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

CELLO-EM: Adaptive Sensor Models without Ground Truth

William Vega-Brown and Nicholas Roy

Abstract— **We present an algorithm for providing a dynamic model of sensor measurements. Rather than depending on a model of the vehicle state and environment to capture the distribution of possible sensor measurements, we provide an approximation that allows the sensor model to depend on the measurement itself. Building on previous work, we show how the sensor model predictor can be learned from data without access to ground truth labels of the vehicle state or true underlying distribution, and we show our approach to be a generalization of non-parametric kernel regressors. Our algorithm is demonstrated in simulation and on real world data for both laser-based scan matching odometry and RGB-D camera odometry in an unknown map. The performance of our algorithm is shown to quantitatively improve estimation, both in terms of consistency and absolute accuracy, relative to other algorithms and to fixed covariance models.**

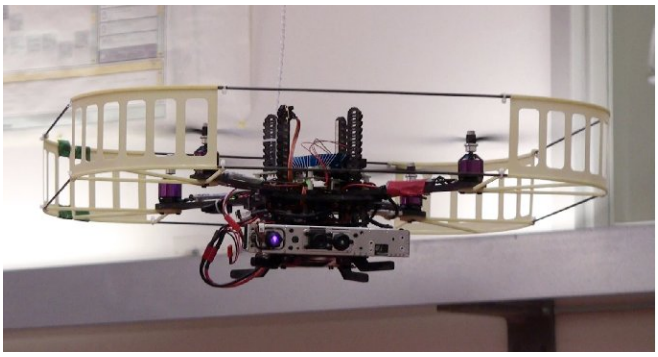


Fig. 1: *Micro air vehicles like this one require accurate state estimates to fly safely, but are limited in the sensing and computational resources they can carry.*

I. Introduction

Probabilistic methods have had substantial impact on robotics in recent years; by explicitly reasoning about uncertainty, robots are better equipped to make intelligent planning and control decisions in the face of limited information. These methods rely on accurate models of the system dynamics and sensors, which are usually parameterised in a way that describes the uncertainty of the vehicle motion or sensor model. For example, the commonly used linear Gaussian sensor model describes sensor noise with a covariance matrix. Identifying the parameters of the model is important to ensuring accuracy, consistency, and robustness.

One common approach is to assume that the model parameters are constant throughout the operation of the vehicle, and to use system identification techniques to learn the model parameters from a collected dataset. This method may fail to perform well in cases where the parameters depend on changing external factors that are unavailable to the robot. For example, as we will show in section III, laser scan matching has a dramatically higher uncertainty in a corridor environment than in a well-structured environment. We will see that a Kalman filter based estimator that incorporates motion estimates generated by a scan matcher suffers if it assumes a constant model of uncertainty of these motion estimates. Using a sensor model whose parameters vary with the sensor data leads to dramatically improved state estimation for navigating robots, such as the UAV shown in fig. 1.

Rather than resorting to a sensor model that depends on the full state of the vehicle and the environment, the sensor measurement itself can provide accurate predictions of sensor model parameters such as the covariance of the measurement distribution. Figure 2a shows such an example, where a laser range scan in a tightly constrained environment may provide very accurate position information for a navigating robot. On the other hand, fig. 2b shows an example of a laser scan in a corridor, where the measurement strongly constrains the robot's cross-range position in the corridor, but provides little information about where along the corridor the robot might be. Modelling the uncertainty in the position estimates of these two scans using the same covariance will cause the robot either to have insufficient confidence in its position in fig. 2a, or overly confident about its position along the corridor in fig. 2b.

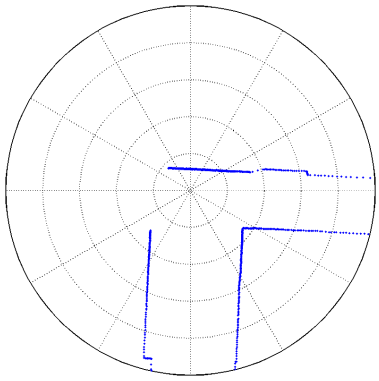
In this paper, we show a way to estimate the parameters of a sensor model directly from the measurements themselves. We learn a predictive model of uncertainty from training data. Because we predict the sensor model parameters from the received measurements themselves, we can incorporate sensor measurements directly into a state estimator without access to an environmental model such as a map. This allows our sensor model both to generalize across environments and to be used inside an online mapping process.

In previous work [1] we presented a related method for estimating these parameters when given access to the true measurement errors at training time—a restriction requiring access to the true state of the vehicle, which is difficult to achieve in domains like mobile robotics. We now develop a general graphical model incorporating varying model parameters, and show how we can efficiently perform approximate inference over this model. We then show the connection between CELLO and kernel regression, and note the method presented can be extended to a much wider class of models. Finally, we demonstrate improvements to estimator accuracy and consistency when using covariances predicted by CELLO in an extended Kalman filter.

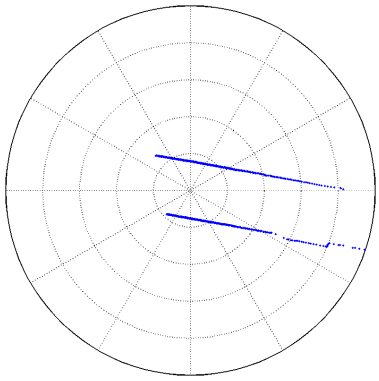
II. Generalized Predictive Estimation

Suppose we have a robot with state $\mathbf{x}_t \in \mathbb{R}^{N_x}$ at time t, with N_x the dimensionality of the state vector. It is equipped with a sensor providing noisy raw measurements $\boldsymbol{\zeta}_t \in \mathbb{R}^{N_\zeta}$ at each time step; these measurements could be the pixel values of a camera, for instance, or the ranges returned by

978-1-4673-6358-7/13/$31.00 ©2013 IEEE



(a) Corner



(b) Hallway

Fig. 2: *Sequential laser scans can be matched to estimate the velocity of a robot, but environmental degeneracy can lead to less information in some directions. Near a corner, the robot can accurately estimate changes in both heading and each position axis; in a hallway, the robot can only estimate motion in the direction of the walls.*

a planar lidar unit, and as such the measurement dimensionality N_ζ may be quite large. The raw sensor measurements are dependent on both the (unobserved) robot state $\mathbf{x}_t$ and on the environment, described by Ω. The structure of this probabilistic model is shown in fig. 3a. Given a sensor model with this structure, as well as a model of the state dynamics, the history of measurements can be used to infer the state of the robot as it moves.[1]

