2009 IEEE International Conference on Robotics and Automation
Kobe International Conference Center
Kobe, Japan, May 12-17, 2009

Integration of a Prediction Mechanism with a Sensor Model: An Anticipatory Bayes Filter

Guoxuan Zhang and Il Hong Suh, *Senior Members, IEEE*

Abstract— **In the task of robot localization, Bayes filters use two processes: the prediction step and the measurement-update step. Briefly, the state transition model is responsible for prediction, and the sensor model is responsible for measurement updates. This paper presents a new approach to the sensor model, called the predictive sensor model, which utilizes a prediction mechanism to improve the efficiency of measurement updates in Bayes filters. By adding sensorial anticipation, we extend the original Bayes filter to an anticipatory Bayes filter. We also propose an entropy-based place-segmentation method for automatic segmentation of sequentially collected vision-sensor data. Our place segmentation technique is most useful for node clustering in the process of constructing topological maps. Our work was verified by experiments using observed data.**

I. INTRODUCTION

In the field of robot mapping and localization problems, Bayes filters are single dominating schemas for integrating temporal data. Bayes filters are highly related to Kalman filters, hidden Markov models, dynamic Bayes networks, and partially observable Markov decision processes [1]. Bayes filters extend Bayes' rule to temporal estimation problems. They use a recursive estimator to compute a sequence of posterior probability distributions over quantities that cannot be observed directly.

A Bayes-filter algorithm usually has two essential steps: *i.e.*, a prediction step and a measurement-update step [2]. In the prediction step, it calculates a belief for the newly predicted state based on the prior belief for the state at a previous time step and the control information. In the measurement-update step, the Bayes filter multiplies the belief for the predicted state by the probability that the new measurement may have been observed. In this step, the probability provided by robotic measurement is called a sensor model. In research on mobile robotics, several methods have been suggested for utilizing the predicted state to enhance the performance of the sensor model.

Mind RACES [3] was a three-year EC-funded project that focused mainly on the concept of anticipation. This systematic research regarding the specialized topic of anticipation produced many important publications.

Butz *et al*. discussed how anticipatory mechanisms may be beneficial for adaptive behavior [4]. From a computation-oriented perspective, the question arises of how predictions, predictive representations, or knowledge about the future can influence sensory processing, learning, decision making, and motor control. They explained the process of sensorial anticipation in more detail [5]. The prediction of future states, and thus the prediction of future stimuli, influences stimulus processing. They proposed a sensory anticipatory mechanism as shown in Fig. 1. To be able to make predictions, the agent must use a predictive model of its environment. That is, sensory anticipation is strongly related to preparatory attention in psychology, in which top-down processes such as task-related expectations influence sensory processing. Sensory anticipatory behavior results in a predisposition of processing sensory input. For example, the agent may become more susceptible to specific sensory input and ignore other sensory input.

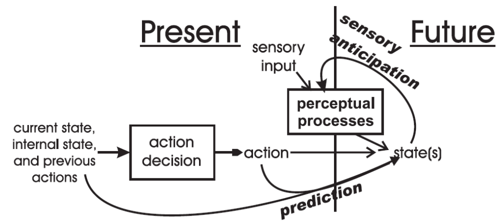


Fig. 1. Sensory anticipatory behavior influences, or predisposes, sensory processing due to future predictions, expectations, or intentions.

There have been few studies dedicated to the application of prediction in robot mapping and localization. An experimental study was reported previously [6] in which two robots had the goal of exchanging places while navigating through an area with or without obstacles. In another study [7], the author proposed an episodic-memory-based approach for computing anticipatory robot behavior in a partially observable environment, and presented the results of a robot-navigation experiment.

