k dataset-wide weight vectors w_j:

$$w = \begin{bmatrix} w_1^T \\ w_2^T \end{bmatrix} = \begin{bmatrix} [\ w_{1,0}\ ] \\ [\ w_{2,0},\ w_{2,1},\ w_{2,2}\ ] \end{bmatrix} \quad (8)$$

The feature vectors and weights combine, as expected to produce k standard deviations:

$$\begin{bmatrix} \sigma_{i,1} \\ \sigma_{i,2} \end{bmatrix} = \begin{bmatrix} f_{i,1}^T w_1 \\ f_{i,2}^T w_2 \end{bmatrix} \quad (9)$$

Combining with (5), this leads to a value for σ_i defined as:

$$\sigma_i = \underset{\sigma_{i,j}}{\arg\max} \left(\log(\alpha_j) - 2\log(\sigma_{i,j}) - \frac{e_i^2}{2\sigma_{i,j}^2} \right) \quad (10)$$

To illustrate the max-mixture approach, suppose that some GPS measurements are nominal (with errors of a few meters), while other measurements are "outliers" (with errors of tens of meters). With a uni-modal approach using $f = [\ 1\ ]$, we might learn $w = 9$. However, by setting $f = [\ [\ 1\ ],\ [\ 1\ ]\ ]$ (a mixture of two constant variances) we might expect to learn $w = [\ [\ 2\ ],\ [\ 30\ ]\ ]$ and a value of α in relation to the frequency of those outliers. While simplistic we will show that this method works well.

VI. Evaluation

In this section we evaluate our proposed GPS noise models on a 14-robot dataset collected within a 220 x 160 m indoor/outdoor region of the Adelaide Showgrounds in South Australia using Garmin GPS18x-5Hz sensors during the MAGIC competition [22].

A. Performance Metrics

To analyze the GPS sensor characterization we analyze three primary metrics for each model: 1) likelihood of observed data, 2) robustness to high-error observations, and, more generally, 3) the distribution of error relative to model-predicted error.

We use the normalized log-likelihood of the data, given the model and parameters learned off-line, to measure the overall fit of the model. This metric represents the expected log-likelihood of each observation and we desire models that maximize this metric. Note that the normalization constant η is not computed, since it does not affect the maximization of this metric.

In the SLAM context, or any similar non-linear optimization problem, a single erroneous measurement can wreak havoc if not properly modeled. For uni-modal models, outliers have low probability and $\min \log P(e_i)$ represents the worst-case likelihood error.

Since SLAM computes an ML solution via non-linear optimization and a 1st order Taylor expansion, the gradient magnitude, $||\nabla_i||$, is another outlier metric. During the optimization process $||\nabla_i||$ represents the "pull" of the constraint, relative to other constraints' gradients. For a given uni-modal Gaussian, the least likely measurement also has the maximum gradient, i.e., $\arg\min_i L_i = \arg\max_j ||\nabla_j||$. However, with max-mixtures an observation may be expected with near zero probability, but have virtually no gradient and thus no 'pull' within the optimization process. We desire models that minimize the "pull" associated with outlier measurements, i.e., we desire small $\max ||\nabla_i|| = \max ||\Sigma_i \mathbf{e}_i||$.

For each observation, e_i/σ_i represents the number of standard deviations predicted by the model. If the noise was truly distributed according to a uni-modal Gaussian distribution, the e_i/σ_i errors would be distributed as a χ distribution. Although this metric does not exactly relate to max-mixtures, we still desire e_i/σ_i distributions that are quantitatively similar to a χ distribution. For example, we hope to find models where all observations fall within the statistically significant region, $\max(e_i/\sigma_i) < 6$.

B. Analysis

Because our evaluation datasets were dominated by inliers, the normalized log likelihood of all of our models, including both simple Gaussian models and max-mixture models, falls within a relatively small range (from -7.3 to -6.3). These

TABLE I: Training and Testing Error.