We can often simplify the process of inference by not directly incorporating the raw sensor data $\boldsymbol{\zeta}$. Instead, we first compute some low-dimensional representation $\mathbf{z}_t = \mathbf{z}(\boldsymbol{\zeta}_t)$ of the information contained in the high-dimensional signal available from sensors, such as a velocity vector computed from optical flow on an image, or the relative displacement of the vehicle computed from laser range scans. Consider again the laser scans in fig. 2; the relative displacement of the vehicle computed by a scan matching process can be treated as a low-dimensional measurement of velocity with some covariance, and then incorporated into a state estimator such as a Kalman filter. This would provide the efficiency of Kalman filtering while at the same time allowing the laser data to be fused with other sensors such as an IMU, camera or GPS. The low-dimensional representation is often chosen to facilitate inference of the latent states $\mathbf{x}$; commonly, we choose a representation that makes the state $\mathbf{x}$ conditionally independent of the raw measurement $\boldsymbol{\zeta}$ given the observation $\mathbf{z}$, with the distribution of $\mathbf{z}$ given $\mathbf{x}$ well-approximated by a simple parametric form, such as a multivariate Gaussian.

$$p\left(\mathbf{z}_t|\mathbf{x}_t, \boldsymbol{\theta}_t\right) = \mathcal{N}\left(h(\mathbf{x}_t), \mathbf{R}_t(\boldsymbol{\theta}_t)\right). \tag{1}$$

Here, $h(\mathbf{x})$ is a known deterministic *measurement function*, and the parameters $\boldsymbol{\theta}_t$ are some parameterization of the $\frac{1}{2}N_z(N_z+1)$ free parameters of the measurement covariance matrix $\mathbf{R}_t$. The problems of filtering, prediction, and smoothing on such models are well-studied, and solved by suitable variants on the Kalman filter if the measurement covariance $\mathbf{R}_t$ is known.

A. Approximate Inference and Graphical Model Structure

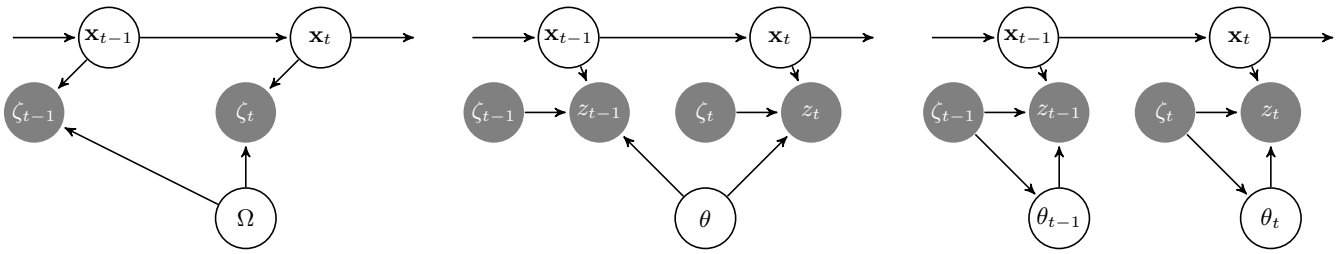
In general, the parameters $\boldsymbol{\theta}$ are not known *a priori*, and must be estimated from data in the problem domain. This can be done using the Baum-Welch algorithm [2], which finds the maximum likelihood parameters $\boldsymbol{\theta}$ by alternately evaluating the expected value of the latent states $\mathbf{x}$ and choosing the maximum likelihood parameters $\boldsymbol{\theta}$ conditioned on those expected states. This method relies only on the low-dimensional measurement $\mathbf{z}$ and not on the high-dimensional observation $\boldsymbol{\zeta}$.

However, due to features of the environment or stochasticity in the sensor, the available information in the raw sensor measurement $\boldsymbol{\zeta}$ may vary, leading the parameters of the measurement distribution $\boldsymbol{\theta}$ to vary as well. Instead of a single, common node for the distribution parameters $\boldsymbol{\theta}$, we now have a distinct node $\boldsymbol{\theta}_t$ for each measurement $\mathbf{z}_t$, as shown in fig. 3c. In order to retain the efficiency gains made by using the low-dimensional measurement $\mathbf{z}$ instead of the observation $\boldsymbol{\zeta}$, we must develop a predictive model for these measurement parameters $\boldsymbol{\theta}_t$.

Given perfect knowledge of the state and the environment, the distribution over measurements $\boldsymbol{\zeta}$ is fixed and thus the distribution over observations $\mathbf{z}$ is likewise fixed. We could then infer the maximum likelihood parameters of the measurement distribution, for a particular state vector $\mathbf{x}_t$ and environment Ω. However, we cannot have perfect information about the state and environment; even if we could, the inferred parameter estimates could not be generalized to new environments.

If we make the approximation that the parameters $\boldsymbol{\theta}$ are well-modelled as a function of the observations $\boldsymbol{\zeta}$, we can frame the choice of parameters as a learning problem. This approximation allows us to generalize to new environments; we capture the dependence of the observation $\mathbf{z}$ on the environment Ω in the dependence of the parameters θ on the raw measurement $\boldsymbol{\zeta}$. We may reduce the complexity of this learning problem by introducing a low-dimensional vector of *predictor features* $\boldsymbol{\phi}$ derived from the raw sensor data $\boldsymbol{\zeta}$. A

[1] Without loss of generality, we omit further discussion of the dynamics model for clarity. We refer the reader to any standard text on Kalman filtering for a reference on how to incorporate dynamics and process noise into the estimate.



(a) True Model (b) Standard Approximation (c) CELLO Approximation

Fig. 3: *The fixed-parameter model provides for efficient inference, but neglects the fact that sensor measurements are both a function of the state and the environment Ω. The approximate model employed in CELLO captures that dependence without sacrificing the sparsity of the hidden Markov model.*

careful choice of predictors allows us to dramatically reduce the dimensionality of the function we are trying to learn without impeding the quality of the resulting predictions; as we showed previously, an image of millions of pixels can be reduced to a few statistics that are sufficient to estimate the covariance of the output of a vision algorithm.

Our goal, then, is to learn a map $\mathbb{R}^{N_\phi} \rightarrow \mathbb{R}^{N_\theta}$ from a vector of predictors $\boldsymbol{\phi}$ to the parameters $\boldsymbol{\theta}$ of low dimensional measurements $\mathbf{z}$. We have at our disposal a set of K raw sensor observations $\{\boldsymbol{\zeta}_t \,\forall\, t \in [1, K]\}$, from which we derive pairs of vectors $(\mathbf{z}_t, \boldsymbol{\phi}_t)$. The latent state variables $\{\mathbf{x}_t \,\forall\, t \in [1, K]\}$ are not observable, but we assume they have known dependence structure as in the models in fig. 3. The CELLO probabilistic model creates an hierarchical conditional independence structure on the distribution of the four random vectors $\{\mathbf{x}_t, \mathbf{z}_t, \boldsymbol{\phi}_t, \boldsymbol{\theta}_t\}$.

$$p\left(\mathbf{x}_t, \mathbf{z}_t, \boldsymbol{\theta}_t, \boldsymbol{\phi}_t\right) = p\left(\mathbf{z}_t | \mathbf{x}_t, \boldsymbol{\theta}_t\right) p\left(\boldsymbol{\theta}_t | \boldsymbol{\phi}_t\right) p\left(\boldsymbol{\phi}_t\right) p\left(\mathbf{x}_t\right) \quad (2)$$