A number of different types of map have been proposed in the fields of robotics and artificial intelligence, such as metric, topological, and hybrid metric-topological maps. We implemented our approach using a topological map for several reasons. The greatest advantage of topological maps is that they can facilitate the interaction of the robot with humans, because the elements in the topological map (e.g., the nodes in a graph) can be made to correspond to concepts that make sense to humans (*e.g.*, rooms, corridors, *etc.*), instead of metric (x, y) coordinates that are not intrinsically meaningful to humans in office and home environments [8]. One way to achieve dynamic information exchange

This work was supported by the IT R&D program of MKE/IITA. [2008-F-038-01, Development of Context Adaptive Cognition Technology]

G. Zhang is with the College of Information and Communications, Hanyang University, Korea. (E-mail: imzgx@incorl.hanyang.ac.kr).

I. H. Suh is with the College of Information and Communications, Hanyang University, Korea. (E-mail: ihsuh@hanyang.ac.kr. All correspondence should be addressed to I. H. Suh.)

between robots and their operators is through mapping the environment at the semantic level [9]. Semantic mapping can provide the means to collaboratively build a shared understanding of an environment. It is easier to assign semantic meaning to map elements with a topological map than with a metric map.

In topological mapping, the first process should be node generation. Different approaches have been proposed in earlier work. Prior methods tended to use an *ad hoc* approach or have a human supervisor indicate which points are to serve as the nodes in the topological map. Alternatively, an existing metric representation was used to derive a higher-level topological map using geometrical methods, such as generalized Voronoi graphs [10]. It is also possible to use the features of sensory data directly for the creation of a higher-level map. In this paper, we propose an entropy-based approach to generate nodes autonomously.

In Section II we present our approach to segmenting places based on analysis of the rate of change of entropy data calculated from vision sensor input. In Section III, we describe how to extend the original Bayes filter by applying a prediction mechanism. Experiments using a real robot platform and conclusions are presented in sections IV and V respectively.

II. Entropy-based Place Segmentation

Topological maps are graph-like spatial representations in which nodes represent states in the agent's configuration space and edges represent the system trajectories that take the agent from one state to another. The meanings of nodes and edges in a topological map vary according to the application as well as the algorithms used to build them [11].

We represent the nodes of a topological map as *distinct places* and edges as *transitions* that a robot makes as it goes from one place to another. When the robot navigates a path, positions that share similar sensorial features are grouped. In our approach, the problem of node generation can be viewed as a problem of place segmentation. This section describes how a robot can use multi-dimensional feature data to automatically identify distinct places in a topological map.

The robot must simultaneously process multi-dimensional feature data, according to the various sensor data obtained from different types of apparatus, and there are a variety of methods to process this raw data. To apply a unified mechanism for place segmentation, it is necessary to transform these multi-dimensional features to those of a comparable data type that shares the same metric unit.

We take the form of $f^i \rightarrow z^i$ to denote our proposed transformation. Here, f^i is the measured value on the ith dimension feature, and z^i is the transformed value for f^i, where $z \in [0, 1]$. The detailed methods of transformation depend on the data of each feature that belong to a different dimension and the properties of that data type, and the magnitude of the transformed value will follow the importance of feature type and the actual measured value. We attach the subscript t for z_t^i to reflect the fact that feature values are measured at consecutive time points.

Now, we can compute the occupancy of the ith feature on total features at time t using the following equation:

$$P(z_t^i) = \frac{z_t^i}{\sum_i z_t^i} \tag{1}$$

Here, we propose a method of entropy-based place segmentation. The entropy is a measure of the average uncertainty in the random variable [12]. Given a set of sensor data, the problem of place segmentation can be viewed as a process of clustering adjacent data into different groups. As a result, the feature data within the same group will show up gradual value changes, but the data lay at the boundaries will show up marked changes. These can either be clustered as the member of both side groups. In this context, we mapped the concept of changes in feature data to the concept of uncertainty in entropy representation.