Model	Uni-Modal						Max-Mixture					
	Train			Test			Train			Test		
	$\frac{1}{n}\sum L_i$	$\max \frac{e_i}{\sigma_i}$	$\max \lVert\nabla_i\rVert$	$\frac{1}{n}\sum L_i$	$\max \frac{e_i}{\sigma_i}$	$\max \lVert\nabla_i\rVert$	$\frac{1}{n}\sum L_i$	$\max \frac{e_i}{\sigma_i}$	$\max \lVert\nabla_i\rVert$	$\frac{1}{n}\sum L_i$	$\max \frac{e_i}{\sigma_i}$	$\max \lVert\nabla_i\rVert$
constant-hdop	-6.855	14.503	2.920	-6.852	17.895	2.750	-6.587	8.810	1.363	-6.623	10.946	1.300
constant-hdop-nsat-vendor	-6.299	6.866	2.260	-6.318	11.451	2.127	-6.289	5.142	1.840	-6.306	7.261	1.836
constant-hdop-vendor	-6.312	6.342	1.957	-6.325	11.548	1.890	-6.303	4.318	1.631	-6.325	7.932	1.664
constant-nsat-vendor	-6.302	6.785	2.208	-6.319	11.595	2.096	-6.271	6.736	2.431	-6.286	10.770	2.275
constant-vendor	-6.312	6.331	1.948	-6.326	11.516	1.878	-6.305	4.286	1.737	-6.321	7.292	1.791
constant	-7.210	21.106	3.263	-7.321	21.618	3.251	-6.733	12.353	1.118	-6.820	12.653	1.114
vendor	-6.329	6.177	2.514	-6.336	9.643	2.331	-6.320	4.204	2.048	-6.330	6.563	2.024

small differences arise from dramatic differences in the log likelihoods of the relatively-infrequent outlier measurements.

As seen in Table I, for any given feature vector, the performance of a mixture of two components invariably outperformed the best single component (simple Gaussian) error model, according to both training and test error. As measured by log likelihood, the magnitude of these differences is small, but again, this is due to the fact that the vast majority of the measurements were inliers that every model handled well. As expected, we find that test error is generally somewhat higher than training error, but the magnitude of the increase is similar between both the simple Gaussian model and the more complex max-mixture models.

Unlike the normalized log likelihood, in which the performance of the models on outliers is masked by the large number of inliers, the *worst case* standard deviation metric clearly shows the advantages of the mixture models. The worst-case standard deviation, $\max e_i/\sigma_i$, dropped for every model, often dramatically, e.g., from 21.618 to 12.653 in the constant-covariance model case. These improvements highlight a significant improvement in modeling the sensor's noise. To be clear, measurements with 12 standard deviations of error may still have too much influence during optimization, but the constant model is a very naive baseline and ignores any uncertainty indications/features the sensor.

Histograms of the empirical χ error compared to the model's predicted density are also revealing (see Fig. 2). In the case of simple Gaussian models (left column), we see that the model is consistently conservative with respect to the inlier data (on the left side of the plots). The outliers have caused an increase in the covariance estimate, with the consequence that inliers are given too little weight. Despite the inflated covariance estimate, outliers still have a very high gradient (see Table I) due to their distance from the Gaussian distribution's mean and will strongly influence the optimization.

