Online Self-Calibration For Mobile Robots

Nicholas Roy and Sebastian Thrun
Robotics Institute and Computer Science Department
Carnegie Mellon University
Pittsburgh, PA 15232

Abstract

This paper proposes a statistical method for calibrating the odometry of mobile robots. In contrast to previous approaches, which require explicit measurements of actual motion when calibrating a robot's odometry, the algorithm proposed here uses the robot's sensors to automatically calibrate the robot as it operates. An efficient, incremental maximum likelihood algorithm enables the robot to adapt to changes in its kinematics on-line, as they occur. The appropriateness of the approach is demonstrated in two large-scale environments, where the amount of odometric error is reduced by an order of magnitude.

1 Introduction

Calibration is the problem of estimating a robot's physical model from data. It has long been recognized that robots change their physical properties over time. In mobile robotics, wear and tear can change the diameter of wheels, loosen belts, and so on. Such effects can introduce significant systematic errors into a robot's odometry. The need for such calibration is as old as the field of robotics itself, and the literature is full of methods for calibrating robots (see e.g., [AAH88, CW90, Vuk89]). As examples shown elsewhere illustrate, the resulting errors can be substantial [Bor94, BEF96, KB91, Ren93, Thr98b].

Virtually all existing calibration methods, however, have certain disadvantages when applied to mobile robots. Many existing calibration methods require human intervention. To calibrate a mobile robot's odometry, a person (or some external device) has to measure the exact motion of the robot, and infer from these measurements the physical model. Such approaches are undesirable for two reasons. First, a certain amount of effort is involved in calibrating a mobile robot, usually disrupting the robot's operation. Second, and more importantly, the physics of mobile robots change, often rapidly. For robot arms, robot platforms and many other stationary devices, the environment is mostly static, thus the calibration parameters are unlikely to change except with changes to the robot. Furthermore, the odometric error of a robot arm joint is strictly internal to the joint and is not affected by most changes to the environment.

By comparison, a mobile robot's odometry is dependent on the kind of surface the robot is travelling on. As the surface changes (from carpet to tile, for example) the calibration parameters change. To maintain an up-to-date model, the calibration has to be repeated at regular intervals. This can be expensive in practical applications—such as robots that are operated in private homes.

Consequently, rather than performing position estimation solely based on odometry data (dead-reckoning), mobile robots typically combine odometry data with sensor feedback from the environment. In such a localization process, the position of the robot is estimated from both uncalibrated odometry and sensor data such as from a laser proximity sensor [BFT97, MD94], eliminating the need for a model of the odometric error.

One problem, however, with this type of position estimation is that in areas of the environment with little or no sensor data, such as large open spaces, or unreliable sensor data (such as in crowded environments) the sensor feedback becomes unreliable or nonexistent, and the robot can become quickly lost.

This paper proposes a method that calibrates a robot's odometers continuously during its everyday operation, without human intervention. The method automatically adapts to changes that might occur over the lifetime of the robot. Our research is based on the observation that mobile robots can use their sensors to estimate their calibration parameters as they move, in areas of the environment where the sensor data is available and uncorrupted by such problems as dynamic environments. This allows the robot to perform more reliable position estimation in areas of the environment where sensor data cannot be used for localization. In particular, we assume that the robot possesses two sensor systems: odometry (e.g., wheel encoders or gyroscopes) and proximity sensors (e.g., sonar sensors, laser range finders).

Robot calibration is not a novel idea; however, few calibration techniques have been applied to mobile robot odometry. Furthermore, our approach do not require any specialized hardware, such as in the approach of Borenstein et al. [BEF96]. We address the problem of odometric calibration through statistical means, using existing sensors. Our approach phrases the calibration problem as a maximum likelihood estimation problem, which seeks to identify the most likely model parameters under the data. While the general maximum likelihood estimation problem is intractable, we have devised an efficient, incremental solution. The estimator proposed here is an exponential estimator that determines calibration parameters through iterative comparisons of pairs of sensor readings. It has a variety of properties that make it well-suited for the lifelong, continuous calibration problem. For example, it does not require the memorization of past data, and it can adapt to changes in the physical properties of the robot.

Experimental results, obtained in two large, populated indoor environments, demonstrate the appropriateness of the approach. The odometric error is reduced by approximately 83%, along trajectories 741m and 269m long.

2 Probabilistic Model of the Kinematics

Our approach models robot motion probabilistically. More specifically, let $\xi = \langle x, y, \theta \rangle$ denote a robot's pose in x-y-space (θ is the robot's heading direction). The model of robot motion is denoted by the conditional probability distribution $P(\xi' | \xi, o)$, where ξ is the robot's pose before executing a control (action), o is the displacement measured by the robot's odometry, and ξ' is the pose after executing the control. To simplify the notation, we will assume that odometric measurements o consist of two numbers, one that measures the robot's rotational displacement (denoted by $o_{\rm rot}$), and one that measures its translational displacement (denoted by $o_{\rm trans}$).