When each $P(z_t^i)$ is calculated from equation (1), the entropy for the total feature data can be easily computed:

$$H(\mathbf{z}_t) = -\sum_i P(z_t^i) \log[P(z_t^i)] \tag{2}$$

Here, $\mathbf{z}_t$ is a multi-dimensional feature vector measured at time t. By tracking the changing rate of uncertainty in measurement data, we can pick out a series of meaningful candidates for segment points from sequential input data. These segment points should be corresponding to the places where the changing rate of entropy will take local maxima. As our goal is machinery computation, we calculate the changing rate of uncertainty by taking the differences in consequent entropy values:

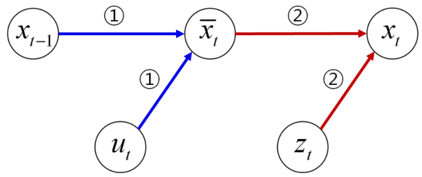
$$\frac{d[H(\mathbf{z}_t)]}{dt} = H(\mathbf{z}_t) - H(\mathbf{z}_{t-1}) \tag{3}$$

The result provides only information about potential candidate points for place segmentation, so two more steps are needed to refine the results. The first is to take the threshold for these data, and the second is to merge adjacent regions by comparing the Euclidean distance of z_t on the results obtained in the first step.

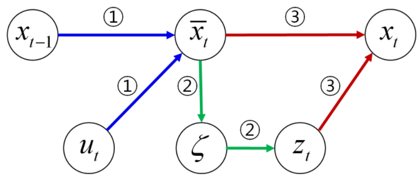
Each resulted region represents distinct places at robot environment, and we can adopt these boundaries between adjacent places as segment points to perform place segmentation. As the segment points correspond to marked changes in the quantity of multi-dimensional input data, they can effectively denote places where sensory features have newly appeared or existing features have disappeared.

III. Anticipatory Bayes Filter

As noted above, a Bayes filter includes prediction and measurement-update steps; these two steps are presented in equations (4) and (5), respectively. In Fig. 2(a), arrows marked with "1" indicate the processes of the prediction step, while those marked with "2" indicate the processes corresponding to the measurement-update step. In the Bayes filter equation, we follow the common notation using x, u, and z to refer to state, control, and measurement, respectively, and can therefore present the Bayes filter as:



(a) Original Bayes filter; 1: prediction step, 2: measurement update step.



(b) Anticipatory Bayes filter; 1: prediction step, 2: sensorial anticipation step, 3: measurement update step.

Fig. 2. The structure of an Anticipatory Bayes Filter.

$$Bel(\bar{x}_t) = \sum_{x_{t-1}} P(\bar{x}_t|u_t, x_{t-1})Bel(x_{t-1}) \tag{4}$$

$$Bel(x_t) = \eta P(\mathbf{z}_t|\bar{x}_t)Bel(\bar{x}_t) \tag{5}$$

where $\bar{x}_t$ is the predicted state that is the product of the prediction step, the probability $P(\bar{x}_t|u_t, x_{t-1})$ is used to model the motion of robots and is called the motion model, the probability $P(\mathbf{z}_t|\bar{x}_t)$ is used to model the measurement of robots and is called the sensor model, and η is the normalizing constant. To allow distinction from the new sensor model described later, we hereafter call the probability $P(\mathbf{z}_t|\bar{x}_t)$ the original sensor model. Our objective is to extend the original sensor model, so a prediction mechanism should be included in this model. In our approach, prediction is composed of three elements: knowledge formed from past experience of measurement, the current state used to activate the process of prediction, and an inhibition weight to counter confusion that arises as a consequence of prediction.

