REVIEW

entro 14/01/08

Robotics and Autonomous Systems

Manuscript Draft

Manuscript Number: ROBOT-D-07-00069

Title: A probabilistic approach to collaborative distributed sensors for mobile robot localization

Article Type: Full Length Article

Section/Category:

Keywords: Collaborative distributed sensors localization; Environmental camera; Joint detection model

Manuscript Region of Origin:

Abstract: This paper presents a probabilistic algorithm ~~to~~ for collaborative distributed sensors for mobile robot localization. Our approach uses a sample-based version of Markov localization, capable of localizing mobile robot in an any-time fashion. During robot localization given a known environment, probabilistic method is employed to synchronize robot's belief whenever one environmental sensor detects the robot. We present an implementation that uses color ~~environmental~~ cameras installed in the environment for robot detection. All the parameters of each environmental camera are unknown in advance and can be calibrated independently by the robot online. Once the cameras are calibrated, the joint detection models are trained using sensor data according to cameras' parameters. As a result, the ~~robot's~~ belief can reduce its uncertainty in response to the detection effectively. ~~A further~~ Experiments obtained with the real robot in an indoor office environment, illustrates that drastic improvements in localization speed and accuracy when compared to conventional robot localization only based its own sensors.

? what do you mean?

about the position of the robot

what do you mean by "independently"?

environments. However, if robot can be detected by the environmental sensors, there is the opportunity to do better. When an environmental sensor determines the location of robot, robot can refine its internal belief based on the sensor's estimate, hence improves its localization accuracy. The ability to exchange information between robot and environmental sensors during localization is particularly attractive in the context of global localization, where each sight of robot can reduce the uncertainty in the estimated location dramatically.

This paper proposes an efficient probabilistic approach to collaborative distributed sensors for mobile robot localization. Our approach is based on *Markov localization* [4-7], a family of probabilistic approaches that have been applied with great practical success to robot localization independently [5,9,19]. In contrast to previous research, which relied on grid-based or coarse-grained topological representations of a robot's state space, our approach adopts a sampling-based representation [11], which is capable of approximating a wide range of belief functions in real-time. To transfer information across different platforms, probabilistic joint detection models are employed to model the abilities of environmental sensors to recognize robot. When one environmental sensor detects the robot, the joint detection model is used to synchronize the robot's belief, thereby reducing the uncertainty of robot during localization. To accommodate the noise and ambiguity arising in real-world domains, joint detection models are probabilistic, capturing the reliability and accuracy of robot detection.

please explain what you mean by "joint detection model"

While our approach is applicable to any sensor capable of (occasionally) detecting robot, we present an implementation that uses color ~~environmental~~ cameras for robot detection. The location and parameters of all environmental cameras is unknown and need to be calibrated by robot online. Once obtained ~~getting~~ the cameras' parameters, the global localization of the detected robot is attained according to the joint probabilistic detection model. The parameters of the corresponding joint probabilistic detection model are learned using a maximum likelihood estimator. Experimental results, carried out with two ~~environmental~~ cameras in an indoor environment, illustrate the appropriateness of the approach.

just "two" cameras are not enough convincing please add more cameras.

In what follows, we will first describe the necessary statistical mechanisms for distributed sensors localization, followed by a description of our sampling-based and Monte Carlo localization technique in Section 3. In Section 3 we also present our method based on environmental cameras to detect mobile robot. Experimental results are provided in Section 4. Finally, we finish this paper with a discussion of the advantages and limitations of the current approach.

2. Theory of Distributed Sensors Localization

Let us begin with a mathematical derivation of our approach to robot localization combining with distributed environmental sensors. In the remainder we assume that the robot is given a model of the environment (e.g., a map [18]), and that it is given sensors that enable it to relate its own position to this model (e.g., range finders). All of these senses are typically confounded by noise. Throughout this paper, we adopt a probabilistic approach to localization. Probabilistic methods have been applied with remarkable success to robot localization only using its onboard sensors [5,12], where they have been demonstrated to solve problems like global localization and localization in dense crowds.

NOT TRUE

2.1. Data

Let M be the number of environmental sensors, and let d denote the data gathered by the robot. Obviously, d is a sequence of three different types of information:

(1) **Odometry measurements.** The robot continuously monitors its wheel encoders (dead-reckoning) and generates, in regular intervals, odometric measurements. These measurements, which will be denoted by a, specify the relative change of position according to the wheel encoders.

(2) **Environment measurements.** The robot also queries its sensors (e.g., range finders) in regular time intervals,

$Bel_t^{(r)}\left(L^{(r)}=l\right)$ of being at location l after incorporating o_t is obtained by multiplying the perceptual model $P\left(o_t \middle| L_t^{(r)}=l\right)$ with the prior brief $Bel_{t-1}^{(r)}\left(L_t^{(r)}=l\right)$.

This observation suggests the following incremental update equation (we omit the time index t and the state variable L for brevity):

$$Bel^{(r)}(l) \leftarrow \alpha P\left(o \middle| l\right) Bel^{(r)}(l) \tag{3}$$

The conditional probability $P\left(o \middle| l\right)$ is called the environment perception model of robot and describes the likelihood of perceiving o given that the robot is at position l. In Markov localization, it is assumed to be given and constant over time. For proximity sensors such as ultrasound sensors or laser range-finders, the probability $P\left(o \middle| l\right)$ can be approximated by $P\left(o \middle| o_l\right)$, which is the probability of observing o conditioned on the expected measurement o_l at location l. The expected measurement, a distance in this case, is easily computed from the map using ray tracing. Figure 1 shows this perception model for laser range-finders. Here the x-axis is the distance o_l expected given the world model, and the y-axis is the distance measured o by the sensor. The function is a mixture of a Gaussian (centered around the correct distance o_l), a Geometric distribution (modeling overly short readings) and a Dirac distribution (modeling max-range readings) [22]. It integrates the accuracy of the sensor with the likelihood of receiving a "random" measurement (e.g., due to obstacles not modeled in the map [9]).