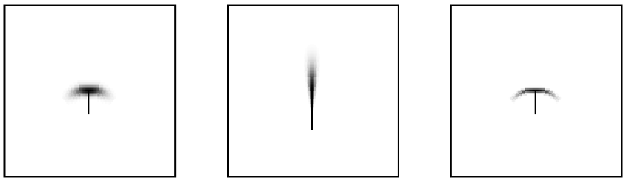

Figure 1: Three example probability distributions of the robot's (x, y) position after a rotation and a forward translation. Figure (b) shows a model with high translational error, and figure (c) shows a model with high rotational error.

Figure 1 illustrates a specific $P(\xi' | \xi, o)$. Here the robot's initial pose, ξ, is shown at the bottom. The shaded grey area depicts the distribution over possible posterior poses, $P(\xi' | \xi, o)$, after measuring that the robot moved as indicated. The darker a value, the more likely it is. Figure 1(b) shows a motion model that assumes high translational error, and figure 1(c) shows a situation with excessive rotational error.

Mathematically speaking, the motion model is defined through the robot's kinematics with the assumption that the robot might non-deterministically suffer errors in its translational and rotational measurements. The robot's final pose is given by

$$\begin{pmatrix} x' \\ y' \\ \theta' \end{pmatrix} = \begin{pmatrix} x + \hat{o}_{\rm trans} \cos(\theta + \hat{o}_{\rm rot}) \\ y + \hat{o}_{\rm trans} \sin(\theta + \hat{o}_{\rm rot}) \\ (\theta + \hat{o}_{\rm rot}) \text{ modulo } 2\pi \end{pmatrix} \quad (1)$$

where $\hat{o}_{\rm trans}$ and $\hat{o}_{\rm rot}$ denote the robot's *true* translation and rotation, respectively. Recall that $o = \langle o_{\rm trans}, o_{\rm rot} \rangle$ is the displacement *measured* by the robot. If the robot's odometry is 100% accurate, $\hat{o}_{\rm trans} = o_{\rm trans}$ and $\hat{o}_{\rm rot} = o_{\rm rot}$, and there is no calibration problem. In practice, however, the measured and actual odometry differ.

In this paper we assume that the difference is accounted for by two factors: a *systematic error* and a *random error*, where the latter has an expected value of zero (zero-mean). More specifically, the true rotation and translation differ from the measured odometry by two additive terms

$$\begin{aligned} \hat{o}_{\rm trans} &= o_{\rm trans} + \delta_{\rm trans} |d| + \varepsilon_{\rm trans} \\ \hat{o}_{\rm rot} &= o_{\rm rot} + \delta_{\rm rot} |d| + \varepsilon_{\rm rot} \end{aligned} \quad (2)$$

Here $\varepsilon_{\rm trans}$ and $\varepsilon_{\rm rot}$ are two random variables with zero mean. The numerical parameters $\delta_{\rm trans}$ and $\delta_{\rm rot}$ describe the systematic error, the drift. The problem of robot calibration, thus, is the problem of estimating $\delta_{\rm trans}$ and $\delta_{\rm rot}$.

As the functional form (2) suggests, our model assumes that both errors grow linearly with the distance traveled. In practice we found this model to be superior over various other choices, including models with more parameters. Our choice of parameters was heavily influenced in fact by experimental evidence. The odometric error that the robot accumulated was almost completely attributable to translational motion. The controller used for this robot performed very little pure rotation; most paths were curved, combining rotation and forward motion. The error occurred almost exclusively during these kinds of trajectories, supporting our use of error parameters only over the translational motion.

3 Parameter Estimation

Our approach estimates the kinematic parameters, $\delta_{\rm trans}$ and $\delta_{\rm rot}$ using data collected during everyday robot motion. Let

$$d = \{s^{(1)}, o^{(1)}, s^{(2)}, o^{(2)}, \ldots\} \quad (3)$$

denote the data, where $s^{(i)}$ denotes a sensor measurement (e.g., a laser scan), and $o^{(i)}$ denotes the displacement measured by the robot's odometry between two consecutive sensor measurement.

In statistical terms, the calibration problem is a maximum likelihood estimation problem where one seeks to identify the kinematic parameters $\delta_{\rm trans}$ and $\delta_{\rm rot}$ that appear most plausible under the data d:

$$\langle \delta^*_{\rm trans}, \delta^*_{\rm rot} \rangle \quad = \quad \mathop{\rm argmax}_{\delta_{\rm trans}, \delta_{\rm rot}} P(\delta_{\rm trans}, \delta_{\rm rot} | d) \qquad (4)$$

If the data set is large, this problem is mathematically intractable (see [TFB98]). In addition, computing (4) would require that the robot memorized all data d, which is undesirable for a robot that calibrates its motion parameters continuously.