Conversely, the right column of Fig. 2, which plots histograms of the max-mixture models, shows a model error that more tightly tracks the ideal distribution. Simultaneously, outliers are shifted closer to the left, indicating that higher probabilities have been predicted for them. These ouliers still influence the graph, but have correspondingly smaller "pull", which would make a SLAM system more resilient to them.
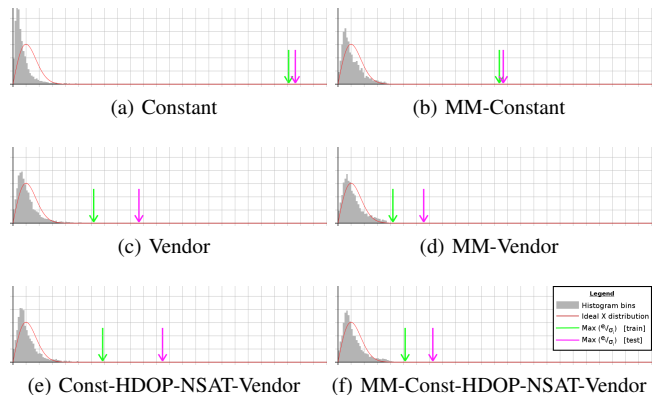

(a) Constant (b) MM-Constant

(c) Vendor (d) MM-Vendor

(e) Const-HDOP-NSAT-Vendor (f) MM-Const-HDOP-NSAT-Vendor

Fig. 2: Histogram of Empirical vs. Ideal χ-error (for select models). Ideally, if the underlying Gaussian assumptions hold, the normalized histogram of e_i/σ_i would fit a 2 DOF χ distribution (with horizontal units representing standard deviations, σ_i). The relative movement of the *worst-case* arrows to lower standard deviations, uni-modal models (left) versus max-mixture models (right), highlights the improvements in modeling capabilities, specifically robustness to outliers.

C. Learned Weights

We next present a few learned parameter settings for discussion purposes correspond to the respective models shown in Table I.

The constant models learned weights of $[9.29]$ for uni-modal and $[[5.2], [16.2]]$ with $\alpha = [0.98, 0.02]$ for the max-mixture model. This reflects the fact that the sensor performs well most of the time, but a uni-modal model must compensate for the high-error observations with an overestimate of σ to fit the Gaussian assumption.

We were pleasantly surprised by the quality of the vendor's uncertainties both in terms of likelihood and in terms of $\max e_i/\sigma_i$. The vendor model learned weights of $[0.8]$ and $[[0.73], [1.16]]$ with $\alpha = [0.97, 0.03]$, reflecting only modest adjustment of their estimates. However, we were still able to improve upon them with our method.

VII. Conclusion

We have described a general approach for computing sensor uncertainty estimates using a machine learning approach. Feature vectors are constructed from observations and a

weight vector is learned from a ground-truthed data set, via maximizing the log-likelihood of the training data set.

We showed how this approach can be extended to more expressive error models using max-mixtures. We take the view in this paper that "outliers" arise from mismatches between the empirical performance of a system and its error model: better models assign higher probabilities to outliers and thus mitigate their impact. Versus an explicit outlier rejection phase, the max-mixture approach both provides an integrated Bayesian mechanism for robust estimation and removes the need for a near-perfect outlier detector.

In evaluating the performance of these models we use standard log-likelihood metrics and introduce a metric that reflects the impact of an outlier on a SLAM system. Our work was evaluated on a large multi-robot dataset and we demonstrated significant performance improvements using our methods on the task of characterizing error-prone customer-grade GPS sensors. Related methods attempting to add robustness to the back-end of SLAM would also benefit from improved robustness on the front-end provided by the approach presented here.

REFERENCES

[1] E. Olson, "Robust and efficient robotic mapping," Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, MA, USA, June 2008.

[2] N. Sunderhauf and P. Protzel, "Switchable constraints for robust pose graph SLAM," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 1879–1884.

[3] Y. Latif, C. Cadena, and J. Neira, "Realizing, reversing, recovering: Incremental robust loop closing over time using the iRRR algorithm," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 4211–4217.

[4] E. Olson and P. Agarwal, "Inference on networks of mixtures for robust robot mapping," in *Proceedings of Robotics: Science and Systems (RSS)*, July 2012.

[5] P. Agarwal, G. D. Tipaldi, L. Spinello, C. Stachniss, and W. Burgard, "Robust map optimization using dynamic covariance scaling," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2013.