A. *Predictive Sensor Model*

To apply the sensorial anticipation mechanism of Fig. 1 to the Bayes filter shown in Fig. 2(a), it is necessary to add one more step to the original Bayes filter. This step will be responsible for applying sensory anticipation to perceptual processes, so sensory input will be affected by prediction activity. We extend $P(\mathbf{z}_t|\bar{x}_t)$ in equation (5) with sensorial experience ζ to compose a new sensor model, $P(\mathbf{z}_t|\bar{x}_t, \zeta)$, and call it the Predictive Sensor Model (PSM). Now, we use Bayes rules to expand PSM to determine what insight it provides in this new model:

$$\begin{aligned} P(\mathbf{z}_t|\bar{x}_t, \zeta) &= \eta P(\mathbf{z}_t, \bar{x}_t, \zeta) \\ &= \eta P(\mathbf{z}_t, \bar{x}_t)P(\zeta|\mathbf{z}_t, \bar{x}_t) \\ &= \eta P(\mathbf{z}_t, \bar{x}_t)P(\zeta|\bar{x}_t) \\ &= \eta' P(\mathbf{z}_t|\bar{x}_t)P(\zeta|\bar{x}_t) \end{aligned} \tag{6}$$

From the second to the third line, we apply conditional independence to $P(\zeta|\mathbf{z}_t, \bar{x}_t)$, where knowledge ζ is independent of temporal measurement $\mathbf{z}_t$ when state x_t is given. Compared to the expansion of the original sensor model:

$$\begin{aligned} P(\mathbf{z}_t|\bar{x}_t) &= \eta'' P(\mathbf{z}_t, \bar{x}_t) \\ P(\mathbf{z}_t, \bar{x}_t) &= \frac{1}{\eta''} P(\mathbf{z}_t|\bar{x}_t) \end{aligned} \tag{7}$$

PSM has one more term $P(\zeta|\bar{x}_t)$ than the original sensor model, and we call this term the Sensorial Anticipation Model (SAM).

SAM provides a method of applying a prediction mechanism to the Bayes filter, and this is the main contribution of this paper. Experience is comprised of states and measurements related to episodes. SAM shows how to use knowledge derived from experience to improve sensor processing when the state is given. Then, the probability value $P(\zeta|\bar{x}_t)$ can be explained as a boosting or suppressing factor in certain sensorial measurements. After introducing SAM as an application in the prediction mechanism, the original Bayes filter can be extended to a new form of Bayes filter as described below:

$$Bel(\bar{x}_t) = \sum_{x_{t-1}} P(\bar{x}_t|u_t, x_{t-1})Bel(x_{t-1}) \tag{8}$$

$$P(\mathbf{z}_t|\bar{x}_t, \zeta) = \eta P(\mathbf{z}_t|\bar{x}_t)P(\zeta|\bar{x}_t) \tag{9}$$

$$Bel(x_t) = \eta P(\mathbf{z}_t|\bar{x}_t, \zeta)Bel(\bar{x}_t) \tag{10}$$

Of course the terms η in the second and third lines are not the same, but we use this term indiscriminately for convenience. Equation (9) comprises the core of above process, and we name this the sensorial anticipation step. When PSM is calculated from this step, it is substituted to the measurement-update step instead of the original sensor model $P(\mathbf{z}_t|\bar{x}_t)$. Based on PSM, we obtain a new version of the measurement-update step, and we call the overall process the Anticipatory Bayes Filter.

In Fig. 2(b), the arrows marked “2” represent the processes of the sensorial anticipation step. Combining the knowledge related to the predicted state $\bar{x}_t$, the measurement input $\mathbf{z}_t$ is affected by the sensorial anticipation model in this step. Finally, the arrows marked “3” represent the processes of the PSM-applied measurement-update step.

It only remains to define the meaning of PSM. The influence of sensorial anticipation on perceptual processing can be explained as the robot becoming more susceptible to specific sensory input and ignoring other sensory input. In the remainder of this paper, we will focus on the aspect of sensorial ignorance. That is, we also express ignoring sensory input by suppressing dispensable sensory input. For the purpose of calculating PSM, we introduce two types of state, the winner state $\tilde{x}_t$ and the confusing state $\hat{x}_t$, which are defined as follows:

$$\tilde{x}_t = \underset{x \in X}{\operatorname{argmax}} \sum_{x_{t-1}} P(\bar{x}_t|u_t, x_{t-1})Bel(x_{t-1}) \tag{11}$$

$$\hat{x}_t \in \{x_t : P(\mathbf{z}_t|x_t) > P(\mathbf{z}_t|\tilde{x}_t)\} \tag{12}$$