(2) **Odometry**: Now suppose the last item in d_t is an odometry measurement, denoted a_t. Using the Theorem of Total Probability and exploiting the Markov property, we obtain

$$\begin{aligned} Bel_t^{(r)}\left(L^{(r)}=l\right) &= P\left(L_t^{(r)}=l \middle| d_t\right) \\ &= \int P\left(L_t^{(r)}=l \middle| d_t, L_{t-1}^{(r)}=l'\right) P\left(L_{t-1}^{(r)}=l' \middle| d_t\right) dl' \\ &= \int P\left(L_t^{(r)}=l \middle| a_t, L_{t-1}^{(r)}=l'\right) P\left(L_{t-1}^{(r)}=l' \middle| d_{t-1}\right) dl' \\ &= \int P\left(L_t^{(r)}=l \middle| a_t, L_{t-1}^{(r)}=l'\right) Bel_{t-1}^{(r)}\left(L^{(r)}=l'\right) dl' \end{aligned} \tag{4}$$

Which suggests the incremental update equation:

$$Bel^{(r)}(l) \leftarrow \int P\left(l \middle| a, l'\right) Bel^{(r)}(l')\, dl' \tag{5}$$

Here $P\left(l \middle| a, l'\right)$ is called the motion model of robot.

These equations together form the basis of Markov localization, an incremental probabilistic algorithm for estimating robot positions. Markov localization relies on knowledge of $P\left(o \middle| l\right)$ and $P\left(l \middle| a, l'\right)$. The former conditional typically requires a model (map) of the environment. As noticed above, Markov localization has been applied with great practical success to mobile robot localization. However, it is only applicable to robot localization based on its own sensors, and cannot take advantage of environmental sensors' detection measurements.

2.3. Distributed Sensors Markov Localization

The key idea of distributed sensors localization is to integrate measurements taken at different platforms, so that robot can benefit from data gathered from itself as well as environmental sensors.

To derive how to integrate detections into the robot's belief, let us assume that robot is detected by environmental sensor m and a detection variable is denoted $r_t^{(m)}$. It provides information about the location of the the robot relative

(2) **Environment measurements** are incorporated by re-weighting the sample set, which is analogous to Bayes rule in Markov localization [8]. More specifically, let $\langle l, p \rangle$ be a sample. Then

$$p \leftarrow \alpha P(o|l)$$

Where o is a sensor measurement, and is a normalization constant that enforces $\sum_{i=1}^{K} p_i = 1$. The incorporation of sensor readings is typically performed in two phases, one in which p is multiplied by $P(o|l)$, and one in which the various p-values are normalized. An algorithm to perform this re-sampling process efficiently in $O(K)$ time ~~is given in~~ [16]. This is a standard tecnique! You should just say you use it not: "we have found"

In practice, we have found it useful to add a small number of uniformly distributed, random samples after each estimation step [11]. Formally, these samples can be understood as a modified motion model that allows, with very small likelihood, arbitrary jumps in the environment. The random samples are needed to overcome local minima: Since MCL uses finite sample sets, it may happen that no sample is generated close to the correct robot position. This may be the case when the robot loses track of its position. In such cases, MCL would be unable to re-localize the robot. By adding a small number of random samples, however, MCL can effectively re-localize the robot, as documented described in [11].

Another modification to the basic approach is based on the observation that the best sample set sizes can vary drastically [17]. During global localization, a robot may be completely ignorant as to where it is; hence, it's belief uniformly covers its full three-dimensional state space. During position tracking, on the other hand, the uncertainty is typically small. MCL determines the sample set size on-the-fly: It typically uses many samples during global localization or if the position of the robot is lost, and only a small number of samples is used during position tracking (see [11] for details).

where is this defined? in Eq. 6? Please, state more clearly.

3.2. Distributed Sensors MCL

When one environmental sensor detects the robot, both sample sets are synchronized using the detection model, according to the update equation

$$Bel^{(r)}(l) \leftarrow Bel^{(r)}(l) \int P\left(L^{(r)} = l \middle| L^{(m)} = l', r^{(m)}\right) Bel^{(m)}(l')\,dl' \quad (8)$$

Notice that this equation requires the multiplication of two densities. The crucial component is the probabilistic detection mode $P\left(L^{(r)} = l \middle| L^{(m)} = l', r^{(m)}\right)$ which describes the conditional probability that robot is at location l, given that sensor m is at possible location l' and perceives robot with measurement $r^{(m)}$. From a mathematical point of view, our approach is sufficiently general to accommodate a wide range of sensors for robot detection, assuming that the conditional density $P\left(L^{(r)} \middle| L^{(m)}, r^{(m)}\right)$ is adjusted accordingly. However, for ~~environmental~~ camera it is not necessary to build the probabilistic detection mode $P\left(L^{(r)} = l \middle| L^{(m)} = l', r^{(m)}\right)$ and the environmental camera localization model $Bel^{(m)}(l')$ respectively. As a substitute, joint detection model $\int P\left(L^{(r)} = l \middle| L^{(m)} = l', r^{(m)}\right) Bel^{(m)}(l')\,dl'$ is constructed directly according to the ~~environmental~~ cameras' parameters. Before acquiring the global location of the detected robot, the cameras parameters need to be calibrated by robot online. ~~We will now describe a specific detection method that integrates information from environmental cameras below.~~ In the following paragraph we will describe the online calibration process.

