Using the CONDENSATION Algorithm for Robust, Vision-based Mobile Robot Localization

Frank Dellaert† Wolfram Burgard‡ Dieter Fox† Sebastian Thrun†
†Computer Science Department, Carnegie Mellon University, Pittsburgh PA 15213
‡Institute of Computer Science III, University of Bonn, D-53117 Bonn

Abstract

To navigate reliably in indoor environments, a mobile robot must know where it is. This includes both the ability of globally localizing the robot from scratch, as well as tracking the robot's position once its location is known. Vision has long been advertised as providing a solution to these problems, but we still lack efficient solutions in unmodified environments. Many existing approaches require modification of the environment to function properly, and those that work within unmodified environments seldomly address the problem of global localization.

In this paper we present a novel, vision-based localization method based on the CONDENSATION *algorithm [17, 18], a Bayesian filtering method that uses a sampling-based density representation. We show how the* CONDENSATION *algorithm can be used in a novel way to track the position of the camera platform rather than tracking an object in the scene. In addition, it can also be used to globally localize the camera platform, given a visual map of the environment.*

Based on these two observations, we present a vision-based robot localization method that provides a solution to a difficult and open problem in the mobile robotics community. As evidence for the viability of our approach, we show both global localization and tracking results in the context of a state of the art robotics application.

1 Introduction

To operate autonomously, mobile robots must know where they are. *Mobile robot localization*, that is the process of determining and tracking the location of a mobile robot relative to its environment, has received considerable attention over the past few years. Accurate localization is a key prerequisite for successful navigation in large-scale environments, particularly when global models are used, such as maps, drawings, topological descriptions, and CAD models [22]. As demonstrated by a recent survey of localization methods by Borenstein, Everett, and Feng [2], the number of existing approaches is diverse. Cox [7] noted that "Using sensory information to locate the robot in its environment is the most fundamental problem to providing a mobile robot with autonomous capabilities."

The current state of the art in localization often uses laser range finders or sonar [2], but these sensor modalities have a number of problems. They tend to be easily confused in highly dynamic environments, e.g., when the laser or sonar beams are blocked by people [12]. Being limited in range, they cannot be expected to provide accurate localization in large open areas, where no landmarks can be observed [15]. Finally, these systems tend to be expensive.

Vision has long been advertised as providing a solution to these problems, but efficient solutions in unmodified environments are still lacking. Many of the approaches reviewed in [2] require artificial landmarks such as bar-code reflectors [10], reflecting tape, or visual patterns that are easy to recognize, such as black rectangles with white dots [1]. But even approaches that use more natural landmarks [26, 20, 23, 15] are used mainly to *track* the position of a robot, and are not able to globally localize the robot.

In this paper, we present a vision-based approach to localization based on the CONDENSATION algorithm, which provides both robustness and the capability to globally localize the robot. The idea is to use a visual map of the ceiling, obtained by mosaicing, and localize the robot using a simple scalar brightness measurement as the sensor input. Because the information coming from this sensor is highly ambiguous, a Kalman filter approach is not applicable. The CONDENSATION algorithm was introduced in the vision community precisely to deal with this type of measurement uncertainty [17, 18]. It belongs to the general class of Monte Carlo filters, invented in the seventies [16], and recently rediscovered independently in the target-tracking [13], statistical [21], and computer vision literature [17, 18]. These algorithms use a set of random samples or *particles* to represent the propagation of arbitrary probability densities over time. It is this property that makes them ideally suited to our application.

The resulting localization method, which we have called the Monte Carlo Localization method, draws on the following contributions:

1. We show how the CONDENSATION algorithm can

be used in a novel way to track the position of the camera platform rather than tracking an object in the scene.

2. We show how in addition to providing robust tracking, the CONDENSATION algorithm can be used to globally localize the robots, given a visual map of the environment.

3. Based on these two observations, we present a vision-based robot localization method that provides a solution to a difficult and open problem in the robotics community.

As evidence for the viability of our approach, we show both global localization and tracking results in the context of a state of the art robotics application. In this application, the mobile robot Minerva functioned as an interactive robotic tour-guide in the Smithsonian's National Museum of American History. During its two weeks of performance, it interacted with over 50,000 people, traversing more than 44 km. Our successful results within this unmodified and highly dynamic environment attest to the promise of our approach.