Given that distribution and the available data, we aim to predict the expected values of the unobservable parameters $\boldsymbol{\theta}_t$. That is, we aim to evaluate

$$\hat{\boldsymbol{\theta}}(\boldsymbol{\phi}) = \int \boldsymbol{\theta}_t p\left(\mathbf{x}_t, \mathbf{z}_t, \boldsymbol{\theta}_t, | \boldsymbol{\phi}_t\right) \, \mathrm{d}\boldsymbol{\theta} \, \mathrm{d}\mathbf{z} \, \mathrm{d}\mathbf{x} \quad (3)$$

B. Covariance Estimation

We initially restrict ourselves to learning the covariances of zero-mean fixed-parameter multivariate Gaussian measurement models, and later generalize, first to the case of varying-parameters and later to the case of arbitrary measurement models. A standard result in statistics is that given a set of K independent samples from a multivariate Gaussian distribution, the minimum variance unbiased estimate of the covariance is the empirical covariance. That is, if $\mathbf{v}_i \sim \mathcal{N}\left(\mathbf{0}, \mathbf{R}\right)$,

$$\mathbf{E}\left[\frac{1}{K}\sum_i^K \mathbf{v}_i \mathbf{v}_i^\top\right] = \mathbf{E}\left[\mathbf{v}\mathbf{v}^\top\right] = \mathbf{E}\left[\mathbf{T}_{\mathbf{R}}(\mathbf{v})\right] = \mathbf{R} \quad (4)$$

where $\mathbf{T}_{\mathbf{R}}(\mathbf{v}) = \mathbf{v}\mathbf{v}^\top$ is an estimator for $\mathbf{R}$ given the set of samples $\{\mathbf{v}_t \,\forall\, t \in [1, K]\}$. For a linear Gaussian measurement model, $h(\mathbf{x}_t) = H\mathbf{x}_t + \mathbf{v}_t$ with $\mathbf{v}_t \sim \mathcal{N}\left(\mathbf{0}, \mathbf{R}_t\right)$, we may generate samples $\mathbf{v}_t = \mathbf{z}_t - H\mathbf{x}_t$ if we have access to both the measurements $\mathbf{z}$ and the states $\mathbf{x}$. The minimum variance estimator for the measurement covariance is then the outer product.

$$\mathbf{E}\left[\mathbf{T}_{\mathbf{R}}(\mathbf{x}, \mathbf{z})\right] = \mathbf{E}\left[(\mathbf{z}_t - H\mathbf{x}_t)(\mathbf{z}_t - H\mathbf{x}_t)^\top\right] = \mathbf{R}_t \quad (5)$$

Given a multivariate Gaussian distribution over the state instead of a direct observation, $\mathbf{x}_t \sim \mathcal{N}\left(\hat{\mathbf{x}}_t, \hat{\boldsymbol{\Sigma}}\right)$, we find a similar relationship.

$$\begin{aligned}
&\mathbf{E}\left[(\mathbf{z}_t - H\mathbf{x}_t)(\mathbf{z}_t - H\mathbf{x}_t)^\top\right] \\
&\quad = \int \mathrm{d}\mathbf{x} \, (\mathbf{z}_t - H\mathbf{x}_t)(\mathbf{z}_t - H\mathbf{x}_t)^\top \mathcal{N}\left(\hat{\mathbf{x}}_t, \hat{\boldsymbol{\Sigma}}_t\right) \\
&\quad = \int \mathrm{d}\mathbf{x} \, (\mathbf{z}_t - H\mathbf{x}_t)(\mathbf{z}_t - H\mathbf{x}_t)^\top \mathcal{N}\left(\hat{\mathbf{x}}_t, \hat{\boldsymbol{\Sigma}}_t\right) \\
&\quad = \mathbf{R}_t + H\hat{\boldsymbol{\Sigma}}_t H^\top
\end{aligned} \quad (6)$$

Consequentially, we may form an unbiased estimator for $\mathbf{R}$ given measurements $\mathbf{z}$ and distributions over the state $\mathbf{x}$.

$$\mathbf{E}\left[(\mathbf{z}_t - H\hat{\mathbf{x}}_t)(\mathbf{z}_t - H\hat{\mathbf{x}}_t)^\top - H\hat{\boldsymbol{\Sigma}}_t H^\top\right] = \mathbf{R}_t \quad (7)$$

C. Kernel Estimation

We now permit the covariance $\mathbf{R}$ to vary with the predictors $\boldsymbol{\phi}$. Let the vector $\boldsymbol{\theta}$ represent the independent elements of the matrix $\mathbf{R}$, and the elements of the vector estimator $\mathbf{T}_{\boldsymbol{\theta}}\left(\mathbf{x}, \mathbf{z}\right)$ represent the corresponding elements of the matrix estimator $\mathbf{T}_{\mathbf{R}}$. Equation (7) remains true when conditioned on the predictor vector $\boldsymbol{\phi}$.

$$\mathbf{E}\left[\boldsymbol{\theta}_t | \boldsymbol{\phi}_t\right] = \int \boldsymbol{\theta}_t p\left(\mathbf{x}_t, \mathbf{z}_t, \boldsymbol{\theta}_t, | \boldsymbol{\phi}_t\right) \, \mathrm{d}\boldsymbol{\theta} \, \mathrm{d}\mathbf{z} \, \mathrm{d}\mathbf{x} \quad (8)$$

$$= \frac{\int \boldsymbol{\theta}_t p\left(\mathbf{x}, \mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\phi}_t\right) \, \mathrm{d}\boldsymbol{\theta} \, \mathrm{d}\mathbf{z} \, \mathrm{d}\mathbf{x}}{\int p\left(\mathbf{x}, \mathbf{z}, \boldsymbol{\theta}, \boldsymbol{\phi}_t\right) \, \mathrm{d}\boldsymbol{\theta} \, \mathrm{d}\mathbf{z} \, \mathrm{d}\mathbf{x}} \quad (9)$$

$$= \frac{\int \mathbf{E}\left[\mathbf{T}_{\boldsymbol{\theta}}\left(\mathbf{x}, \mathbf{z}\right)\right] p\left(\boldsymbol{\theta}, \boldsymbol{\phi}_t\right) \, \mathrm{d}\boldsymbol{\theta}}{\int p\left(\boldsymbol{\theta}, \boldsymbol{\phi}_t\right) \, \mathrm{d}\boldsymbol{\theta}} \quad (10)$$

The expectation in eq. (10) is taken over the latent states $\mathbf{x}$; the simplification exploits the conditional independence relation of eq. (2). Given a set of observations

$$\mathcal{D} = \{\mathbf{z}_i, \boldsymbol{\phi}_i \quad \forall i \in [1, N]\} \quad (11)$$

we may approximate that joint distribution using kernel regression techniques.

$$\hat{p}\left(\boldsymbol{\theta}, \boldsymbol{\phi}\right) = \frac{1}{N}\sum_{i=1}^N k_{\rho_{\boldsymbol{\phi}}}\left(\|\boldsymbol{\phi} - \boldsymbol{\phi}_i\|\right) k_{\rho_{\boldsymbol{\theta}}}\left(\|\boldsymbol{\theta} - \boldsymbol{\theta}_i\|\right) \quad (12)$$

The *kernel functions* $k_\rho(\|\mathbf{x}\|)$ are required only to be positive, normalized, and decreasing in $\|\mathbf{x}\|$, although typically they will also be symmetric. The kernel scale ρ is a scalar defining the size of the kernel, defined such that $k_\rho(\|\mathbf{x}\|) = \rho k_1(\rho\|\mathbf{x}\|)$.