The winner state $\tilde{x}_t$ has the highest belief value at the prediction step, and the confusing state $\hat{x}_t$ has a higher score than $\tilde{x}_t$ when they are evaluated by the original sensor model for the same measurement $\mathbf{z}_t$. We can imagine the confusing state as an outlier from the viewpoint of pattern classification. As mentioned before, PSM can be described as a weight term that operates on unwanted sensory input, and we name it the inhibition weight $w_{\tilde{x}_t\hat{x}_t}$. When the winner state $\tilde{x}_t$ is calculated at the prediction step, the inhibition weight has the effect of suppressing unwanted sensor inputs, which seem to be more frequently observed in confusing states $\hat{x}_t$.

B. Inhibition Weight

As described in the section dealing with entropy-based place segmentation, measurement $\mathbf{z}_t$ presents the multi-dimensional feature vector $(z_t^1, z_t^2 \cdots z_t^n)$. Assuming these features are conditionally independent given state $\bar{x}_t$, the original sensor model can be expressed in a multiplication form:

$$\begin{aligned} P(\mathbf{z}_t|\bar{x}_t) &= P(z_t^1, z_t^2 \cdots z_t^n|\bar{x}_t) \\ &= P(z_t^1|\bar{x}_t)P(z_t^2|\bar{x}_t)\cdots P(z_t^n|\bar{x}_t) \\ &= \prod_i P(z_t^i|\bar{x}_t) \end{aligned} \tag{13}$$

According to the definition of confusing state $\hat{x}_t$, the following inequalities stand:

$$\begin{aligned} P(\mathbf{z}_t|\hat{x}_t) &> P(\mathbf{z}_t|\tilde{x}_t) \\ \prod_i P(z_t^i|\hat{x}_t) &> \prod_i P(z_t^i|\tilde{x}_t) \end{aligned} \tag{14}$$

At this point, we know only that in equation (14) the term on the left is greater than that on the right, but still cannot decide which dimensions of features contribute to this relationship. It is necessary to find the features that must be suppressed by using inhibition weights. For the purpose of expressional conciseness, we adopt the new notation $LoP(x)$ to present $\log(P(x))$, and then design a new threshold γ to check which dimensions of features contribute to the aforementioned inequality relationship:

$$\gamma = \frac{LoP(\mathbf{z}_t|\hat{x}_t) - LoP(\mathbf{z}_t|\tilde{x}_t)}{N}, N = |\mathbf{z}| \tag{15}$$

Here, N is the cardinality of the feature space. By applying a logarithm operator to the multiplication formula, we can calculate the averaged difference between these two terms. After threshold γ is given, we can then pick out confusing feature $\hat{z}^i$ using the criterion:

$$\hat{z}^i \in \left\{z^i : LoP(z^i|\hat{x}) - LoP(z^i|\tilde{x}) > \gamma\right\} \tag{16}$$

Confusing feature $\hat{z}^i$ are considered to be those features that contribute to inequality (14), and can also be explained as the cause of generation of outlier data. The ratio of the original sensor model applied to the confusing state and predicted state is defined as:

Applying Predictive Sensor Model
1: get measurement $\mathbf{z}_t$ at time t
2: calculate the winner state $\tilde{x}_t$
3: when $P(\tilde{x}_t)$ is smaller than θ, then return to 1
4: apply PSM according to the result of $Draw(P(\tilde{x}_t))$
5: continue loop 1 through 4.