In the remainder of this paper we first discuss the general problem of robot localization, in Section 2. This is made more explicit in Section 3, where we discuss how a simple vision-based sensor can give rise to complex likelihood densities. In Section 4, we provide a taxonomy of algorithms based on the way they deal (or not) with these types of densities. In Section 5, we show how the CONDENSATION algorithm provides an elegant solution to the problem, and use it as the basis for a novel localization method. Finally, Section 6 shows global localization and tracking results in our museum testbed application.

2 Robot Localization

In robot localization, we are interested in estimating the state of the robot at the current time-step k, given knowledge about the initial state and all measurements $Z^k = \{\mathbf{z}_k, i = 1..k\}$ up to the current time. Typically, we work with a three-dimensional state vector $\mathbf{x} = [x, y, \theta]^T$, i.e. the position and orientation of the robot. This estimation problem is an instance of the Bayesian filtering problem, where we are interested in constructing the posterior density $p(\mathbf{x}_k|Z^k)$ of the current state conditioned on all measurements. In the Bayesian approach, this probability density function (PDF) is taken to represent all the knowledge we possess about the state $\mathbf{x}_k$, and from it we can estimate the current position. Often used estimators are the mode (the maximum a posteriori or MAP estimate) or the mean, when the density is unimodal. However, particularly during the global localization phase, this density will be multi-modal and calculating a single position estimate is not appropriate.

Summarizing, to localize the robot we need to recursively compute the density $p(\mathbf{x}_k|Z^k)$ at each time-step. This is done in two phases:

Prediction Phase In the first phase we use a *motion model* to predict the current position of the robot in the form of a predictive PDF $p(\mathbf{x}_k|Z^{k-1})$, taking only motion into account. We assume that the current state $\mathbf{x}_k$ is only dependent on the previous state $\mathbf{x}_{k-1}$ (Markov) and a known control input $\mathbf{u}_{k-1}$, and that the motion model is specified as a conditional density $p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{u}_{k-1})$. The predictive density over $\mathbf{x}_k$ is then obtained by integration:

$$p(\mathbf{x}_k|Z^{k-1}) = \int p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{u}_{k-1})\, p(\mathbf{x}_{k-1}|Z^{k-1})\, d\mathbf{x}_{k-1} \tag{1}$$

Update Phase In the second phase we use a *measurement model* to incorporate information from the sensors to obtain the posterior PDF $p(\mathbf{x}_k|Z^k)$. We assume that the measurement $\mathbf{z}_k$ is conditionally independent of earlier measurements Z^{k-1} given $\mathbf{x}_k$, and that the measurement model is given in terms of a likelihood $p(\mathbf{z}_k|\mathbf{x}_k)$. This term expresses the likelihood of the state $\mathbf{x}_k$ given that $\mathbf{z}_k$ was observed. The posterior density over $\mathbf{x}_k$ is obtained using Bayes' theorem:

$$p(\mathbf{x}_k|Z^k) = \frac{p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|Z^{k-1})}{p(\mathbf{z}_k|Z^{k-1})} \tag{2}$$

After the update phase, the process is repeated recursively. At time t_0 the knowledge about the initial state $\mathbf{x}_0$ is assumed to be available in the form of a density $p(\mathbf{x}_0)$. In the case of global localization, this density might be a uniform density over all allowable positions. In tracking work, the initial position is often given as the mean and covariance of a Gaussian centered around $\mathbf{x}_0$. In our work, as in [4], the transition from global localization to tracking is automatic and seamless, and the PDF evolves from spanning the whole state space to a well-localized peak.

3 Vision-Based Localization

In this section, we explain the basic idea behind our vision-based approach to localization, and show that it gives rise to complex and multi-modal likelihood densities $p(\mathbf{z}|\mathbf{x})$. This provides the motivation for using the CONDENSATION algorithm rather than a Kalman filter based approach.