This approximation gives rise to an estimator for the expectation in eq. (10).

$$\hat{\boldsymbol{\theta}}(\boldsymbol{\phi}_t) = \frac{\sum_i \mathbf{E}\left[\mathbf{T}_{\boldsymbol{\theta}}(\mathbf{x}_i, \mathbf{z}_i)\right] k_{\boldsymbol{\phi}}(\|\boldsymbol{\phi}_t - \boldsymbol{\phi}_i\|)}{\sum_i k_{\boldsymbol{\phi}}(\|\boldsymbol{\phi}_t - \boldsymbol{\phi}_i\|)} \tag{13}$$

This is a Nadaraya-Watson estimator [3, 4], extended to estimate an unobservable quantity in terms of the observable statistic $\mathbf{E}\left[\mathbf{T}_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{z})\right]$. The kernel function allows us to compute an expected covariance by averaging over a set of nearby, but not identical, measurements in the data set. However, the kernel functions themselves have parameters defining the notion of 'nearness'; we must choose these parameters well to ensure good performance.

It is helpful to observe the asymptotic properties of the kernel estimator, to demonstrate that there exist parameters for which performance is guaranteed. We first define the function $\boldsymbol{\theta}(\boldsymbol{\phi}) = \mathbf{E}\left[\boldsymbol{\theta}|\boldsymbol{\phi}\right]$ for convenience. We denote the kernel scale as ρ and assume the kernel metric to be of *generalized Euclidean form*, with a metric tensor $\mathbf{M}$:

$$\|\boldsymbol{\phi} - \boldsymbol{\phi}_i\| = \sqrt{(\boldsymbol{\phi} - \boldsymbol{\phi}_i)^\top \mathbf{M} (\boldsymbol{\phi} - \boldsymbol{\phi}_i)} \tag{14}$$

This permits the definition of a local coordinate system $\boldsymbol{\varphi} = \mathbf{L}(\boldsymbol{\phi} - \boldsymbol{\phi}_i)$, where $\mathbf{M} = \rho \mathbf{L}^\top \mathbf{L}$, such that

$$\int \mathrm{d}\boldsymbol{\varphi}\, \rho k(\boldsymbol{\varphi}) = \int \mathrm{d}\boldsymbol{\phi}\, k_\rho(\boldsymbol{\phi} - \boldsymbol{\phi}_i). \tag{15}$$

With these assumptions, it can be shown that, in the limit of many samples and small scale, the estimator is unbiased to first order.

$$\lim_{\substack{\rho\to 0 \\ N\rho\to\infty}} \mathbf{E}\left[\hat{\boldsymbol{\theta}}(\boldsymbol{\phi}) - \boldsymbol{\theta}(\boldsymbol{\phi})\right] = \left(\frac{1}{2}\nabla^2_{\mathbf{M}}\theta_i(\boldsymbol{\phi}) + \nabla \log p(\boldsymbol{\phi})^\top \mathbf{M}^{-1} \nabla\boldsymbol{\theta}(\boldsymbol{\phi})\right)\rho^2 c_K \tag{16}$$

Here, $\nabla^2_{\mathbf{M}}\theta_i(\boldsymbol{\phi}) = \mathrm{tr}\left[\nabla\nabla^\top \theta_i(\boldsymbol{\phi})\mathbf{M}^{-1}\right]$ is the Laplacian under the metric $\mathbf{M}$, and $c_k = \int \varphi_i^2 k(\boldsymbol{\varphi})\mathrm{d}\boldsymbol{\varphi}$ is the second moment of any element under the kernel, since each element is treated identically. In the special case of the Euclidean metric, where $\mathbf{M} = \mathbf{1}$, and uniform sampling density, this can be reduced further.

$$\lim_{\substack{\rho\to 0 \\ N\rho\to\infty}} \mathbf{E}\left[\hat{\boldsymbol{\theta}}(\boldsymbol{\phi}) - \boldsymbol{\theta}(\boldsymbol{\phi})\right] = \frac{1}{2}\nabla^2\theta_i(\boldsymbol{\phi})\rho^2 c_K \tag{17}$$

The same assumptions give an asymptotic variance.

$$\lim_{\substack{\rho\to 0 \\ N\rho\to\infty}} \mathbf{V}\left[\hat{\boldsymbol{\theta}}(\boldsymbol{\phi})\right] = \frac{d_K}{p(\boldsymbol{\phi}) N\rho} \mathbf{V}\left[\theta_i|\boldsymbol{\phi}\right] \tag{18}$$

Here, $d_k = \int k(\boldsymbol{\varphi})^2 \mathrm{d}\boldsymbol{\varphi}$. As N increases, the variance of the estimate will asymptotically approach zero; provided the kernel scale ρ decreases in N the estimator bias likewise becomes zero. Thus, given a sufficiently large dataset $\mathcal{D}$, we have obtained a consistent estimator for $\mathbf{E}\left[\boldsymbol{\theta}|\boldsymbol{\phi}\right]$.

D. CELLO-EM

Given a set of data and an appropriate choice of kernel parameters, we can predict the covariance of any future measurement from the sum of outer products of the previous measurements, weighted by a similarity measure. Measurements that are close together in feature space will more strongly influence the covariance prediction than measurements far apart in feature space. To make predictions using any fixed data set, we must choose kernel parameters $\{\rho, \mathbf{M}\}$ appropriately. Choosing analytically optimal $\{\rho, \mathbf{M}\}$ is infeasible as we do not know the underlying parameter function $\boldsymbol{\theta}(\boldsymbol{\phi})$. Instead, we parameterize the metric and scale by a vector of hyperparameters $\boldsymbol{\alpha}$ and choose the hyperparameters which maximize the likelihood of the dataset.

$$\boldsymbol{\alpha}^* = \operatorname*{argmax}_{\boldsymbol{\alpha}} \prod_{i=1}^{N} p(\mathbf{z}_i, \hat{\boldsymbol{\theta}}(\boldsymbol{\phi}_i), \boldsymbol{\alpha}) \tag{19}$$

$$= \operatorname*{argmax}_{\boldsymbol{\alpha}} \prod_{i=1}^{N} \int p(\mathbf{x}_i, \mathbf{z}_i, \hat{\boldsymbol{\theta}}(\boldsymbol{\phi}_i), \boldsymbol{\alpha}) \mathrm{d}\mathbf{x}_1 \ldots \mathrm{d}\mathbf{x}_N \tag{20}$$

This optimization may be efficiently evaluated through the use of the Expectation-Maximization algorithm [5]. We iterate through two steps; first, we evaluate the expected latent state sequence.

$$\hat{\mathbf{x}}_1^{(n)} \ldots \hat{\mathbf{x}}_N^{(n)} = \mathbf{E}\left[\mathbf{x}_1 \ldots \mathbf{x}_N | \mathbf{z}_i, \hat{\boldsymbol{\theta}}(\boldsymbol{\phi}_i), \boldsymbol{\alpha}^*_{(n)}\right] \tag{21}$$

This expectation is well-studied and efficient solutions exist for many models; the general solution for a hidden Markov model is the forward-backward algorithm. We then evaluate the maximum likelihood hyperparameters.

$$\boldsymbol{\alpha}^*_{(n+1)} = \operatorname*{argmax}_{\boldsymbol{\alpha}} \prod_{i=1}^{N} \log p(\hat{\mathbf{x}}_i^{(n)}, \mathbf{z}_i, \hat{\boldsymbol{\theta}}(\boldsymbol{\phi}_i), \boldsymbol{\alpha}) \tag{22}$$

This optimization may be done using standard techniques, such as stochastic gradient ascent. Iterating between these steps will converge to a maximum of eq. (20).