How do you perform this calibration?

3.2.1. Automated camera calibration

In our method, all parameters of the environmental cameras are unknown in advance and their visual fields are not overlaid each other. So, in order to apply them to localize the robot, each cameras' parameters need to be calibrated

Please, explain better!

location of robot is known. Whenever AN environmental camera takes AN image which is analyzed as to whether robot is in its visual field, it is to exploit the fact that the locations of robot are known during training. Then, the image is analyzed, and for detected robot global location is computed according to the calibrated parameters of the environmental camera above. This data is sufficient to train the joint detection model $\int P\left(L^{(r)}=l \middle| L^{(m)}=l', r^{(m)}\right) Bel^{(m)}(l')\, dl'$.

The Gaussian distribution shown in Figure 4 models the error in the estimation of robot's location. Here the x-axis represents the error of x direction in the world coordinates, and the y-axis the y direction error. This Gaussian has been obtained through maximum likelihood estimation [21] based on the training data. As is easy to be seen, the Gaussian is zero-centered along both dimensions, and it assigns low likelihood to large errors. The correlation between both components of the error are approximately zero, suggesting that both errors might be independent. Assuming independence between the two errors, we found both the mean error of the estimation to be 15cm. Additionally, because the environmental camera can not detect the robot's orientation, the distribution of robot's heading angle employs uniform distribution in a range $[-\pi\ \pi]$.

To obtain the training data, the "true" location was *not* determined manually; instead, MCL was applied for position estimation (with a known starting position and very large sample sets). Empirical results in [17] suggest that MCL is sufficiently accurate for tracking a robot with only a few centimeters error. * The robots' positions, while moving at speeds like 30 cm/sec through our environment, were synchronized and then further analyzed geometrically to determine whether (and where) robots are in the visual fields of environmental cameras. As a result, data collection is extremely easy as it does not require any manual labeling; however, the error in MCL leads to a slightly less confined joint detection model than one would obtain with manually labeled data (assuming that the accuracy of manual position estimation exceeds that of MCL).

4. Experimental Results

In this section we present experiments conducted with real robot. The mobile robot used is Pioneer3 DX, which is equipped with a laser sensor. The central question driving our experiments was: *To what extent can cooperative distributed sensors localization improve the localization quality, when compared to conventional robot self-localization.*

Figure 5(a) shows the setup of our experiments along with a part of the occupancy grid map used for position estimation and two cameras are placed on the wall applied to detect and localize the robot. Figure 5(a) also shows the path from point A to C taken by Pioneer3 DX with laser sensor, which was in the process of global localization. Figure 5(b) represents the uncertain belief of the robot at point A from scratch. Before it passes point B(shown in Figure 5(c)), the robot is still highly uncertain about its exact location only depending on its onboard laser sensor. The key event, illustrating the utility of cooperation in localization, is a detection event. More specifically, the environmental camera 1 detects the robot as it moves through its visual field (see Figure 6). Using the joint detection model described in Section 3 , the robot integrates it into its current belief. The effect of this integration on robot's belief is shown in Figure 5(d). As this figure illustrates, this single incident almost completely resolves the uncertainty in robot's belief and shortens the time of robot global localization effectively.

We conducted ten experiments of this kind and compared the performance to conventional MCL for robot which ignores environmental cameras' detections. To measure the performance to localization we determined the true locations of the robot by measuring the starting position of each run and performing position tracking off-line using MCL. For each run, we then computed the estimation error at the reference positions. The estimation error is measured

* I do not agree. The "few centimeter error" is a statistical results.

? why the particles are only along the edges of the environment ?

[8] H. Zhou, S. Sakane, Sensor planning for mobile robot localization using Bayesian network representation and inference, in: Proc. of IEEE/RSJ Int.Conf. on Intelligent Robots and Systems, 2002, pp. 440–446.

[9] D. Fox, W. Burgard, and S. Thrun. Markov localization for mobile robots in dynamic environments. Journal of Artificial Intelligence Research, 1999, 11(1), pp.391-727.

[10] S. Thrun, M. Bennewitz,W. Burgard, A.B. Cremers, F. Dellaert, D. Fox, D. Hahnel, C. Rosenberg, J. Schulte, and D. Schulz. MINERVA: A second-generation museum tour-guide robot. International Journal of Robotics Research, 2000, 19(11), pp.972-999.

[11] D. Fox, W. Burgard, F. Dellaert, and S. Thrun, Monte Carlo localization: Efficient position estimation for mobile robots, In Proc. of the National Conference on Artificial Intelligence (AAAI), Orlando, Florida, 1999, pp.343-349.

[12] 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 Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 1998.

[13] Tsai, Roger Y, An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision, proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Miami Beach, FL, 1986, pp. 364–374.

[14] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, Monte Carlo localization for mobile robots, In Proc. of the IEEE International Conference on Robotics & Automation (ICRA), 1999, pp.1322-1328.

[15] D.B. Rubin, Using the SIR algorithm to simulate posterior distributions, In M.H. Bernardo, K.M. an DeGroot, D.V. Lindley, and A.F.M. Smith, editors, Bayesian Statistics 3. Oxford University Press, Oxford, UK, 1988.

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

[17] D. Koller and R. Fratkina, Using learning for approximation in stochastic processes, In Proc. of the International Conference on Machine Learning (ICML), 1998, pp.287-295.