The basic idea of our approach is to generate an image mosaic of the ceiling of the environment in which the robot needs to operate, and use that during operation to localize the robot. An example of such a ceiling mosaic from our testbed-application is shown in Fig. 1. The figure shows a large portion (40 m. deep by 60 m. wide) of the first floor ceiling of the Museum for American History (MAH), one of the Smithsonians in Washington, DC. The mosaic was generated from 250 images by globally aligning the

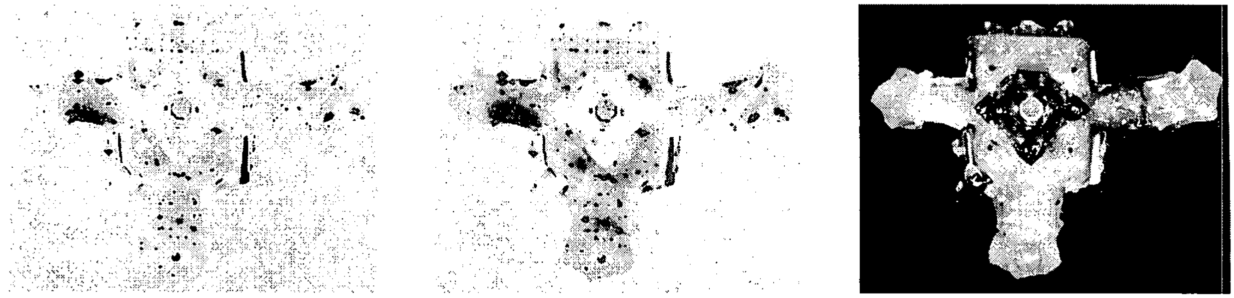

Figure 2: The likelihood densities $p(\mathbf{z}|\mathbf{x})$ of being at position $\mathbf{x}$ given a brightness measurement $\mathbf{z}$ for three values of $\mathbf{z}$. Left: high $\mathbf{z}$ indicates the robot is under a light. Middle: intermediate $\mathbf{z}$. Right: low $\mathbf{z}$: not under a light.

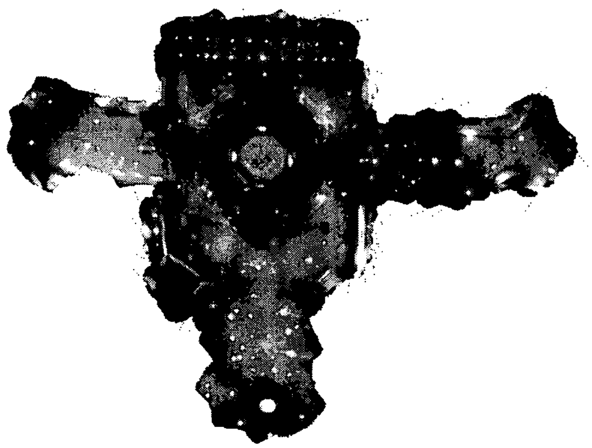

Figure 1: Example of a large-scale ceiling mosaic. The area shown is 40 m. deep by 60 m. wide.

images, and estimating the texture of the ceiling from the aligned images. The alignment was done using a variant of an algorithm by Lu and Milios [25] for map-building from laser range finder scans. The details are beyond the scope of the current paper, and are described elsewhere [8].

At run time, we use a simple scalar measurement to localize the robot. Rather than working with entire images, the idea is to use a very simple sensor that can be integrated at high frequencies while requiring very little computation and memory. This is an important property in enabling small, low cost robots. In particular, the brightness of a small ceiling patch directly above the robot is measured. In our testbed application, the measurement is done using a camera pointed at the ceiling, by extracting a small window of 25 by 25 pixels from the center of the image. A Gaussian filter with a standard deviation of 6 pixels is then applied, to create a single scalar measurement. The blurring is essential to filter out frequency components that cannot be represented within the mosaic, which is of finite resolution.

The likelihood densities $p(\mathbf{z}|\mathbf{x})$ associated with this simple sensor model are in general complex and multimodal, which explains the need for more powerful probabilistic reasoning. To illustrate this, three different likelihood densities are shown in Fig. 2, respectively when the robot is under a light, at the border of a light, and not under a light. It is clear that these densities do not resemble anything remotely like a Gaussian density, and that more powerful density representations are required than provided by the Kalman filter.