E. General Parameter Estimation

The analysis of the previous section is not restricted to multivariate Gaussian measurement models; any parametric family may be used, provided there exists an unbiased estimator for its parameters. That is, we may generate a consistent estimator any measurement model $\mathbf{z} \sim p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta})$ so long as there exists a function $\mathbf{T}_{\boldsymbol{\theta}}(\mathbf{z})$ such that $\mathbf{E}\left[\mathbf{T}_{\boldsymbol{\theta}}(\mathbf{z})\right] = f(\boldsymbol{\theta})$, where $f(\cdot)$ is an invertible deterministic function. Every member of the exponential family has such an estimator; because we do not restrict the estimator to be a fixed-size statistic, there may be viable distributions outside the exponential family as well.

The procedure for generating a consistent estimator is given explicitly in algorithm 1. We cycle between evaluating the expected value of the estimator $\mathbf{T}_{\boldsymbol{\theta}}(\cdot)$ as in eq. (21), and determining the maximum likelihood hyperparameters as in eq. (22). Iterating between these steps will converge to an optimal set of hyperparameters and a distribution over the state sequence $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ of estimator values, which may be used at run time to predict the parameters of new

measurements. In practice, this convergence happens very quickly, generally after just a few iterations.

Algorithm 1 Covariance estimation through Expectation Maximization

Initialize $\hat{\mathbf{R}}$
repeat
  $\{\hat{\mathbf{x}}\} \leftarrow$EXPECTATION$(\mathcal{D}, \hat{\mathbf{R}})$
  $\boldsymbol{\alpha}^* \leftarrow$MAXIMIZE$(\{\hat{\mathbf{x}}\}, \{\mathbf{z}\})$
until convergence
function MAXIMIZE
  Randomly initialize parameter vector $\boldsymbol{\alpha}$ and set learning rate η
  repeat
    $\mathbf{I} \leftarrow$ SHUFFLE$([1, \ldots, N])$
    for $i \in \mathrm{I}$ **do**
      $\hat{\mathbf{R}}_i \leftarrow$PREDICTCOVARIANCE$(\boldsymbol{\phi}_i)$
      $\boldsymbol{\alpha} \leftarrow \boldsymbol{\alpha} - \eta \nabla \mathcal{L}(\boldsymbol{\alpha}, \mathcal{D})$
    end for
  until convergence
end function
function PREDICTCOVARIANCE$(\boldsymbol{\phi})$
  $\hat{\mathbf{R}} \leftarrow \mathbf{0}_{p \times p}$
  $n \leftarrow 0$
  $N_{\boldsymbol{\phi}} \leftarrow$NEARESTNEIGHBORS$(\boldsymbol{\phi})$
  for $i \in N_{\boldsymbol{\phi}}$ **do**
    $\hat{\mathbf{R}} \leftarrow \hat{\mathbf{R}} + k(\boldsymbol{\phi}, \boldsymbol{\phi}_i)((\mathbf{z}_i - h(\hat{\mathbf{x}}_i))(\mathbf{z}_i - h(\hat{\mathbf{x}}_i))^\top - H\hat{\boldsymbol{\Sigma}}_i H^\top)$
    $n \leftarrow n + k(\boldsymbol{\phi}, \boldsymbol{\phi}_i)$
  end for
  $\hat{\mathbf{R}} \leftarrow \frac{1}{n}\hat{\mathbf{R}}$
  return $\hat{\mathbf{R}}$
end function

III. EXPERIMENTAL RESULTS

The expectation-maximizing learning process was first validated in simulation on a toy problem representative of target problem domains, and then validated on real-world domains.

A. Simulation

We consider a fictional robot taking random steps, drawn from a Gaussian distribution of known covariance, around a room of varying brightness. After each step, the robot takes a position measurement, which is corrupted by Gaussian noise. The fictional position sensor performs well in the light, but poorly in the darkness; as the robot wanders around the room, it records both a noisy measurement of position and a predictor vector consisting of the observed brightness, and the direction of the nearest light source. This is a linear Gaussian system, and therefore if we know the measurement covariances we may exactly solve for the least-squares optimal solution using a standard Kalman filter. This exact solution is used as a baseline for comparison in all simulation experiments, as it represents the best possible estimate given the data available, and provides a lower bound on for the possible filter error.

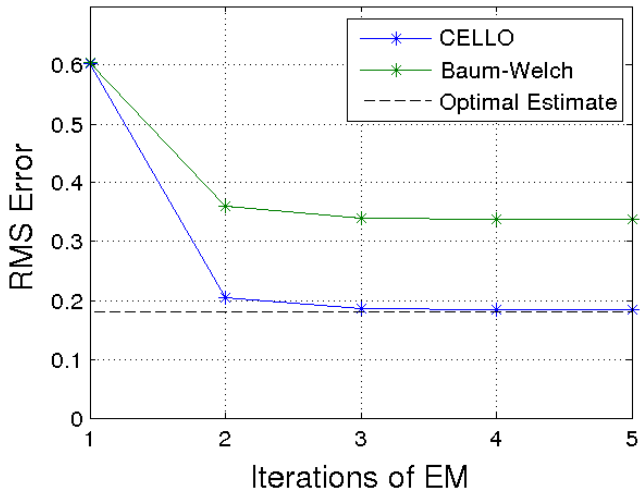


Fig. 4: *Root mean squared error between the estimated and true trajectories, using covariances generated by CELLO and the Baum-Welch algorithm. Both algorithms converge after only a few iterations, but due to the varying covariance Baum-Welch fails to achieve optimality. CELLO converges to the optimum, and results in half the mean error.*

We process a simulated dataset as described in algorithm 1, initializing the covariance to a fixed estimate and iteratively improving that estimate through cycles of Kalman filtering and CELLO maximization. We run the Baum-Welch algorithm [2] on the same dataset, assuming a fixed measurement covariance. The resulting mean squared error for each algorithm is shown in fig. 4. Because we have access to the true measurement covariance used to generate the data, we may provide a lower bound on filter performance; the measurement and transition distributions are linear Gaussian and therefore the Kalman filter with the true covariances yields the least-squares optimal solution. The resulting error using both CELLO and Baum-Welch rapidly converges to a minimum; however, the Baum-Welch model cannot capture variations in covariance, and thus remains suboptimal even after hundreds of iterations.