[18] S. Thrun. Learning metric-topological maps for indoor mobile robot navigation, Artificial Intelligence, 1998, 99(1), pp.27-71.

[19] Fox, D.; Hightower, J.; Liao, L.; Schulz, D.; and Borriello, G., Bayesian filtering for location estimation, IEEE Pervasive Computing Magazine, 2003, 2(3), pp.24-33.

[20] Huiying Chen, Kohsei Matsumoto, Jun Ota, Tamio Arai, Self-calibration of environmental camera for mobile robot navigation, Robotics and Autonomous Systems 55 (2007), pp.177–190.

[21] A. Howard, MJ. Mataric, Localization for mobile robot teams using maximum likelihood estimation, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, EPFL Switzerland, 2002, pp.434-459.

[22] D. Fox, W. Burgard, H. Kruppa, S. Thrun, Efficient multi-robot localization based on Monte Carlo approximation, in: Proceedings of the. Ninth International Symposium of Robotics Research, Snowbird, UT, 1999, pp. 113–120.

Table 1

for each location l **do** /*initialize the belief*/

$$Bel^{(r)}(l) \leftarrow P\left(L_n^{(0)} = l\right)$$

end for

forever do

if the robot receives new sensory inputs o_n **do**

for each location l **do** / *apply the perception model*/

$$Bel^{(r)}(l) \leftarrow \alpha P\left(o \middle| l\right) Bel^{(r)}(l)$$

end for

e nd if

if the robot receives new odometry readings a_n **do**

for each location l **do** / *apply the perception model*/

$$Bel^{(r)}(l) \leftarrow \int P\left(l \middle| a, l'\right) Bel^{(r)}(l')\, dl'$$

end for

e nd if

if the robot is detected by the m-th environmental sensor **do**

for each location l **do** / *apply the detection model*/

$$Bel^{(r)}(l) \leftarrow Bel^{(r)}(l) \int P\left(L_i^{(r)} = l \middle| L^{(m)} = l', r^{(m)}\right) Bel^{(m)}(l')\, dl'$$

end for

e nd if

end forever

Table 1: Distributed sensors Markov localization algorithm for robot

Figure 3

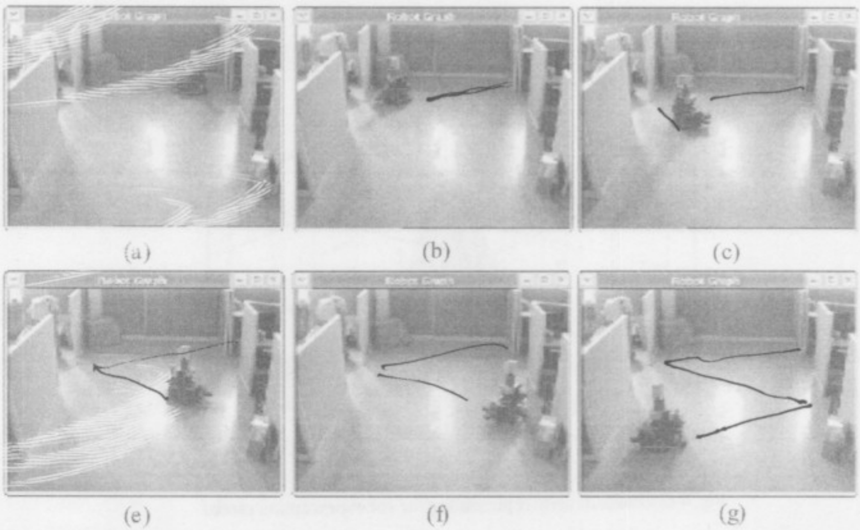
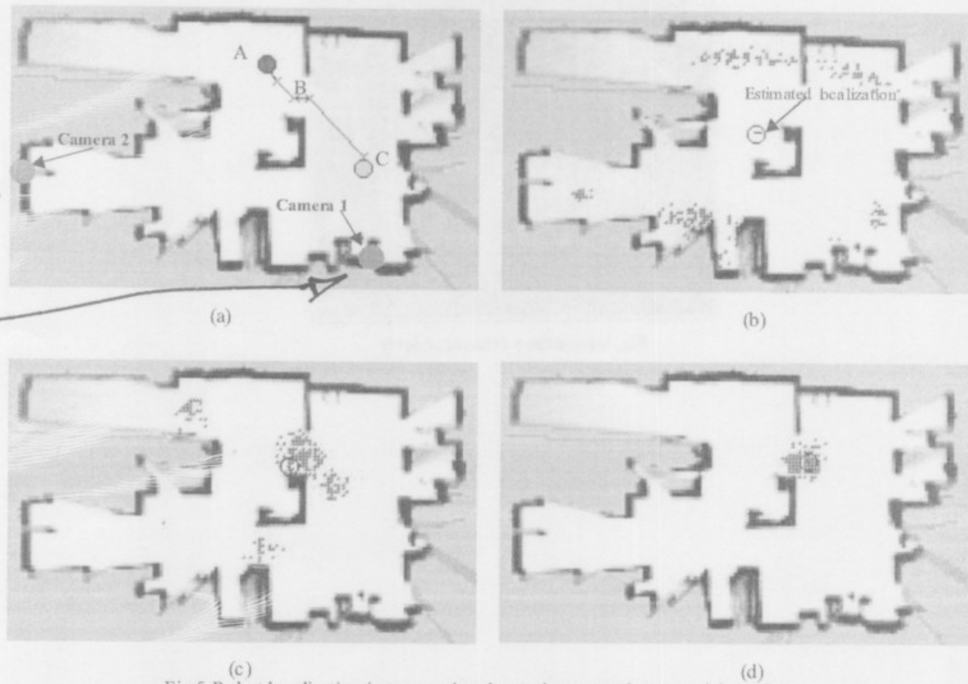


Fig. 3 Image sequences of successful detecting the robot which moves as a zig

This figure is not informative at all.
One cannot appreciate the ~~g~~ shape of the
Gaussian. A bird's eye view would have been better.