Fig. 3. PSM algorithm

$$\kappa^i_{\tilde{x}_t\hat{x}_t} = \frac{P(z^i|\hat{x}_t)}{P(z^i|\tilde{x}_t)} \tag{17}$$

We can use this ratio $\kappa^i_{\tilde{x}_t\hat{x}_t}$ to measure the degree to which the confusing state causes ambiguity against the predicted state. As we have already picked out confusing features from equation (16), we can now multiply the value of $\kappa^i_{\tilde{x}_t\hat{x}_t}$ by the confusing features m to calculate inhibition weight:

$$w_{\tilde{x}_t\hat{x}_t} = \frac{1}{\prod_m \kappa^m_{\tilde{x}_t\hat{x}_t}} \tag{18}$$

From the definition of a confusing feature, $\kappa^i_{\tilde{x}_t\hat{x}_t}$ is always greater than 1 when z^m is the confusing feature, and accordingly $w_{\tilde{x}_t\hat{x}_t}$ is less than 1.

C. Anticipatory Bayes Filter

To perform prediction, robots must have knowledge derived from experience as *a prior* requirement, and can then use the present input to activate this knowledge; therefore, the future state of robots can be predicted reliably. For PSM, we use an inhibition weight to replace the functionality of SAM, so a practical way to obtain weight values is needed.

Here, we present a method to determine the values of inhibition weights. First, we gather the training data D for each state j. For D, perform the following steps sequentially. Extract confusing state k on j with formula (12); obtain confusing feature i within k using criterion (16); compute κ^i_{jk} for i with equation (17); compute inhibition weight w_{jk} using equation (18). Finally, calculate the expected inhibition weight over D and use it as SAM.

Once an inhibition weight has been obtained, we can then substitute this value into the Anticipatory Bayes Filter to implement a practical version:

$$Bel(\bar{x}_t) = \sum_{x_{t-1}} P(\bar{x}_t|u_t, x_{t-1})Bel(x_{t-1}) \tag{19}$$

$$P(\mathbf{z}_t|\bar{x}_t, \zeta) = \eta P(\mathbf{z}_t|\bar{x}_t)w_{\tilde{x}_t\hat{x}_t} \tag{20}$$

$$Bel(x_t) = \eta P(\mathbf{z}_t|\bar{x}_t, \zeta)Bel(\bar{x}_t) \tag{21}$$

Here, $w_{\tilde{x}_t\hat{x}_t} = 1$ when PSM is not applicable, in which case the Anticipatory Bayes Filter is exactly the same as the original Bayes filter. There are two cases where PSM is not applicable; one is that the belief at the predicted state is too low, and the other is when there is no confusing state for some state in the training process.

Applying PSM in the Anticipatory Bayes Filter should be performed carefully. If the predicted state is proven to

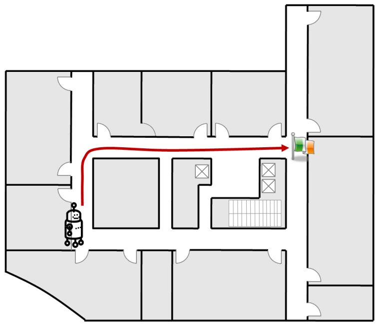


Fig. 4. Experimental environment. The start and finish points are marked with the robot and flag icons respectively.

Fig. 5. Real robot platform. A Pioneer 3-AT mounted with three web cameras for image gathering.

be correct, then the Anticipatory Bayes filter can improve the measurement-update step as intended. However, if the predicted state is proven to be incorrect, this can cause an even worse result. We propose the statistical algorithm shown in Fig. 3 to avoid this problem. In this algorithm, the function $Draw(P(\tilde{x}_t))$ determines whether PSM is performed, according to the results of sampling over distribution $P(\tilde{x}_t)$.