The trajectories estimated by CELLO, Baum-Welch, and the initial fixed covariance are presented in fig. 6, and compared to the optimal estimate. In the dark, both CELLO and the fixed covariance estimate perform similarly, trusting the dynamics model more than the sensor measurements, resulting in very smooth trajectories. In the light regions, however, CELLO is able to recover the random motions of the vehicle from the sensor data, while the fixed covariance estimator remains smooth, placing too much confidence in the system dynamics model over the sensor measurements.

These trends are reflected in the absolute errors for each estimate, presented in fig. 5. In the dark areas, CELLO, Baum-Welch, and the optimal estimator perform comparably. But the performance of the fixed covariance estimate fails to improve in the regions of high information, resulting in large peaks in the error.

B. Real-world performance

As previously observed [1, 6], environmental ambiguities lead planar range-finding systems to suffer from widely

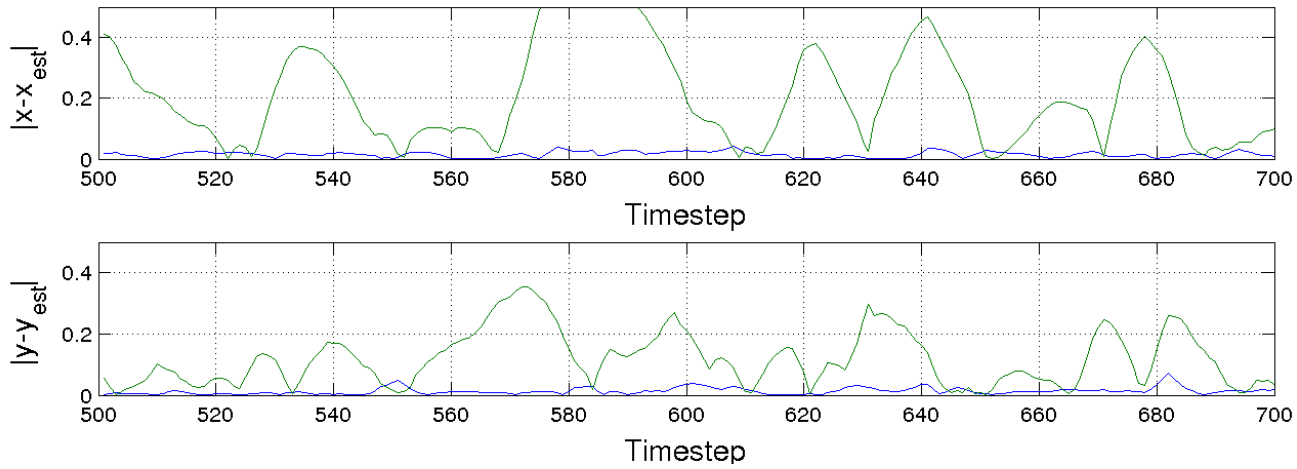


Fig. 5: *Comparison of estimation error magnitude using CELLO (blue) and Baum-Welch (green) in each dimension. Error is relative to the optimal estimate; note that when there is little information available, all systems perform comparably, but when Baum-Welch fails to incorporate information when it is available, leading to degraded performance relative to the optimum. CELLO performs comparably to the optimal estimator regardless of information quality.*

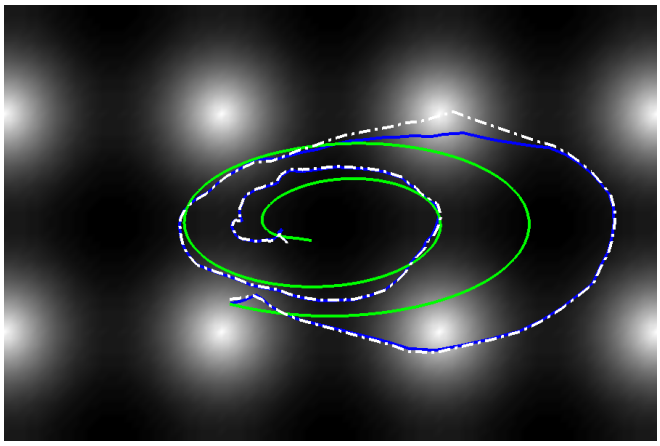


Fig. 6: *Estimated trajectory using CELLO (blue), Baum-Welch (green), and the true measurement covariances (white). Trajectories are presented in front of the light field indicating sensor quality: the sensor degrades in the dark. Note that using the true covariances gives an upper bound on estimator performance; CELLO nearly achieves this upper bound, while the Baum-Welch covariances fail to do so, especially in the dark areas.*

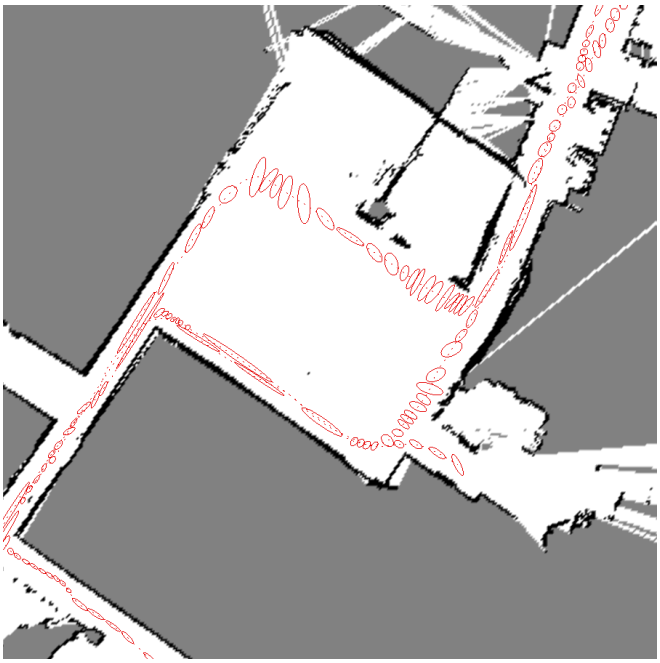


Fig. 7: *Predicted covariances for laser scan-matching odometry in a hallway. Note that environmental ambiguity creates large uncertainty along the axis of the hallway*

varying covariances. Such units are extremely common in mobile robotics; they have been used to build maps [7], to localize within a known map [8], or—by matching sequential scans—for motion estimation via dead reckoning [9].

We demonstrate the benefits of an adaptive covariance scheme through prediction of the covariances of the transform parameters between matched sequential simulated laser scans in a hallway environment. Our predictions were made using a predictor vector composed of histograms of angles returning viable ranges, and the angles formed by lines between sequential points. Sample predicted covariance ellipses are drawn in fig. 7. The predictions consistently align the covariance ellipse with the walls of the hallway whenever the far walls are out of range, reflecting the lack of information in the direction the scanner cannot see.

Figure 8 compares the predictions of our algorithm to those of Bachrach et al. [9] and Andrea Censi [6], along with hand-tuned and empirical fixed covariances. Although all three algorithms agree on the direction of maximal uncertainty and on the location of the regions of high uncertainty, they vary widely in the estimated magnitude of uncertainty.