Instead, our approach decomposes the estimation problem into a sequence of single-step problems, which can be estimated much more efficiently:

$$\langle \delta^{(i)*}_{\rm trans}, \delta^{(i)*}_{\rm rot} \rangle = \mathop{\rm argmax}_{\delta_{\rm trans}, \delta_{\rm rot}} P(\delta_{\rm trans}, \delta_{\rm rot} | s^{(i)}, o^{(i)}, s^{(i+1)}) \qquad (5)$$

Here i is a time index. This series of local maximum likelihood estimators determines the motion parameters $\delta^*_{\rm trans}$ and $\delta^*_{\rm rot}$ based on data just perceived. More specifically, the estimator considers only the sensor data obtained *before* the transition, $s^{(i)}$, sensor data obtained after the transition, $s^{(i+1)}$, and the displacement measured by the robot's odometry $o^{(i)}$. The probability on the right-hand side of (5) is called the *parameter likelihood function*. It will be derived in the next section.

Finally, the desired kinematic parameters, $\delta_{\rm trans}$ and $\delta_{\rm rot}$, are estimated recursively using an *exponential estimator*, where past estimates are discounted exponentially over time:

$$\left(\begin{array}{c} \delta^*_{\rm trans} \\ \delta^*_{\rm rot} \end{array} \right) \longleftarrow \gamma \left(\begin{array}{c} \delta^*_{\rm trans} \\ \delta^*_{\rm rot} \end{array} \right) + (1-\gamma) \left(\begin{array}{c} \delta^{(i)*}_{\rm trans} \\ \delta^{(i)*}_{\rm rot} \end{array} \right) \qquad (6)$$

Here $\gamma \lessapprox 1$ is an exponential discount factor that decays the weight of measurements over time. This exponential estimator has three important advantages over the original maximum likelihood estimator (4):

- it is incremental, i.e., it does not require that the robot memorizes past data,
- it can be computed in constant time, independent of the data set size d, and
- it adapts to changes in the robot's drift, by exponentially decaying past measurements.

Thus, with an appropriate choice of γ, it can be used for continuously calibrating a robot whose drift parameters change slowly over time, e.g., with wear and tear. In our experiments, we used $\gamma = 0.9$.

4 Parameter Likelihood Function

It remains to be shown how to compute the parameter likelihood function $P(\delta_{\rm trans}, \delta_{\rm rot} | s^{(i)}, o^{(i)}, s^{(i+1)})$ in equation (5). Since the parameters are estimated based on actual sensor data (e.g., laser range measurements), the parameter likelihood function involves the definition of a sensor model.

According to Bayes' rule, the parameter likelihood can be transformed into:

$$\begin{aligned} & P(\delta_{\rm trans}, \delta_{\rm rot} | s^{(i)}, o^{(i)}, s^{(i+1)}) \quad = \qquad (7) \\ & \alpha P(s^{(i+1)} | s^{(i)}, o^{(i)}, \delta_{\rm trans}, \delta_{\rm rot}) P(\delta_{\rm trans}, \delta_{\rm rot} | s^{(i)}, o^{(i)}), \end{aligned}$$

where α is a constant normalizer that can safely be ignored in the maximization. Since knowledge of just $s^{(i)}$ and $o^{(i)}$ (without $s^{(i+1)}$!) does not convey any information about the parameters $\delta_{\rm trans}$ and $\delta_{\rm rot}$, $P(\delta_{\rm trans}, \delta_{\rm rot} | s^{(i)}, o^{(i)}) = P(\delta_{\rm trans}, \delta_{\rm rot})$, and equation (7) can further be simplified to

$$\begin{aligned} & P(\delta_{\rm trans}, \delta_{\rm rot} | s^{(i)}, o^{(i)}, s^{(i+1)}) \quad = \qquad (8) \\ & \qquad \alpha P(s^{(i+1)} | s^{(i)}, o^{(i)}, \delta_{\rm trans}, \delta_{\rm rot}) \;\; P(\delta_{\rm trans}, \delta_{\rm rot}). \end{aligned}$$

The probability $P(\delta_{\rm trans}, \delta_{\rm rot})$ is the *prior* on the parameters $\delta_{\rm trans}$ and $\delta_{\rm rot}$. Typically, one might have a Gaussian or a uniform prior on the drift parameters.

The other term in equation 8, the probability $P(s^{(i+1)} | s^{(i)}, o^{(i)}, \delta_{\rm trans}, \delta_{\rm rot})$ is called the *perceptual likelihood*. It specifies the likelihood of observing $s^{(i+1)}$ under the assumptions that

- the robot initially observed $s^{(i)}$,
- then measured an odometric displacement $o^{(i)}$,
- but its odometry was corrupted according to $\delta_{\rm trans}$ and $\delta_{\rm rot}$.

5 The Perceptual Likelihood

It remains to show how to compute the perceptual likelihood. According to the theorem of total probability (and under some obvious independence assumptions), the perceptual likelihood can be expressed as

$$\begin{aligned} & P(s^{(i+1)} | s^{(i)}, o^{(i)}, \delta_{\rm trans}, \delta_{\rm rot}) \\ & \quad = \int\!\!\int P(s^{(i+1)} | W, \Delta\xi, s^{(i)}, o^{(i)}, \delta_{\rm trans}, \delta_{\rm rot}) \cdot \\ & \qquad\qquad P(W, \Delta\xi | s^{(i)}, o^{(i)}, \delta_{\rm trans}, \delta_{\rm rot}) \; dW \; d\Delta\xi \\ & \quad = \int\!\!\int P(s^{(i+1)} | W, \Delta\xi) \cdot \qquad (9) \\ & \qquad\qquad P(W | s^{(i)}) P(\Delta\xi | o^{(i)}, \delta_{\rm trans}, \delta_{\rm rot}) \; dW \; d\Delta\xi \end{aligned}$$

where W denotes the *world*, the configuration of all obstacles, and $\Delta\xi$ denotes the relative displacement between the robot's pose $\xi^{(i+1)}$ and $\xi^{(i)}$. Of course, integrating over all possible worlds W and all displacements $\Delta\xi$ is infeasible.