IV. EXPERIMENTS

This paper proposed two methods for robot mapping and localization tasks: entropy-based place segmentation and the Anticipatory Bayes Filter. We evaluated our approaches through actual robot experiments in an indoor environment as shown in Fig. 4. We used Pioneer 3-AT as the real robot platform in the experiments, with three Web cameras mounted on top. These cameras were placed on the same plane and at a height of 100cm from the floor; one faced leftward, one forward, and one rightward. Images at a resolution of 320 by 240 pixels were collected from the three cameras in turn, at a frame rate of 10 fps. However, we used only the images from the leftward- and rightward-facing cameras for our experiments.

A. Visual Data Encoding

There are a number of methods for encoding vision data, and the choice of method depends mainly on the intended application. In corridor-like environments, there are few visual features that can be uniquely identified. However, there are numerous straight lines formed by interior moldings, doors, and vanishing lines as shown in Fig. 7; therefore in processing vision data in our experiments, we decided to adopt solely the line features.

First, all line segments in an image were extracted into one of four groups according to the angle formed between the line and the horizontal plane. The four groups were divided according to the quantized angles of 0^o, 45^o, 90^o and 135^o, respectively. After allocating a line to a group, we added all line lengths in the same group to form a four-element histogram. As we used images taken from the left and right cameras, we had eight values at the end of each image acquisition step. These data were put into one vector and divided by the length of that vector, which yielded a normalized vector of length 1. By inspecting this vector value, we obtained an understanding of the presentation of line components and their quantitative tendencies in the measurement data.

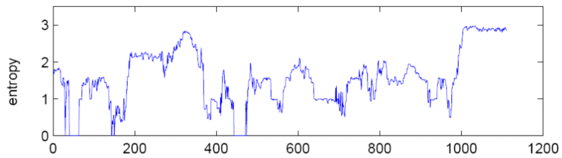
B. Place Segmentation

The first experiment was performed to evaluate entropy-based place segmentation. In Fig. 4, the start and finish points are marked with robot and flag icons, respectively; the distance between these two points was 25 m. As the robot navigated from the start to the finish, 1112 sets of images were consequently taken by the left- and right-hand cameras. The entropy-based place segmentation algorithm generated 14 segment points. Figure 6 shows the data processing for entropy, derivatives of entropy at (a) and (b), and thin vertical lines as the resulting segment points at (c). Some typical images taken at these segment points are shown in Fig. 7. Most segment points were related to the time indices where vertical lines either appeared or disappeared. There were also some points (fourth and eighth images in Fig. 7) that corresponded to dramatic changes in the line components. As shown by these results, the entropy-based place segmentation algorithm provides a very efficient approach for segmenting sequential data in a natural way.

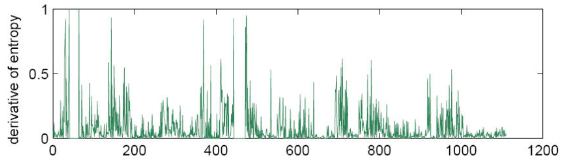
C. Anticipatory Bayes Filter

This experiment had two purposes: to evaluate the improvement in performance of the sensor model and to assess the total enhancement attributable to the Anticipatory Bayes filter. We compared the performance of the original sensor model and PSM by measuring the improvement in the resulting probability values. The main difference between the original sensor model and PSM is that the latter utilizes the information of the predicted state $\bar{x}_t$, to filter out ambiguity derived from confusing states. As equation (9) shows, PSM is based on multiplication of the original sensor model by SAM, so the two models share the process of computing used in the original sensor model.

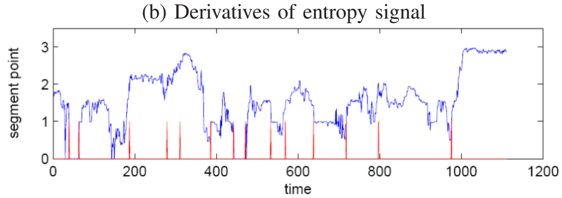
We constructed our original sensor model from the viewpoint of a distinct place classification problem. Using labeled feature data collected in the training phase, the original



(a) Entropy signal



(b) Derivatives of entropy signal



(c) Thin vertical red lines as segment points

Fig. 6. Data representation for entropy-based place segmentation.