Figure 5



Please draw the field of view of the two cameras

Fig.5 Robot localization incorporating the environmental cameras' detection

Figure 7

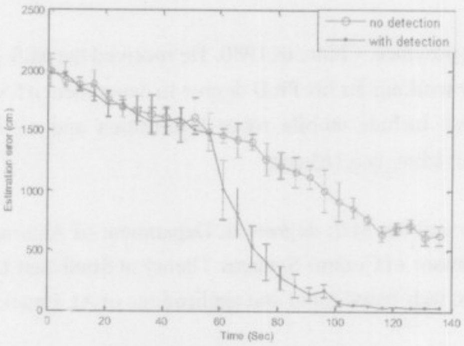


Fig.7 Comparison between no detection localization and localization making use of environmental detection.

Biography of each author

Zhiwei Liang was born in ShanDong province, China, in 1980. He received the M.S. degree in ShanDong Institute of Light Industry in 2004. He is currently working for his Ph.D degree in department of Automatic control from southeast university. His main research interests include mobile robot localization and navigation, distributed sensors and multi-sensor information fusion. Email: lzhw_ly@163.com

Xudong Ma received the B.S. degree and the M.S. degrees in Department of Automation from Tsinghua University, Beijing, China. He is currently a Professor of Control Systems Theory at Southeast University. His research interests are in the areas of Autonomous Robots, web-based robot and applications of AI. Email: xudma@seu.edu.cn.

Xianzhong Dai received Ph.D degree from the Tsinghua University, Beijing, China. He is currently a Professor of Control Systems Theory at Southeast University. His research interests are in the areas of control system, robotics and applications of AI. Email: xzdai@163.com.

Figure 6

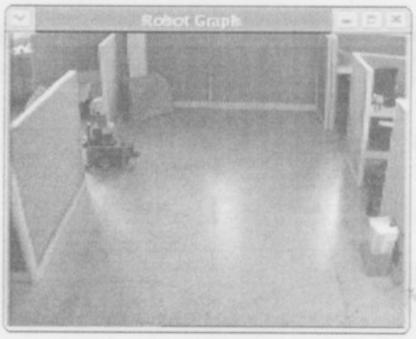


Fig. 6 Detection event at point B

Please draw the field of view of the two cameras

Figure 4

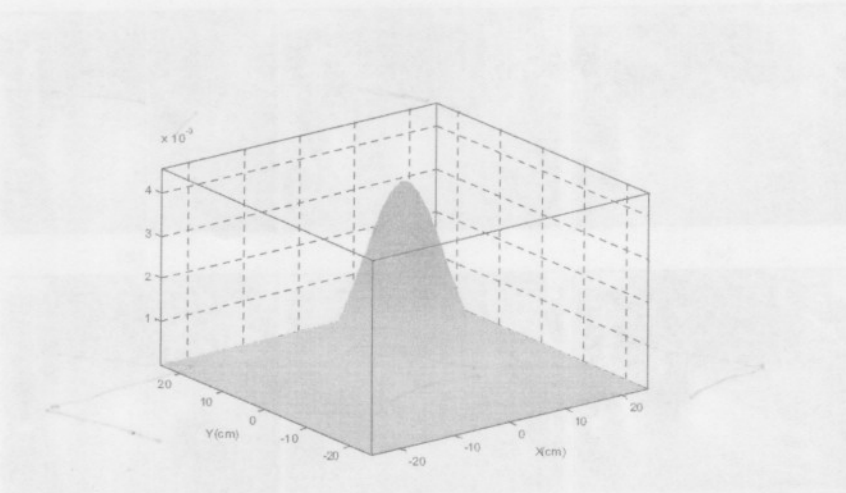


Fig. 4 Gaussian density representing the robot perception model

This figure is not informative at all. One cannot appreciate the shape of the Gaussian. A bird's eye view would have been better.

Figure 2

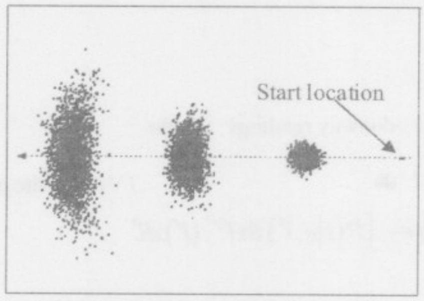


Fig. 2 Sampling-based approximation of the position belief for a non-sensing robot

? a robot moving with odometry only?

Figure 1

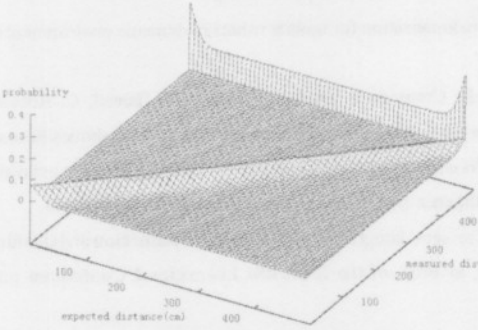


Fig. 1 : Perception model for laser range finders

actually is your implementation of MCL with laser scanner that performs quite poorly. One can do much better than plotted in Fig.7 with Laser in such simple environment.