Our approach approximates the perceptual likelihood by replacing the integrals in (9) with their expected values, which are much easier to compute (as the need to integrate over W and $\Delta\xi$ is obviated):

$$\begin{aligned} &P(s^{(i+1)}|s^{(i)}, o^{(i)}, \delta_{\text{trans}}, \delta_{\text{rot}}) \qquad (10)\\ &\approx \quad P(s^{(i+1)}|W{=}E[W|s^{(i)}], \Delta\xi{=}E[\Delta\xi|o^{(i)}, \delta_{\text{trans}}, \delta_{\text{rot}}]) \end{aligned}$$

Here $E[\,]$ denotes the (conditional) expected value of a random variable. This expression is only approximately correct, but can be computed efficiently (whereas the original expression cannot). In our implementation, it is computed in three steps, each of which correspond to one of the terms in (10).

1. $E[W|s^{(i)}]$: First, the initial sensor scan $s^{(i)}$ is transformed into an occupancy grid [Elf87, Mor88, BK91]. This occupancy grid describes the *expected* world W under the sensor scan [Thr98b]. Figure 2 depicts an example occupancy grid. The technique for generating this occupancy grid has been adopted from the literature.

Figure 2: An example occupancy grid. The light areas are high-probability occupancy, and the dark areas are low-probability occupancy. The mid-grey areas are 50% occupancy, indicating no information about these cells.

2. $E[\Delta\xi|o^{(i)}, \delta_{\text{trans}}, \delta_{\text{rot}}]$: The expected relative pose $\Delta\xi$ is obtained by computing the expected pose at time $i+1$ relative to the pose at time i, using the kinematic motion equations (1) and (2). Notice that we explicitly exploit the fact that $\varepsilon_{\text{trans}}$ and ε_{rot} have zero mean and can thus safely be ignored in the computation of the expected relative pose.

3. $P(s^{(i+1)}|W, \Delta\xi)$: Finally, the likelihood of each individual sensor measurement in $s^{(i+1)}$, the scan recorded in the final position, is computed using a geometric sensor model adopted from [BFHS96a]. Let $s_k^{(i+1)}$ be the k-th individual sensor value (a single distance measurement) in the sensor scan $s^{(i+1)}$. The conditional probability of this measurement, $P(s_k^{(i+1)}|s^{(i)}, o^{(i)}, \delta_{\text{trans}}, \delta_{\text{rot}})$, is obtained by ray tracing [FvDFH90], where the likelihood of a "hit" depends on the occupancy probability of the grid cell that is being traced. As a result, sensor measurements that match the occupancy map will have high likelihood, whereas measurements that contradict the occupancy map have low likelihood. Assuming conditional independence between the different measurements (i.e., the noise in the measurements is independent), the desired probability is obtained as the product of the individual sensor probabilities:

$$P(s^{(i+1)}|W, \Delta\xi) \;=\; \prod_k P(s_k^{(i+1)}|W, \Delta\xi) \qquad (11)$$

See [BFHS96a] for a more detailed description.

1. Acquire a sensor scan, $s^{(i)}$.
2. Update the occupancy grid with $s^{(i)}$.
3. Move to a new location, and record odometry measurement $o^{(i)}$.
4. Acquire a second sensor scan, $s^{(i+1)}$.
5. For each possible position error $\langle\delta_{\text{trans}}, \delta_{\text{rot}}\rangle$, compute the probability of new data given potential pose, $P(s^{(i+1)}|s^{(i)}, o^{(i)}, \delta_{\text{trans}}, \delta_{\text{rot}})$
6. Choose most likely $\langle\delta_{\text{trans}}^{(i)*}, \delta_{\text{rot}}^{(i)*}\rangle$ from maximum likelihood given by argmax $P(s^{(i+1)}|s^{(i)}, o^{(i)}, \delta_{\text{trans}}^{(i)*}, \delta_{\text{rot}}^{(i)*})$
7. Compute new global $\langle\delta_{\text{trans}}^{*}, \delta_{\text{rot}}^{*}\rangle$ from $\langle\delta_{\text{trans}}^{(i)*}, \delta_{\text{rot}}^{(i)*}\rangle$ and previous global $\langle\delta_{\text{trans}}^{*}, \delta_{\text{rot}}^{*}\rangle$
8. Update position as $\xi' = \xi + \begin{pmatrix} o_{\text{trans}} \\ o_{\text{rot}} \end{pmatrix} + \begin{pmatrix} \delta_{\text{trans}}^{*} \\ \delta_{\text{rot}}^{*} \end{pmatrix} \cdot |d|$
9. Set $s^{(i)} = s^{(i+1)}$, and repeat from step 2

Figure 3: The algorithm for computing the probabilistic error model in pose estimates using Bayes' Rule.