To evaluate which uncertainty estimates most closely mirror the true uncertainty, we determine the vehicle trajectory to high accuracy using a longer-range laser, capable of seeing the ends of the hallways traversed. These longer-ranged laser scans are processed by the SLAM algorithm presented in M. Kaess et al. [10] to obtain accurate position estimates.

Table I compares filter performance using several metrics.

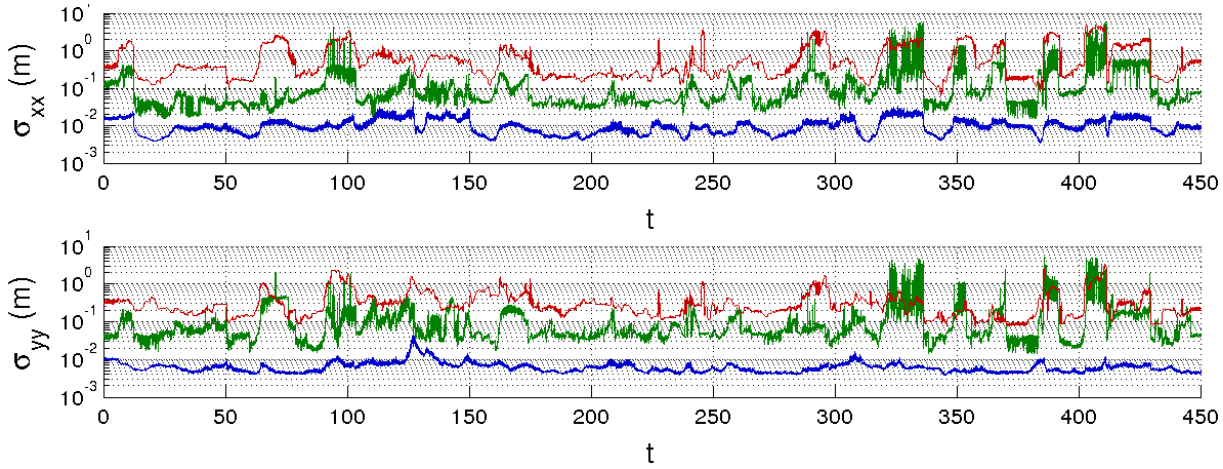


Fig. 8: *Comparison of the marginal covariances predicted by Bachrach et al. [9] (green), Andrea Censi [6] (blue), and CELLO (red). Note the agreement across methods on the locations of regions of uncertainty, but the wide disparity in their magnitude. Using the smaller covariances results in the high variance velocity estimates seen in fig. 9; in particular, note that the spikes in velocity variance using Bachrach's covariance scheme coincide with regions where the covariance increases, suggesting that only CELLO sufficiently increases the covariance.*

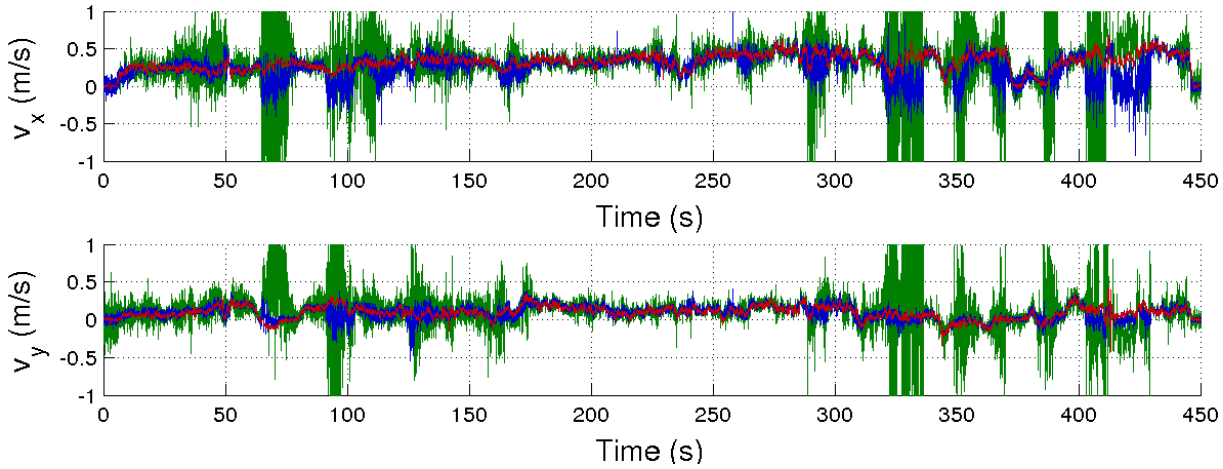


Fig. 9: *Estimated velocity using covariances predicted by CELLO (red), Bachrach (green), Censi (blue), and fixed to hand-tuned values (purple). Note the reduced variance when using CELLO.*

Filters are evaluated for each covariance scheme, using both a planar laser scan matcher and an RGB-D optical flow algorithm for odometry. We additionally present filtering results using just the optical flow and just the scan matcher. The availability of both sensors reduces the mean error in each case, as expected; however, it diminishes filter consistency, as measured by the normalized estimation error and the dataset likelihood. Using CELLO yields the smallest mean error, as well as the highest consistency for either metric.

IV. Related Work

The problem of estimating sensor covariances has attracted attention for many years. One early attempt was the adaptive Kalman filter [12], which modifies the elements of the measurement noise covariance and the process noise covariance on-line. This method relies purely on local noise characteristics; as such, it can guarantee that the covariances it generates are near the true covariances, but changes in the noise parameters will always be delayed while adaption occurs. This greatly reduces its effectiveness in cases of abrupt dramatic changes, as in outlier rejection.

There have many attempts to learn the uncertainty of individual perception and odometry algorithms. Andrea Censi [6] developed a scheme based on information-theoretic analysis for estimating the uncertainty of laser scan matching odometry through the iterative closest point algorithm. Bachrach et al. [9] developed an alternate scheme by fitting a covariance ellipse to the three-dimensional transform likelihood map. Brenna [13] presented extensive work on transform covariance estimation for laser scan matching, while Huang et al. [14] developed a prediction scheme for visual odometry. These algorithms all share similar disadvantages. Because they were designed to estimate the uncertainty in the output of a single algorithm, they cannot be generalized to other sensors. This implies that using different sensors, or different algorithms to process sensor data, requires the generation of entirely new uncertainty models, which represents a substantial investment in time and effort. In contrast, CELLO-EM learns its uncertainty model from data, and integrating

	RMSE[1]	MAE[2]	NEES[3]	NMEE[4]	LL[5]
Fixed	0.0303	9.1797	34.7087	1.5950	-11.0200
BSM	0.0595	18.3735	464.3963	5.1583	-356.8934
CSM	0.0407	11.9108	7025.3429	33.0318	3.1346
FOVIS	0.2397	63.0931	14431.6146	29.2197	-712.3076
BSM+FOVIS	0.0199	6.4020	1234.3538	5.3928	-620.1783
CSM+FOVIS	0.0323	9.6486	6739.2444	25.8061	-139.8690
CELLO	0.0179	6.0304	19.0661	0.9536	6.1715

TABLE I: *Comparison of filter performance for a laser scan-matcher and optical flow system. The filter was tested using fixed covariances, using Bachrach's scan matcher alone, using Censi's scan matcher alone, using the FOVIS optical flow system alone, using Bachrach's scan matcher in conjunction with FOVIS, using Censi's scan matcher in conjunction with FOVIS, and using Bachrach's scan matcher and FOVIS with covariances predicted by CELLO. Using CELLO results in modest gains in terms of absolute accuracy, but enormous gains in consistency and estimator bias. All metrics are taken as specified in Bar-Shalom et al. [11]*

[1] Root mean squared error, $\sqrt{\frac{1}{K}\sum_{n=1}^{K}(\hat{\mathbf{x}}_n - \mathbf{x}_n)^\top(\hat{\mathbf{x}}_n - \mathbf{x}_n)}$. Low values indicate an accurate estimator; lower is better.