4 Existing Approaches: A Tale of Density Representations

The solution to the robot localization problem is obtained by recursively solving the two (1) and (2). Depending on how one chooses to represent the density $p(\mathbf{x}_k|Z^k)$, one obtains algorithms with vastly different properties:

The Kalman filter If both the motion and the measurement model can be described using a Gaussian density, and the initial state is also specified as a Gaussian, then the density $p(\mathbf{x}_k|Z^k)$ will remain Gaussian at all times. In this case, (1) and (2) can be evaluated in closed form, yielding the classical Kalman filter [27]. Kalman-filter based techniques [24, 31, 14] have proven to be robust and accurate for keeping track of the robot's position. Because of its concise representation (the mean and covariance matrix suffice to describe the entire density) it is also a particularly efficient algorithm. However, it is clear that the basic assumption of Gaussian densities is violated in our application, where the likelihood densities are typically complex and multi-modal, as discussed above.

Topological Markov Localization To overcome these disadvantages, different approaches have used increasingly richer schemes to represent uncertainty, moving away from the restricted Gaussian density assumption inherent in the Kalman filter. These different methods can be roughly distinguished by the type of discretization used for the representation of the state space. In [28, 32, 19, 34], Markov localization is used for landmark-based corridor navigation and the state space is organized according to the topological structure of the environment. However, in many applications one is interested in a more fine-grained position estimate, e.g., in environments with a simple topology but large open spaces, where accurate placement of the robot is needed.

Grid-based Markov Localization To deal with multimodal and non-Gaussian densities at a fine resolution, one can discretize the interesting part of the state space, and use it as the basis for an approximation of the density $p(\mathbf{x}_k|Z^k)$, e.g. by a piece-wise constant function [3]. This idea forms the basis of the grid-based Markov localization approach [5, 11]. Methods that use this type of representation are powerful, but suffer from the disadvantages of computational overhead and *a priori* commitment to the size of the state space. In addition, the resolution and thereby also the precision at which they can represent the state has to be fixed beforehand. The computational requirements have an effect on accuracy as well, as not all measurements can be processed in real-time, and valuable information about the state is thereby discarded.

Sampling-based Methods Finally, one can represent the density by a set of samples that are randomly drawn from it. This is the representation we will use, and it forms the topic of the next section.

5 Monte Carlo Localization

In this section, we present a novel localization method based on a sampling-based representation, using the CONDENSATION algorithm [17, 18] to propagate the posterior density over time. In the localization context, CONDENSATION is used to estimate the position of the camera platform rather than an object in the scene. In addition, the ability to represent arbitrary densities enables us to perform global localization without modification.

In sampling-based methods one represents the density $p(\mathbf{x}_k|Z^k)$ by a set of N random samples or *particles* $S_k = \{s_k^i; i = 1..N\}$ drawn from it. We are able to do this because of the essential duality between the samples and the density from which they are generated [33]. From the samples we can always approximately reconstruct the density, e.g. using a histogram or a kernel based density estimation technique.

The goal is then to recursively compute at each time-step k the set of samples S_k that is drawn from $p(\mathbf{x}_k|Z^k)$. A particularly elegant algorithm to accomplish this is the CONDENSATION algorithm [17, 18], known alternatively as the bootstrap filter [13] or the Monte-Carlo filter [21]. These methods are generically known as *particle filters*, and an overview and discussion of their properties can be found in [9].

In analogy with the formal filtering problem outlined in Section 2, the algorithm proceeds in two phases:

Prediction Phase In the first phase we start from the set of particles S_{k-1} computed in the previous iteration, and apply the motion model to each particle s_{k-1}^i by sampling from the density $p(\mathbf{x}_k|s_{k-1}^i, \mathbf{u}_{k-1})$:

(i) for each particle s_{k-1}^i:
  draw one sample s'^i_k from $p(\mathbf{x}_k|s_{k-1}^i, \mathbf{u}_{k-1})$

In doing so a new set S'_k is obtained that approximates a random sample from the predictive density $p(\mathbf{x}_k|Z^{k-1})$. The prime in S'_k indicates that we have not yet incorporated any sensor measurement at time k.

Update Phase In the second phase we take into account the measurement $\mathbf{z}_k$, and weight each of the samples in S'_k by the weight $m_k^i = p(\mathbf{z}_k|s'^i_k)$, i.e. the likelihood of s'^i_k given $\mathbf{z}_k$. We then obtain S_k by resampling from this *weighted* set:

(ii) for j=1..N:
  draw one S_k sample s_k^j from $\{s'^i_k, m_k^i\}$

The resampling selects with higher probability samples s'^i_k that have a high likelihood associated with them, and in doing so a new set S_k is obtained that approximates a random sample from $p(\mathbf{x}_k|Z^k)$. An algorithm to perform this resampling process efficiently in O(N) time is given in [6].