Figure 3 summarizes the parameter estimation algorithm, based on the parameter likelihood function computed from the perceptual likelihood.

6 Experimental Results

6.1 Overview

Our approach was tested using the RWI B21 robot shown in figure 4. The robot is equipped with a 4-wheel synchro drive, an array of 24 sonar sensors, and a SICK laser range finder. The datasets used in our evaluations were collected in two museums: the Carnegie Museum of Natural History

Figure 4: The RWI B21 robot used in our research.

in Pittsburgh, PA, and the Smithsonian National Museum of American History in Washington, DC. In both datasets, people occasionally blocked the robot's sensors.

	Carnegie Museum	Smithsonian Museum
Path Length	269 m	741 m
Raw Odometry Error	18.0 m	69.7 m
Corrected Odometry Error	3.05 m	12.25 m

Table 1: Summary of errors for raw and corrected odometry in the two museums.

The basic result of our evaluation is that the approach presented here improves the robot's odometry by an order of magnitude. As the results in table 1 indicate, the final odometric error in two extensive runs was 18.0 m, or 69.7m, which was reduced by our algorithm to 3.05m, or 12.45m, respectively. Thus, our approach reduces the odometric error by 83.1%, or 82.4%, by automatically calibrating the kinematic model as the robot is in operation.[1]

6.2 Single Step Calibration

Figures 5 and 6 illustrate the basic estimation step in our algorithm. Figure 5 shows an example range scan in the left panel. The right panel shows a second scan, superimposed using the raw odometry measurement. As is easy to be seen, the scans do not align properly if the robot's raw odometry is used. The result of applying our calibration algorithm to this pair of sensor scans is shown in Figure 6. Here the superimposed scans line up much better. This example illustrates a single step in the estimation of robot's motion parameters.

[1]These results are correspond to the results of similar efforts reported elsewhere [Bor94], but instead of changing the robot's hardware (the approach in [Bor94] actually requires that the robot has a trailer), our approach uses the robot's sensors to identify systematic errors in the robot's kinematics.

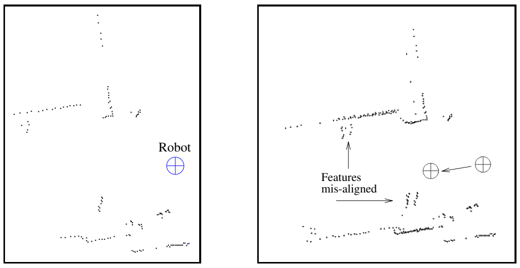

Figure 5: An example sensor scan, on the left. The points represent obstacles, and the circle is the robot position. A second scan acquired after a motion is on the right, superimposed on the first. Note the misalignment after the motion, which is caused mainly by systematic drift.

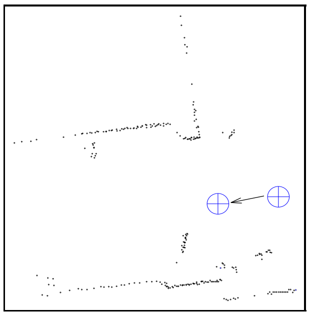

Figure 6: The superposition of the two sensor scans after calibrating the robot's odometry. Here the scans line up much better.

6.3 Results Obtained in the Carnegie Museum

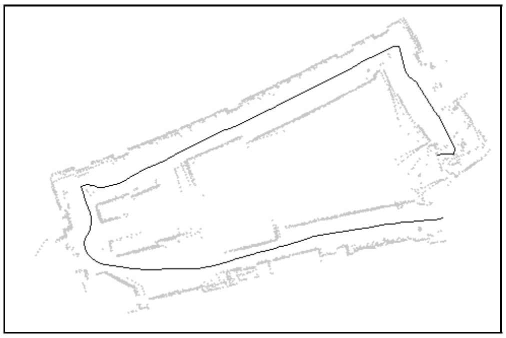

Figure 7: Path of the robot, using the uncalibrated raw odometry data. Shown in gray are the obstacles, as detected by the robot's laser range finder.

A more extensive experiment is shown in figure 7. This diagram shows a fraction of the dataset gathered in the Carnegie Museum of Natural History. As the diagram indicates, the error in the robot's odometry, if uncalibrated, is substantial (the path should be closed in this figure). After 269 meters (full dataset), the uncalibrated robot has accumulated an odometric error of 18.0 meters.

Figures 8 and 9 show the parameter estimations as a function of time, as estimated by our algorithm. Each

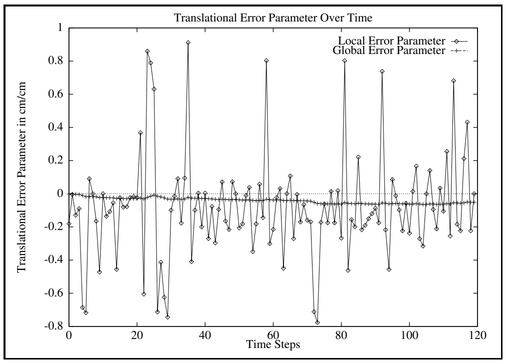

Figure 8: The parameter estimates $\delta^{(i)*}_{\tt trans}$ and $\delta^{*}_{\tt trans}$ at each iteration.