[2] Mean absolute error, $\frac{1}{K}\sum_{n=1}^{K}\sum_{i=1}^{N_x}|(x_i)_n|$. Low values indicate an accurate estimator; lower is better.

[3] Normalized estimation error squared, $\frac{1}{K*N_x}\sum_{n=1}^{K}(\hat{\mathbf{x}}_n - \mathbf{x}_n)^\top \mathbf{\Sigma}_n^{-1}(\hat{\mathbf{x}}_n - \mathbf{x}_n)$. A lower value indicates a more consistent estimator.

[4] Normalized mean estimation error, $\frac{1}{K}\sum_{i=1}^{N_x}\left|\sum_{n=1}^{K}\frac{(\mathbf{x}_i)_n}{(\mathbf{\Sigma}_{ii})_n}\right|$. A value of zero implies an unbiased estimator for all states; lower is better.

[5] Normalized log likelihood, $\frac{1}{K}\sum_{i=1}^{K}\log p\left(\mathbf{z}_i|\mathbf{R}_i, \hat{\mathbf{\Sigma}}_0\right)$. High values indicate a good measurement model; higher is better.

a new sensor just requires collecting data and running the algorithm. This generality comes at negligible cost, as CELLO is competitive even with algorithms specialized to a single sensor.

Various nonparametric estimation schemes have been developed in recent years, both within and without the robotics community. Ko and Fox [15] employ Gaussian processes in a filtering context to learn not just the uncertainty parameters but the entire measurement model. Wilson and Ghahramani [16] presented a non-parametric Bayesian method for volatility estimation, for use in quantitative finance. Melkumyan and Ramos [17] describes a related method which uses Gaussian processes to estimate uncertainty. Because the methods of Wilson and Melkumyan require sampling to generate posterior distributions, they are computationally intractable in real-time domains; moreover, their formulations are designed to scale for high-dimensional predictions, rather for the high-dimensional feature vectors needed in mobile robotics. An alternative class of methods, such as that of Graham et al. [18], generates robustness to outliers while avoiding explicitly modeling the covariance by instead solving a convex optimization problem at each filtering step.

V. Conclusions

We have presented a method for predicting unobservable model parameters applicable to arbitrary models. Applying this method to the prediction of sensor measurement covariances yields an extension to our previous work, allowing for fast covariance predictions without access to ground truth. The method presented outperforms both hand-tuned fixed covariances and domain-specific algorithms, as measured by several metrics; importantly, it improves both estimator accuracy and consistency.

Although we only present results for the case of multivariate Gaussian measurements, there are other useful models that could benefit from this kind of analysis. In the context of sensor estimation, learning the state transition covariance, or process noise, could improve estimation fidelity in applications where there exists a good model which is known to break down, such as flying vehicles near stall. This method could also be applied to discrete estimation problems, if the transition and emission probabilities are suspected to vary. We believe it opens important avenues towards improving estimation of both state and uncertainty, and promises to be useful in many contexts.

References

[1] W. Vega-Brown, A. Bachrach, A. Bry, J. Kelly, and N. Roy, "CELLO: a fast algorithm for covariance estimation," in *Proc. ICRA*, Karlsruhe, Germany, 2013.

[2] L. E. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains," *The Annals of Mathematical Statistics*, vol. 41, no. 1, pp. 164–171, 1970.

[3] E. A. Nadaraya, "On estimating regression," *Theory of Probability and Its Applications*, vol. 9, no. 1, pp. 141–142, 1964.

[4] G. S. Watson, "Smooth regression analysis," *Sankhy: The Indian Journal of Statistics, Series A*, p. 359372, 1964.

[5] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society, Series B*, vol. 39, no. 1, p. 138, 1977.

[6] Andrea Censi, "An accurate closed-form estimate of ICP's covariance." in *Proc. ICRA*, Rome, Italy, 2007.

[7] H. Durrant-Whyte, S. Majumder, S. Thrun, M. de Battista, and S. Scheding, "A bayesian algorithm for simultaneous localisation and map building," in *Robotics Research*, ser. Springer Tracts in Advanced Robotics. Springer Berlin / Heidelberg, 2003, vol. 6, pp. 49–60.

[8] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, "Monte carlo localization for mobile robots," in *Proc. ICRA*, 1999.

[9] A. Bachrach, S. Prentice, R. He, and N. Roy, "RANGE - robust autonomous navigation in GPS-denied environments," *Journal of Field Robotics*, vol. 28, no. 5, pp. 644–666, 2011.

[10] M. Kaess, A. Ranganathan, and F. Dellaert, "iSAM: incremental smoothing and mapping," *IEEE Trans. on Robotics, TRO*, vol. 24, no. 6, pp. 1365–1378, 2008.

[11] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. John Wiley and Sons, Inc, 2001.

[12] R. K. Mehra, "On the identification of variances and adaptive kalman filtering," *IEEE Transactions on Automatic Control*, vol. AC-15, pp. 175–184, 1970.

[13] M. Brenna, "Scan matching covariance estimation and SLAM: models and solutions for the scanSLAM algorithm," Ph.D. dissertation, Artificial Intelligence and Robotics Laboratory Politecnico di Milano, 2009.

[14] A. S. Huang, A. Bachrach, P. Henry, M. Krainin, D. Maturana, D. Fox, and N. Roy, "Visual odometry and mapping for autonomous flight using an RGB-D camera," in *Proc. ICRA*, Flagstaff, AZ, 2011.

[15] J. Ko and D. Fox, "GP-BayesFilters: bayesian filtering using gaussian process prediction and observation models," *Auton. Robots*, vol. 27, no. 1, p. 7590, 2009.

[16] A. G. Wilson and Z. Ghahramani, "Generalised wishart processes," *Uncertainty in Artificial Intelligence*, 2011.

[17] A. Melkumyan and F. Ramos, "Multi-kernel gaussian processes," *Proc. JCAI*, pp. 1408–1413, 2011.

[18] M. Graham, T. Steiner, and J. How, "Robust vision-aided navigation in urban environments," in *Proc. AIAA GNC*, Boston, MA, 2013.