After the update phase, the steps (i) and (ii) are repeated recursively. To initialize the filter, we start at time $k = 0$ with a random sample $S_0 = \{s_0^i\}$ from the prior $p(\mathbf{x}_0)$.

A Graphical Example

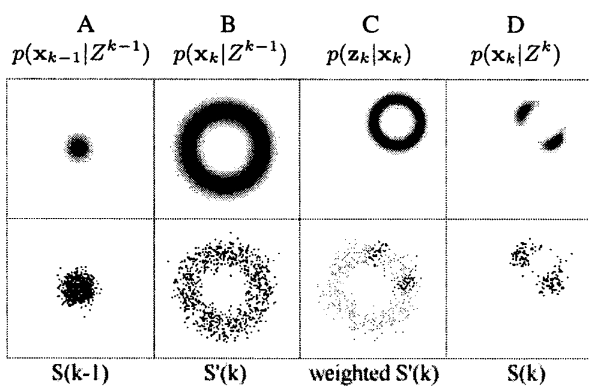

Fig. 3: The probability densities and particle sets for one iteration of the algorithm. See text for detail.

One iteration of the algorithm is illustrated in Figure 3. In the figure each panel in the top row shows the exact density, whereas the panel below shows the particle-based representation of that density. In panel A, we start out with a

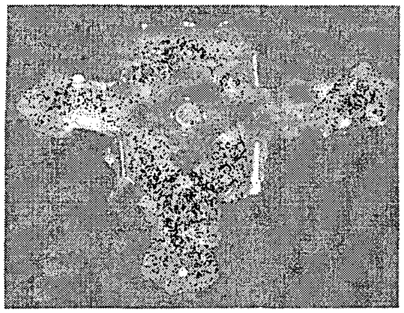

Fig. 5: Initial uncertainty at iteration 15.

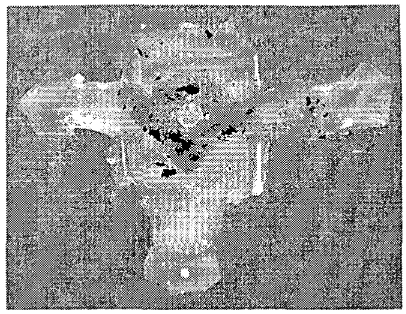

Fig. 6: Representing ambiguity at iteration 38.

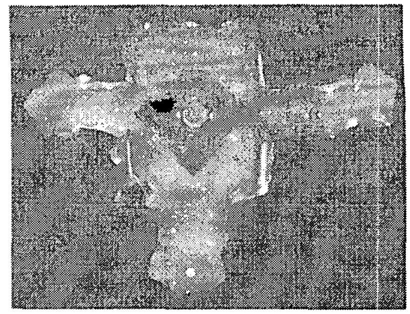

Fig. 7: Global localization at iteration 126.

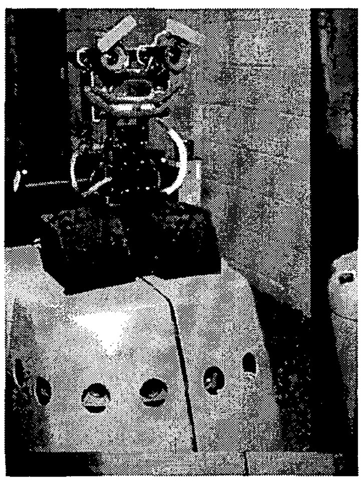

Figure 4: Our testbed, the robotic museum tour-guide Minerva. The ceiling camera can be seen on the right.

cloud of particles S_{k-1} representing our uncertainty about the robot position. In the example, the robot is fairly localized, but its orientation is unknown. Panel B shows what happens to our belief state when we are told the robot has moved exactly one meter since the last time-step: we now know the robot to be somewhere on a circle of 1 meter radius around the previous location. Panel C shows what happens when we observe a landmark, half a meter away, somewhere in the top-right corner: the top panel shows the likelihood $p(\mathbf{z}_k|\mathbf{x}_k)$, and the bottom panel illustrates how each sample s'^i_k is weighted according to this likelihood. Finally, panel D shows the effect of resampling from this weighted set, and this forms the starting point for the next iteration.