[6] Y. Latif, C. Cadena, and J. Neira, "Robust loop closing over time," in *Proceedings of Robotics: Science and Systems (RSS)*, 2012.

[7] P. J. Huber, "Robust estimation of a location parameter," *The Annals of Mathematical Statistics*, vol. 35, no. 1, pp. 73–101, 1964.

[8] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2004.

[9] S. Thrun and M. Montemerlo, "The GraphSLAM algorithm with applications to large-scale mapping of urban structures," *International Journal of Robotics Research*, vol. 25, no. 5-6, pp. 403–430, May-June 2006.

[10] A. Stentz, C. Dima, C. Wellington, H. Herman, and D. Stager, "A system for semi-autonomous tractor operations," *Autonomous Robots*, vol. 13, no. 1, pp. 87–104, 2002.

[11] J. Kim and S. Sukkarieh, "SLAM aided GPS/INS navigation in GPS denied and unknown environments," in *Proceedings of the 2004 International Symposium on GNSS/GPS*, 2004, pp. 1–5.

[12] M. Agrawal and K. Konolige, "Real-time localization in outdoor environments using stereo vision and inexpensive GPS," in *Proceedings of the International Conference on Pattern Recognition*. IEEE, 2006.

[13] W. Ding, J. Wang, S. Han, A. Almagbile, M. Garratt, A. Lambert, and J. Wang, "Adding optical flow into the GPS/INS integration for uav navigation," in *Proc. of International Global Navigation Satellite Systems Society Symposium*. Citeseer, 2009, pp. 1–13.

[14] K. Ohno, T. Tsubouchi, B. Shigematsu, and S. Yuta, "Differential GPS and odometry-based outdoor navigation of a mobile robot," *Advanced Robotics*, vol. 18, no. 6, pp. 611–635, 2004.

[15] D. C. Moore, A. S. Huang, M. Walter, E. Olson, L. Fletcher, J. Leonard, and S. Teller, "Simultaneous local and global state estimation for robotic navigation," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*. IEEE, 2009, pp. 3794–3799.

[16] L. Cremean, T. Foote, J. Gillula, G. Hines, D. Kogan, K. Kriechbaum, J. Lamb, J. Leibs, L. Lindzey, C. Rasmussen, *et al.*, "Alice: An information-rich autonomous vehicle for high-speed desert navigation," *Journal of Field Robotics*, vol. 23, no. 9, pp. 777–810, 2006.

[17] J. Wang, M. Garratt, A. Lambert, J. Wang, S. Han, and D. Sinclair, "Integration of GPS/INS/vision sensors to navigate unmanned aerial vehicles," in *XXI Congress of the Int. Society of Photogrammetry & Remote Sensing, Beijing, PR China*, 2008, pp. 963–970.

[18] B. Ben-Moshe, E. Elkin, H. Levi, and A. Weissman, "Improving accuracy of GNSS devices in urban canyons." in *CCCG*, 2011.

[19] D. Maier and A. Kleiner, "Improved GPS sensor model for mobile robots in urban terrain," *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4385–4390, 2010.

[20] E. Takeuchi, M. Yamazaki, K. Ohno, and S. Tadokoro, "GPS measurement model with satellite visibility using 3d map for particle filter," in *Robotics and Biomimetics (ROBIO), 2011 IEEE International Conference on*. IEEE, 2011, pp. 590–595.

[21] N. Sunderhauf, M. Obst, G. Wanielik, and P. Protzel, "Multipath mitigation in GNSS-based localization using robust optimization," in *Intelligent Vehicles Symposium (IV), 2012 IEEE*. IEEE, 2012, pp. 784–789.

[22] E. Olson, J. Strom, R. Morton, A. Richardson, P. Ranganathan, R. Goeddel, and M. Bulic, "Progress towards multi-robot reconnaissance and the MAGIC 2010 Competition," *Journal of Field Robotics*, vol. 29, September 2012.