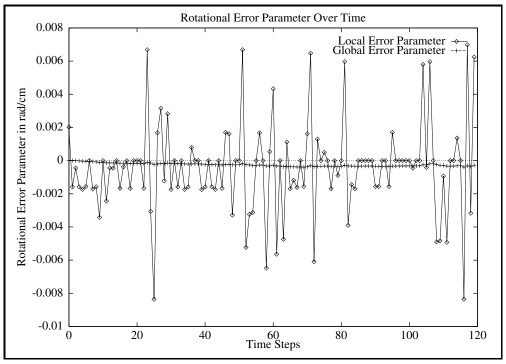

Figure 9: The parameter estimates $\delta^{(i)*}_{\tt rot}$ and $\delta^{*}_{\tt rot}$ at each iteration.

of these plots corresponds to one of the two parameters ($\delta_{\rm trans}$ and $\delta_{\rm rot}$). In each diagram, one curve shows the local maximum likelihood estimates $\delta^{(i)*}_{\rm trans}$ and $\delta^{(i)*}_{\rm rot}$ (c.f., equation (5)), whereas the other curve shows the exponentially smoothed estimates $\delta^{*}_{\rm trans}$ and $\delta^{*}_{\rm rot}$ (c.f., equation (6)). Noise in the robot's measurements make each individual local estimation, $\delta^{(i)*}_{\rm trans}$ and $\delta^{(i)*}_{\rm rot}$, fairly inaccurate, as the high variance of the measurements indicates. The exponential estimator, however, provides stable estimates for $\delta^{*}_{\rm trans}$ and $\delta^{*}_{\rm rot}$, the parameters used in the robot's calibrated motion model.

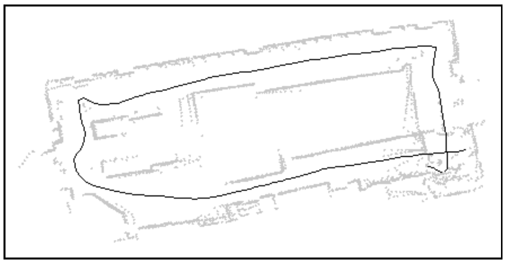

Figure 10: The path as computed with the calibration model.

Figure 10 shows the robot's path using the calibrated model. As is easy to be seen, the corrected model yields more accurate odometry. The error parameters at the end of the data set were $\delta_{\rm trans} = -.073$, and $\delta_{\rm rot} = -.0001607$. After 269 meters (full dataset), the final odometric error is only 3.05m, which amounts to a reduction of 83.1%.

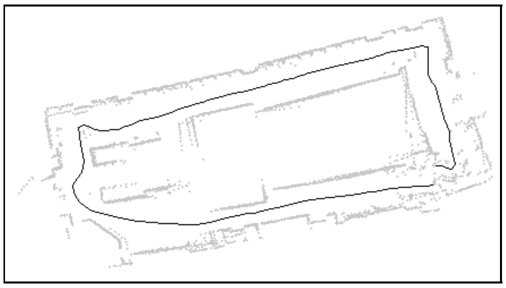

Figure 11: The map generated using corrected position estimates. The corrections were made using only the correction parameters $\delta_{\tt trans}$ and δ_r, computed from the entire data set.

The results shown in figure 10 are not as good as they could be, since the robot began its run with an uncalibrated model, and was only allowed to calibrate its motion model as it moved. Thus, the initial odometry is poor. In practice, the robot will have some initial, reasonable calibration, thereby reducing its error further. Figure 11 shows the result using a well-calibrated model throughout the entire experiment. Here the final error is even smaller.

6.4 Results Obtained in the Smithsonian Museum

To verify these results and further investigate the robustness of this approach, we applied the algorithm to a dataset that we recently collected in the National Museum of American History. In many aspects, this environment makes calibration more difficult. Most of this building consists of large, open spaces that lack the structure necessary for self-calibration (there are not many obstacles that the laser range finder could detect). It is also much larger, amplifying small rotational errors even more. Thus, in our experiments, the parameters $\delta_{\rm trans}$ and $\delta_{\rm rot}$ were initialized with the values obtained in the Carnegie Museum (and not just with 0, as in the previous experiment).

The results indicate that our approach is well-suited for calibrating the robot even in this environment. Figure 12 shows the path according to the raw, uncalibrated odometry — the robot trajectory is clearly unreliable. The start and end-point in the trajectory were in fact identical, and yet the end-point is located in the upper-right corner of the map in figure 12, whereas the start-point is in the lower-center part of the map. In metric terms, the final error is 69.7m, which resulted after a total motion of 741m.