Good explanations of the mechanism underlying the CONDENSATION algorithm are given in [6, 29]. The entire procedure of sampling, reweighting and subsequently resampling to sample from the posterior is called *Sampling/Importance Resampling* (SIR) [30], and an accessible introduction to it can be found in [33].

6 Experimental Results

As a test-bed for our approach we used the robotic tour-guide Minerva, a prototype RWI B18 robot shown in Fig. 4. In the summer of 1998, Minerva functioned for two weeks as an interactive robotic tour-guide in the Smithsonian's National Museum of American History. During this time, it interacted with thousands of people, traversing more than 44 km. The datasets we work with consist of logs of odometry and vision-based measurements collected while Minerva operated within the MAH environment. The upward pointing camera, used both for building the ceiling-mosaic as performing the brightness measurement, can be seen on the right-hand side of Fig. 4. Using the time stamps in the logs, all tests have been conducted in real-time.

6.1 Global Localization

One of the key advantages of using a sampling-based representation over Kalman-filter based approaches is its ability to represent multi-modal probability distributions. This ability is a precondition for localizing a mobile robot from scratch, i.e., without knowledge of its starting location. This global localization capability is illustrated in Fig. 5 through Fig. 7. In the first iteration, the algorithm is initialized by drawing 40,000 samples from a uniform probability density save where there are known to be (static) obstacles. After 15 iterations of the algorithm, the samples are still scattered over the area, but tend to be concentrated in areas with high brightness. This is shown in Fig. 5. After 38 iterations, all but a few possibilities have been eliminated, as shown in Fig. 6. This is possible because all samples that disagree with the actual temporal sequence of brightness measurements become very unlikely. It should be noted that in this early stage of localization, the ability

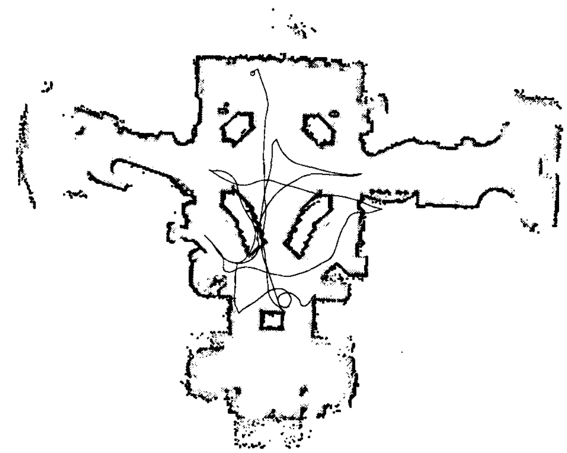

Fig. 8: Estimated path
using odometry only

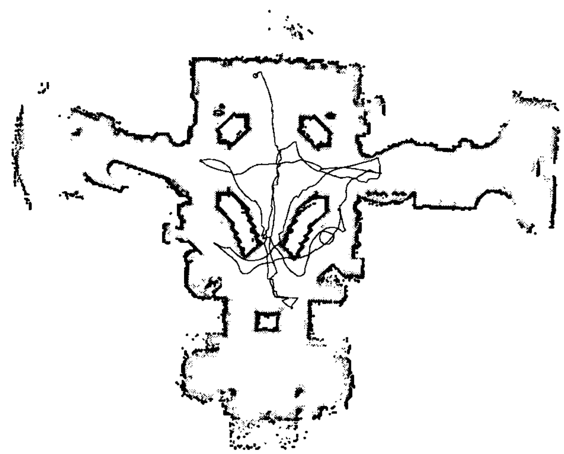

Fig. 9: Corrected path
using vision

to represent ambiguous probability distributions is vital for successful position estimation. Finally, after 126 iterations only one localized peak is left, as shown in Fig. 7, which indicates that the robot has been able to uniquely determine its position. Note that the samples are shown after the re-sampling step, where they are unweighted, and that most of the off-peak samples have in fact low probability.