Please explain better!

by the average distance of all samples from the reference position. The results are summarized in Figure 7. The graph plots the estimation error (y -axis) as a function of time (x -axis), averaged over the ten experiments, along with their 95%confidence intervals (bars). As can be seen in the figure, the quality of position estimation increases much faster when using environmental camera detection. Please note that the detection event typically took place 60-80 seconds after the start of an experiment. Obviously, this experiment is specifically well-suited to demonstrate the advantage of detections in robot global localization. Of course, the performance of our approach in more complex situations, especially highly symmetrical environments, is more attractable to solve robot's global localization.

This sentence is the strong point of your paper. You should redesign new experiment to show/stress this point.

5. Conclusion

In this paper we presented an approach to collaborative distributed sensors for mobile robot localization that uses a sample-based representation of the state space of a robot, resulting in an extremely efficient and robust technique for global position estimation. Here we use environmental cameras whose parameters is unknown in advance to determine robot's localization. In order to apply environmental cameras to localize the robot, all parameters of each environmental camera are calibrated independently by robot online. During calibration, the robot global localization is known and can navigate by its onboard laser sensor. Once calibrated, the environmental cameras can detect robot during robot localization. During localization, detections are used to introduce additional probabilistic constraints. To combine detection event of environmental camera, the robot's belief can reduce its uncertainty in response to the detection.

*

As a result, our approach makes it possible to collect sensor information at different platforms including robot itself. Experimental results demonstrate that our approach yields significantly better localization results than conventional MCL localization only based on robot's own sensors.

Additionally, the current approach possesses a limitation that warrants future research. The limitation is as follows: In our current system, only "positive" detections are processed. Not seeing robot is also informative, even though not as informative as positive detections. Incorporating such negative detections is generally possible in the context of our statistical framework (using the inverse weighing scheme). However, such an extension would drastically increase the computational overhead, and it is unclear as to whether the effects on the localization accuracy justify the additional computation and communication.

This is another interesting issue you should address in an improved version of your paper.

References

[1] N. Roy, W. Burgard, D. Fox, S. Thrun, Coastal navigation: Robot navigation under uncertainty in dynamic environments, in: Proc. of IEEE Int. Conf. on Robotics and Automation, Detroit, MI,1999, pp. 35-40.

[2] S. Thrun, D. Fox, W. Burgard, F. Dellaert, Robust Monte Carlo localization for mobile robots, Artificial Intelligence (AI) ,2001 (128), pp.99-141.

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

[4] L.P. Kaelbling, A.R. Cassandra, and J.A. Kurien, Acting under uncertainty: Discrete Bayesian models for mobile robot navigation, In Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 1996, pp.963-972.

[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 National Conference on Artificial Intelligence (AAAI), Portland, Oregon, USA, 1996, pp.896-901.

[6] D. Fox, W. Burgard, and S. Thrun, Markov localization for mobile robots in dynamic environments, Journal of Artificial Intelligence Research (JAIR), 1999(11), pp.391-427.

[7] R. Simmons and S. Koenig. Probabilistic robot navigation in partially observable environments, In Proc. of the International Joint Conference on Artificial Intelligence (IJCAI), Montreal, 1995, pp.1080-1087.

* I do not agree. The "few centimeter error" is a statistical results.

? Why the particles are only along the edges of the environment

Do you mean: "the system is already installed for different applications"?

at first. Assuming that the system is always ready for using in different environments, calibration instruments (such as patterns and measuring devices) may more or less hinder portability. Our objective is to introduce a self-calibration concept [20] into the system and take the mobile robot as a calibration instrument. Because the visual fields of all cameras are not overlaid, each camera's calibration is independent.

This is a VERY strong assumption. Is the robot localizing itself with a stronger method than the one described here?

During the calibration, the robot location is known. When the robot moves depending on laser and odometry in visual field of any environmental camera, the camera does detect the robot and gathers the relative data between the robot global location and detected image pixels. The sample space of relative data is designed to satisfy a condition that the distance between two neighbor global locations of relative data is more than 0.2 m. Once the number of relative data sums up to a threshold which is set as 200 in this paper, camera calibration can be conducted. Because the mobile robot always moves in a plane, the coplanar camera calibration method of Tsai's is adopted here [13].

I do not understand. Please explain better.

In addition, unlike ordinary calibration devices, the mobile robot is much less accurate when moving. As the most distinct point of the robot's error, it is cumulative and increase over time or repeated measurements. Moreover, the random motion input of the robot, which may take too much time, is not suitable for our method. For all these reasons, robot's motion during calibration process should be designed to avoid serious calibration error and to meet the accuracy demands of calibration. In our method, the robot in the cameras' visual field moves as a zig, which is shown in Figure 3.

what do you mean?

3.2.2. Detection

To determine the location of the robot, our approach combines visual information obtained from environmental cameras. Camera images are used to detect mobile robot and determine the relative position of the detected robot. The two row (s) in Figure 3 shows examples of camera images recorded in a room. Each image shows a (the) robot, marked by a unique, colored marker to facilitate its recognition. Even though the robot is only shown with a fixed orientation in the (this) figure, the marker can be detected regardless of the robot's orientation.

To find robots in a camera image, our approach first filters the image by employing local color histograms and decision trees tuned to the colors of the marker. Threshold is then employed to search for the marker's characteristic color transition. If found, this implies that the robot is present in the image. The small black points, superimposed on each marker in the images in Figure 3, illustrate the center of the marker as identified by this distributed environmental camera. Currently, images are analyzed at a rate of 10Hz.