Figure 13 shows the corrected path, estimated using

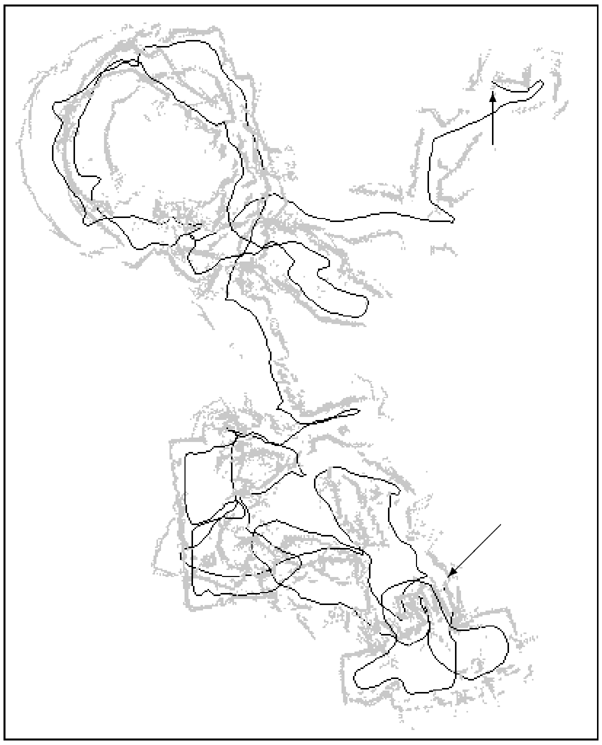

Figure 12: Path generated using the robot's raw, uncalibrated odometry, from data acquired in the Smithsonian National Museum of American History. The arrows point to the start and end positions of the robot, which correspond to the same point in the actual museum.

Figure 13: The map of the Smithsonian, generated with corrected position estimates, from correction parameters computed in the Carnegie Museum. Again the arrows correspond to the same point in the museum.

our new self-calibration routine. Here the odometry is much more accurate, reducing the final error substantially. The odometric error that resulted after this motion fell to 12.25m (from 69.7m) over the same 741m, a reduction in error of 82.4%. While these results have to be taken with a grain of salt — due to the high variance in real-world robot experiments — they nevertheless indicate the importance of on-line calibration in mobile robotics, and demonstrate the benefits of the work presented here.

7 Conclusion

This paper presented an algorithm for the life-long self-calibration of a mobile robot. The algorithm estimates kinematic calibration parameters by comparing consecutive sensor scans. The result of this comparison is used to adapt the kinematic model of the robot, thereby improving its odometry. The key advantage of this approach over previous calibration methods lies in the fact that it obviates the need for external measurements and explicit calibration procedures; instead, the robot calibrates itself while it is operating. The advantage to this approach is that the calibration parameters adapt to changes in the environment rapidly and without human intervention. Experimental results obtained in two large and irregularly shaped indoor environments illustrate that the algorithm can reduce a robot's odometry error significantly.

The statistical framework, on which our approach is based, relates closely to a family of recent statistical methods that have been applied with great success to various problems in mobile robotics. For example, similar statistical methods have been devised for mobile robot localization [MD94, NPB95, KCK96, SK95, BFHS96a, BFHS96b], landmark learning [Thr98a], mapping [LM97, GN97, SK97, KS96, TFB98, Thr98b], collision avoidance [FBT98] and motion planning [BFT97, Thr98b]. The method presented here demonstrates that statistical approaches are well-suited to calibrating a robot's model without external assistance. Since all of these approaches rely on odometry measurements, the algorithm proposed here should lead to superior results, if applied in combination.

To us, the results presented here have significant practical importance. We have successfully installed a mobile robotic tour-guide in the Deutsches Museum Bonn [BFL+97] and the Smithsonian National Museum of American History. The robot's task in these museums was to give tours to visitors of the museum. Accurate odometry was essential for the success of the robot, as many of the obstacles, especially in the Deutsches Museum Bonn were practically "invisible" to the robot's sensors. The datasets used in our experiments have been obtained in two much larger museums, one of which (the Smithsonian) not only has few reference points for our localization methods, but was a crowded environment where dynamic obstacles (i.e., people) corrupted the sensor data regularly.

Although this paper addresses exclusively the calibration of a robot's kinematic model, the statistical approach presented is much more general, in that it can be applied to a much broader range of parametric calibration problems. For example, with small modifications our approach can be used to calibrate a robot's sensors and to detect and accommodate sensor or actuator failures. Extending the approach to such problems, which we believe to have great practical importance for a long-running robot, is subject to future research.

Acknowledgments

The authors would like to thank Tom Mitchell for his advice and support of this research. We would also like to thank both the Carnegie Museum of Natural History, and the National Museum of American History for the use of their space for these experiments.

This research was supported in part by Le Fonds pour la Formation de Chercheurs et l'Aide à la Recherche (Fonds FCAR), Daimler-Benz Research (via Frieder Lohnert) and the Defense Advanced Research Projects Agency (DARPA) via the Airforce Missile System Command under contract number F04701-97-C-0022. The views and conclusions contained in this document are those of the author and should not be interpreted as necessarily representing official policies or endorsements, either expressed or implied, of FCAR, Daimler-Benz Research, the Airforce Missile System Command, or the United States Government.