6.2 Position Tracking

Using the same approach, we are also able to track the position of a robot over time, even when the robot is moving at high speeds. In this experiment, we used recorded data from Minerva, as it was moving with speeds up to 1.6 m/sec through the museum. At the time of this run there were no visitors in the museum, and the robot was remotely controlled by users connected through the world wide web.

To illustrate this, Fig. 8 shows an occupancy grid map of the museum along with the estimated trajectory *using odometry alone.* In the tracking experiment, the robot position is initialized with the true position at the top of the map. Due to the accumulated odometry error, the robot is hopelessly lost after traversing 200 meters. In contrast, the path estimated using the vision-sensor keeps track of the global position at all times. This can be appreciated from Fig. 9, where the vision-corrected path is shown. This path was generated by running the CONDENSATION algorithm with 1000 samples, and plotting the mean of the samples over time. The resulting estimated path is noisier than the one obtained by odometry, but it ends up within 10 cm. of the correct end-position, indicated with a black dot at the right of the figure.

In general, far fewer samples are needed for position tracking than for global localization, and an issue for future research is to adapt the number of samples appropriately. Also, the noisy nature of the estimated position occurs often when two or more probability peaks arise. In this case, rather than plotting one mean position, it might be more appropriate to display several alternative paths, corresponding to the modes of the posterior probability density.

7 Conclusion and Future Work

In this paper, we have shown that CONDENSATION type algorithms can be used in a novel way to perform global and local localization for mobile robots. The ability of these Monte Carlo methods to deal with arbitrary likelihood densities is a crucial property in order to deal with highly ambiguous sensors, such as the vision-based sensor used here. In addition, the ability to represent multimodal posterior densities allows them to globally localize the robot, condensing an initially uniform prior distribution into one globally localized peak over time.

Based on these properties, the resulting Monte Carlo Localization approach has been shown to address one of the fundamental and open problems in the mobile robot community. In order to support this claim, we have shown results within the context of a challenging robot application. Our method was able to both globally localize and locally track the robot within a highly dynamic and unmodified environment, and this with the robot traveling at high speeds.

The results in this paper were obtained using a single brightness measurement of the ceiling directly above the robot's position. In part, this is done to investigate how little sensing one can get away with. In future work, this can be further developed in two opposite directions: one can envision building robots with even less sensory input, e.g., by omitting wheel encoders that provide odometry measurements. This shifts the burden towards the development of even more powerful paradigms for probabilistic reasoning. In the other direction, it is clear that integrating more complex measurements will provide more information at each iteration of the algorithm, resulting in faster global localization and more accurate tracking than shown here. In future work, we hope to investigate both these possibilities in more depth.

References

[1] J. Borenstein. *The nursing robot system*. PhD thesis, Technion, Haifa, Israel, 1987.

[2] J. Borenstein, B. Everett, and L. Feng. *Navigating Mobile Robots: Systems and Techniques*. A. K. Peters, Ltd., Wellesley, MA, 1996.

[3] R. S. Bucy. Bayes theorem and digital realisation for nonlinear filters. *Journal of Astronautical Science*, 17:80–94, 1969.

[4] W. Burgard, A. Derr, D. Fox, and A.B. Cremers. Integrating global position estimation and position tracking for mobile robots: the Dynamic Markov Localization approach. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 1998.

[5] W. Burgard, D. Fox, D. Hennig, and T. Schmidt. Estimating the absolute position of a mobile robot using position probability grids. In *Proc. of the Fourteenth National Conference on Artificial Intelligence (AAAI-96)*, pages 896–901, 1996.

[6] J. Carpenter, P. Clifford, and P. Fernhead. An improved particle filter for non-linear problems. Technical report, Department of Statistics, University of Oxford, 1997.

[7] I.J. Cox. Blanche—an experiment in guidance and navigation of an autonomous robot vehicle. *IEEE Transactions on Robotics and Automation*, 7(2):193–204, 1991.

[8] Frank Dellaert, Sebastian Thrun, and Chuck Thorpe. Mosaicing a large number of widely dispersed, noisy, and distorted images: A Bayesian approach. Technical report, Computer Science Department, Carnegie Mellon University, 1999.

[9] A. Doucet. On sequential simulation-based methods for Bayesian filtering. Technical Report CUED/F-INFENG/TR.310, Department of Engineering, University of Cambridge, 1998.