Here is just a single camera.

Once a robot has been detected, the current environmental camera is analyzed for the location of the robot in image coordinates. Then it is done by transforming the detection pixels in image coordinates to positions in world coordinates according to the calibrated parameters of the camera. Here, tight synchronization of photometric data is very important, especially because the mobile robot might shift and rotate simultaneously when it is ~~sensed~~. In our framework, sensor synchronization is fully controllable because all data is tagged with timestamps.

Do you mean: "of the different cameras"?

3.2.3. Joint Detection Model

Next, we have to devise a joint detection model of the type $\int P\left(L^{(r)}=l \,\middle|\, L^{(m)}=l', r^{(m)}\right) Bel^{(m)}(l')\,dl'$. To recap, $r^{(m)}$ denotes a detection event by the m-th environmental camera, which comprises location of the detected robot in image coordinates. The variable $L^{(r)}$ describes the location of the detected robot, and $L^{(m)}$ ranges over locations of the m-th environmental camera. As described above, we will restrict our considerations to "positive" detections, i.e., cases where a(n) environmental camera did detect a robot. Negative detection events (a environmental camera does *not* see a robot) are beyond the scope of this paper and will be ignored.

The joint detection model is trained using data. More specifically, during training we assume that the exact to environmental sensor m . Then

$$\begin{aligned} Bel_t^{(r)}\left(L^{(r)}=l\right) &= P\left(L_t^{(r)}=l \middle| d_t\right) P\left(L_t^{(r)}=l \middle| r_t^{(m)}\right) \\ &= P\left(L_t^{(r)}=l \middle| d_t\right) \int P\left(L_t^{(r)}=l \middle| L_t^{(m)}=l', r_t^{(m)}\right) P\left(L_t^{(m)}=l' \middle| r_{t-1}^{(m)}\right) dl' \end{aligned} \tag{6}$$

Which suggests the incremental update equation:

$$Bel^{(r)}(l) \leftarrow Bel^{(r)}(l) \int P\left(L_t^{(r)}=l \middle| L^{(m)}=l', r^{(m)}\right) Bel^{(m)}(l')\, dl' \tag{7}$$

Here $\int P\left(L_t^{(r)}=l \middle| L^{(m)}=l', r^{(m)}\right) Bel^{(m)}(l')\, dl'$ describes environmental sensor's belief about the detected robot's position.

Table 1 summarizes robot Markov localization algorithm combining distributed sensors. The time index t and the state variable L is omitted whenever possible.

REFER TO A BOOK

3.□ Implementation of Distributed Sensors Localization

The previous section left open how the belief about the robot position is represented. In general, the space of all sensors and robot position is continuous-valued and no parametric model is known that would accurately model arbitrary beliefs in such robotic domains. However, practical considerations make it impossible to model arbitrary beliefs using digital computers.

?

3.1. Monte Carlo Localization

The vital idea here is to approximate belief functions using a Monte Carlo method. More specifically, our approach is an extension of Monte Carlo localization (MCL), which was ~~recently~~ proposed in [11, 14]. MCL is a version of Markov localization that relies on sample-based representations and the sampling/importance re-sampling algorithm for belief propagation [15]. MCL represents the posterior beliefs $Bel^{(r)}\left(L^{(r)}\right)$ by a set of K weighted random samples, or *particles*, denoted $S=\left\{s_i \middle| i=1\cdots K\right\}$. A sample set constitutes a discrete distribution and samples in MCL are of the type

$$s_i = \langle l_i, p_i \rangle$$

Where $l_i = \langle x_i, y_i, \theta_i \rangle$ denotes a robot position, and $p_i \geq 0$ is a numerical weighting factor, analogous to a discrete probability. For consistency, we assume $\sum_{i=1}^{K} p_i = 1$. In the remainder we will omit the subscript i whenever possible.

In analogy with the general Markov localization approach outlined in Section 2, MCL proceeds in two phases:

(1) **Robot motion.** When a robot moves, MCL generates K new samples that approximate the robot's position after the motion command. Each sample is generated by *randomly* drawing a sample from the previously computed sample set, with likelihood determined by their p-values. Let l' denote the position of this sample. The new sample's l is then generated by generating a single, random sample from $P\left(l \middle| l', a\right)$, using the odometry measurement a. The p-value of the new sample is K^{-1}. Figure 2 shows the effect of this sampling technique for a single robot, starting at an initial known position and executing actions as indicated by the dash line. As can be seen there, the sample sets approximate distributions with increasing uncertainty, representing the gradual loss of position information due to slippage and drift.

which generates measurements denoted by o. The measurements o establish the necessary reference between the robot's local coordinate frame and the environment's frame of reference. In our experiments below, o will be laser range scans.

(3) **Detections.** Additionally, robot queries distributed environmental sensors for the presence or absence of itself. The resulting measurements will be denoted by r. Robot detection might be accomplished through different sensors than environment measurements. In our experiments, we use environmental cameras (color camera embedded in the environment) for robot detection.

please explain what you mean by "joint detection model"

2.2. Markov Localization It seems you don't need this. Please cite relevant book chapter.

Before turning to the topic of this paper—collaborative distributed sensors localization—let us first review a common approach to robot localization only based on its own resource, which our approach is built upon: Markov localization. Markov localization uses only dead reckoning measurements a and environment measurements o; it ignores detections r. In the absence of detections, information gathered at different platforms cannot be integrated.