References

[AAH88] C. H. An, C. G. Atkeson, and J. M. Hollerbach. *Model-Based Control of a Robot Manipulator*. The MIT Press, Cambridge, MA, 1988.

[BEF96] J. Borenstein and L. Feng. Measurement and correction of systematic odometry errors in mobile robots. *IEEE Transactions on Robotics and Automation*, 12:869–880, December 1996.

[BFHS96a] W. Burgard, D. Fox, D. Hennig, and T. Schmidt. Estimating the absolute position of a mobile robot using position probability grids. In *Proc. 13th Nat. Conf. on Artificial Intelligence*, Menlo Park, Aug. 1996.

[BFHS96b] W. Burgard, D. Fox, D. Hennig, and T. Schmidt. Position tracking with position probability grids. In *Proc. 1st Euromicro Workshop on Advanced Mobile Robots*. IEEE Computer Society Press, 1996.

[BFL+97] W. Burgard, A.B. Cremers, D. Fox, G. Lakemeyer, D. Hähnel, D. Schulz, W. Steiner, and S. Thrun. The Interactive Museum Tour-Guide Robot. In *Proc. 15th Nat. Conf. on Artificial Intelligence*, Madison, WI. Jul. 1998.

[BFT97] W. Burgard, D. Fox, and S. Thrun. Active mobile robot localization. In *Proceedings of IJCAI-97*. IJCAI, Inc., 1997. 1997.

[BK91] J. Borenstein and Y. Koren. The vector field histogram – fast obstacle avoidance for mobile robots. *IEEE Journal of Robotics and Automation*, 7(3):278–288, June 1991.

[Bor94] J. Borenstein. Internal correction of dead-reconing errors with the smart encoder trailer. In *Proc. IEEE/RSJ/GI Int. Conf. on Intelligent Robots and Systems*, pages 127–134, September 1994.

[CW90] I.J. Cox and G.T. Wilfong, editors. *Autonomous Robot Vehicles*. Springer Verlag, 1990.

[Elf87] A. Elfes. Sonar-based real-world mapping and navigation. *IEEE Journal of Robotics and Automation*, RA-3(3):249–265, June 1987.

[FBT98] D. Fox, W. Burgard, and S. Thrun. A hybrid collision avoidance method for mobile robots. In *Proc. IEEE Int. Conf. on Robotics and Automation*, 1998. to appear.

[FvDFH90] J. Foley, A. van Dam, S. Feiner, and J. Hughes. *Interactive Computer Graphics: Principles and Practice*. Addison-Wesley, 1990.

[GN97] J.-S. Gutmann and B. Nebel. Navigation mobiler roboter mit laserscans. In *Autonome Mobile Systeme*. Springer Verlag, Berlin, 1997.

[KB91] B. Kuipers and Y.-T. Byun. A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. *Journal of Robotics and Autonomous Systems*, 8:47–63, 1991.

[KCK96] L.P. Kaelbling, A.R. Cassandra, and J.A. Kurien. Acting under uncertainty: Discrete bayesian models for mobile-robot navigation. In *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 1996.

[KS96] S. Koenig and R. Simmons. Passive distance learning for robot navigation. In L. Saitta, editor, *Proc. 13th Int. Conf. on Machine Learning*, 1996.

[LM97] F. Lu and E. Milios. Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, 4:333–349, 1997.

[MD94] Paul MacKenzie and Gregory Dudek. Precise positioning using model-based maps. In *Proc. IEEE Int. Conf. on Robotics and Automation*, San Diego, CA, 1994.

[Mor88] H. P. Moravec. Sensor fusion in certainty grids for mobile robots. *AI Magazine*, pages 61–74, Summer 1988.

[NPB95] I. Nourbakhsh, R. Powers, and S. Birchfield. DERVISH an office-navigating robot. *AI Magazine*, 16(2):53–60, Summer 1995.

[Ren93] W.D. Rencken. Concurrent localisation and map building for mobile robots using ultrasonic sensors. In *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 2129–2197, Yokohama, Japan, July 1993.

[SK95] R. Simmons and S. Koenig. Probabilistic robot navigation in partially observable environments. In *Proceedings of IJCAI-95*, pages 1080–1087, Montreal, Canada, August 1995. IJCAI, Inc.

[SK97] H Shatkay and L. Kaelbling. Learning topological maps with weak local odometric information. In *Proceedings of IJCAI-97*. IJCAI, Inc., 1997. 1997.

[TFB98] S. Thrun, D. Fox, and W. Burgard. A probabilistic approach to concurrent mapping and localization for mobile robots. *Machine Learning and Autonomous Robots (joint issue)*, 1998. to appear.

[Thr98a] S. Thrun. Bayesian landmark learning for mobile robot localization. *Machine Learning*, 1998. to appear.

[Thr98b] S. Thrun. Learning maps for indoor mobile robot navigation. *Artificial Intelligence*, 1998. to appear.

[Vuk89] M. Vukobratović. *Introduction to Robotics*. Springer Publisher, Berlin, New York, 1989.