[10] H. Everett, D. Gage, G. Gilbreth, R. Laird, and R. Smurlo. Real-world issues in warehouse navigation. In *Proceedings of the SPIE Conference on Mobile Robots IX*, volume 2352, 1994.

[11] D. Fox, W. Burgard, and S. Thrun. Active Markov localization for mobile robots. *Robotics and Autonomous Systems*, 25(3-4):195–207, 1998.

[12] D. Fox, W. Burgard, S. Thrun, and A.B. Cremers. Position estimation for mobile robots in dynamic environments. In *Proceedings of the AAAI Fifteenth National Conference on Artificial Intelligence*, 1998.

[13] N.J. Gordon, D.J. Salmond, and A.F.M. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Procedings F*, 140(2):107–113, 1993.

[14] J.-S. Gutmann and C. Schlegel. Amos: Comparison of scan matching approaches for self-localization in indoor environments. In *Proceedings of the 1st Euromicro Workshop on Advanced Mobile Robots*. IEEE Computer Society Press, 1996.

[15] G. Hager, D. Kriegman, E. Yeh, and C. Rasmussen. Image-based prediction of landmark features for mobile robot navigation. In *IEEE Conf. on Robotics and Automation (ICRA)*, pages 1040–1046, 1997.

[16] J.E. Handschin. Monte Carlo techniques for prediction and filtering of non-linear stochastic processes. *Automatica*, 6:555–563, 1970.

[17] M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *Eur. Conf. on Computer Vision (ECCV)*, pages 343–356, 1996.

[18] M. Isard and A. Blake. Condensation – conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1):5–28, 1998.

[19] L.P. Kaelbling, A.R. Cassandra, and J.A. Kurien. Acting under uncertainty: Discrete Bayesian models for mobile-robot navigation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1996.

[20] S. King and C. Weiman. Helpmate autonomous mobile robot navigation system. In *Proceedings of the SPIE Conference on Mobile Robots*, volume 2352, 1990.

[21] G. Kitagawa. Monte carlo filter and smoother for non-gaussian nonlinear state space models. *Journal of Computational and Graphical Statistics*, 5(1):1–25, 1996.

[22] D. Kortenkamp, R.P. Bonasso, and R. Murphy, editors. *AI-based Mobile Robots: Case studies of successful robot systems*, Cambridge, MA, 1998. MIT Press.

[23] D. Kortenkamp and T. Weymouth. Topological mapping for mobile robots using a combination of sonar and vision sensing. In *Proceedings of the Twelfth National Conference on Artificial Intelligence*, pages 979–984, 1994.

[24] J.J. Leonard and H.F. Durrant-Whyte. *Directed Sonar Sensing for Mobile Robot Navigation*. Kluwer Academic, Boston, 1992.

[25] F. Lu and E Milios. Globally consistent range scan alignment for environment mapping. Technical report, Department of Computer Science, York University, April 1997.

[26] M. J. Matarić. A distributed model for mobile robot environment-learning and navigation. Master's thesis, MIT, Artificial Intelligence Laboratory, Cambridge, 1990.

[27] P. Maybeck. *Stochastic Models, Estimation and Control*, volume 1. Academic Press, New York, 1979.

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

[29] M. K Pitt and N. Shephard. Filtering via simulation: auxiliary particle filters. Technical report, Department of Mathematics, Imperial College, London, October 1997.

[30] D.B. Rubin. Using the SIR algorithm to simulate posterior distributions. In *Bayesian Statistics*, volume 3, pages 395–402. Oxford University Press, 1988.

[31] B. Schiele and J. L. Crowley. A comparison of position estimation techniques using occupancy grids. In *IEEE Conf. on Robotics and Automation (ICRA)*, volume 2, pages 1628–1634, 1994.

[32] R. Simmons and S. Koenig. Probabilistic robot navigation in partially observable environments. In *Proc. International Joint Conference on Artificial Intelligence*, 1995.

[33] A.F.M. Smith and A.E. Gelfand. Bayesian statistics without tears: A sampling-resampling perspective. *American Statistician*, 46(2):84–88, 1992.

[34] S. Thrun. Bayesian landmark learning for mobile robot localization. *Machine Learning*, 33(1), 1998.