The key idea of Markov localization is that the robot maintains a belief over its position. The belief of robot at time t will be denoted by $Bel_t^{(r)}\left(L^{(r)}\right)$. Here L is a three-dimensional random variable composed of the robot's x - y position and its heading direction θ. Accordingly, $Bel_t^{(r)}\left(L^{(r)}=l\right)$ denotes the belief of the robot of being at a specific location l. Initially, at time $t=0$, $Bel_0^{(r)}\left(L^{(r)}\right)$ reflects the initial knowledge of the robot. In the most general case, which is being considered in the experiments below, the initial position of robot is unknown, hence $Bel_0^{(r)}\left(L^{(r)}\right)$ is initialized by a uniform distribution.

just "two" cameras are not enough convincing please add more cameras

At time t, the belief $Bel_t^{(r)}\left(L^{(r)}\right)$ is the posterior with respect to all data collected up to time t:

$$Bel_t^{(r)}\left(L^{(r)}\right)=P\left(L_t^{(r)}\middle|d_t\right) \tag{1}$$

where d_t denotes the data collected by the robot up to time t. By assumption, the most recent sensor measurement in d_t is either an environment or an odometry measurement. Both cases are treated differently, so let's consider the former first:

(1) **Sensing the environment**: Suppose the last item in d_t is an environment measurement, denoted o_t. Using the Markov assumption (and exploiting that the robot position does not change when the environment is sensed), we obtain for any location l:

$$\begin{aligned}
Bel_t^{(r)}\left(L^{(r)}=l\right) &= P\left(L_t^{(r)}=l\middle|d_t\right)\\
&= \frac{P\left(o_t\middle|L_t^{(r)}=l,d_{t-1}\right)P\left(L_t^{(r)}=l\middle|d_{t-1}\right)}{P\left(o_t\middle|d_{t-1}\right)}\\
&= \frac{P\left(o_t\middle|L_t^{(r)}=l\right)P\left(L_t^{(r)}=l\middle|d_{t-1}\right)}{P\left(o_t\middle|d_{t-1}\right)}\\
&= \alpha P\left(o_t\middle|L_t^{(r)}=l\right)P\left(L_t^{(r)}=l\middle|d_{t-1}\right)\\
&= \alpha P\left(o_t\middle|L_t^{(r)}=l\right)P\left(L_{t-1}^{(r)}=l\middle|d_{t-1}\right)\\
&= \alpha P\left(o_t\middle|L_t^{(r)}=l\right)Bel_{t-1}^{(r)}\left(L_t^{(r)}=l\right)
\end{aligned} \tag{2}$$

NOT TRUE

Where α is a normalizer that does not depend on the robot position l. Notice that the posterior brief

cite Menegatti } for distributed sensors
Stroupe } measurement merging.

* Manuscript

A probabilistic approach to collaborative distributed sensors for mobile robot localization

Zhiwei Liang, Xudong Ma, Xianzhong Dai

Department of Automatic Control, Southeast University, Nanjing, Jiangsu Province, China

Abstract

This paper presents a probabilistic algorithm to collaborative distributed sensors for mobile robot localization. Our approach uses a sample-based version of Markov localization, capable of localizing mobile robot in an any-time fashion. During robot localization given a known environment, probabilistic method is employed to synchronize robot's belief whenever one environmental sensor detects robot. We present an implementation that uses color environmental cameras for robot detection. All the parameters of each environmental camera are unknown in advance and can be calibrated independently by robot online. Once cameras' calibrated, the joint detection models are trained using sensor data according to cameras' parameters. As a result, the robot's belief can reduce its uncertainty in response to the detection effectively. A further experiment, obtained with the real robot in an indoor office environment, illustrates that drastic improvements in localization speed and accuracy when compared to conventional robot localization only based its own sensors.

Keywords: Collaborative distributed sensors localization; Environmental camera; Joint detection model

1. Introduction

keep tracks of its motion

Mobile robot localization is the problem of estimating a robot's pose (location, orientation) relative to its environment. The localization problem is a key problem in mobile robotics. There are two classes of localization problem, position tracking and global localization. In position tracking, a robot knows its initial position [1] and only needs to ~~reduce uncertainty in the odometer reading~~. If the initial position is not known or the robot is kidnapped to somewhere, the problem is ~~one of~~ global localization, i.e., the mobile robot has to estimate its global position through a ~~sequence of sensing~~ actions [2]. In recent years, a flurry of publications on localization documents the importance of the problem. Occasionally, it has been referred to as "the most fundamental problem to ~~providing~~ provide a mobile robot with autonomous capabilities" [3].

called

the problem mounted of the

So far, virtually all existing works addresses localization only using sensors onboard ~~mobile~~ robot. However, in robot navigation, the robot cannot always determine its unique situation only by local sensing information since the sensor is prone to errors and a slight change of the robot's situation deteriorates the sensing results. Along with the rapid development of computer networks and multimedia technology, research on how to make an 'intelligent' environment ~~for the robot to fulfill the same functions~~ makes sense, especially in home environment. In this case, ~~various~~ sensors are embedded into the environment (environmental sensors), and communication between the robot and environmental sensors is utilized. The problem of combining distributed sensors (including the sensors of robot itself) to localize mobile robot remains virtually unexplored. At first glance, one could solve the problem of localizing robot by ~~localizing robot~~ *independently*, which is a valid approach that might yield reasonable results in many

to support robot localization

giving an estimation of the robot position

for each sensor

Tel.:+086 25 8379 2418
Email address: lzhw_ly@hotmail.com

I would not say that. Several works on robot loc. with camera networks. Small Size League

NOT true. Find references: MENEGATTI ECMR05 SAFFIOTTI