Fig. 7. Images taken at segment points. At these time indices the change in entropy reached local maxima.

sensor model for each dimensional feature was computed at all distinct places. An inhibition weight $w_{\tilde{x}_t \hat{x}_t}$ for confusing states at each winner state was also computed using the algorithm described in Section III. Then, PSM was easily calculated using equation (20). The total correct score of the original sensor model for distinct place classification was 44.42%, whereas that of PSM was 56.65%, a performance improvement of 27.53%. Here, *correctly classified* means that a place was correctly identified based on the highest probability value.

Our experiment showed that the results of evaluation using the Bayes filter provide improved performance. The scores using the original Bayes filter and the Anticipatory Bayes filter were 57.01% and 75.63%, respectively, indicating a performance improvement of 32.65%. These two scores were measured in the same manner as the experiments conducted for the sensor model. A related demonstration video is available [13]. This enhancement of the Bayes filter is mainly due to integration of the prediction mechanism with the sensor model, allowing knowledge acquired by experience to be used in a current situation. As these experiments were performed in a rather simple environment, more comprehensive evaluation must be performed in future studies.

V. Conclusions and Future Work

In this paper, we proposed a predictive sensor model to improve the efficiency of the measurement-update step in the Bayes filter. We proposed an anticipatory Bayes filter based on the addition of a sensorial anticipation step to the ordinary Bayes filter. Our experimental results show that applying the prediction mechanism to the Bayes filter affects the perceptual process of sensor input, and provides higher accuracy than the original sensor model. We also implemented an entropy-based place segmentation method, which was shown to be a very reliable approach for automatically selecting segment points for node clustering.

In our experiments, we tested our proposed methods only in a simple environment. In future work, we will evaluate the degree of improvement that the predictive sensor model brings to overall localization performance. We will also refine the method of computing inhibition weights for the predictive sensor model, and apply the method to large-scale problems.

References

[1] S. Thrun, "Robotic Mapping: A Survey," *Exploring Artificial Intelligence in the New Millennium*, 2002.

[2] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. MIT press, Cambridge, Massachusetts, USA, 2005.

[3] http://www.mindraces.org/.

[4] M. Butz, O. Sigaud, G. Pezzulo, and G. Baldassarre, "Anticipations, Brains, Individual and Social Behavior: An Introduction to Anticipatory Systems," *LECTURE NOTES IN COMPUTER SCIENCE*, vol. 4520, p. 1, 2007.

[5] M. Butz, O. Sigaud, and P. Gerard, "Internal Models and Anticipations in Adaptive Learning Systems," *LECTURE NOTES IN COMPUTER SCIENCE*, pp. 86–109, 2003.

[6] B. Johansson and C. Balkenius, "An Experimental Study of Anticipation in Simple Robot Navigation," *LECTURE NOTES IN COMPUTER SCIENCE*, vol. 4520, p. 365, 2007.

[7] Y. Endo, "Anticipatory robot control for a partially observable environment using episodic memories," *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pp. 2852–2859, 2008.

[8] Z. Zivkovic, B. Bakker, and B. Krose, "Hierarchical map building using visual landmarks and geometric constraints," *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, pp. 2480–2485, 2005.

[9] F. Dellaert, D. Bruemmer, and A. Workspace, "Semantic SLAM for Collaborative Cognitive Workspaces," *AAAI Fall Symposium Series 2004: Workshop on The Interaction of Cognitive Science and Robotics: From Interfaces to Intelligence, 2004*.

[10] H. Choset and K. Nagatani, "Topological simultaneous localization and mapping (SLAM): toward exact localization without explicit localization," *Robotics and Automation, IEEE Transactions on*, vol. 17, no. 2, pp. 125–137, 2001.

[11] E. Remolina and B. Kuipers, "Towards a general theory of topological maps," *Artificial Intelligence*, vol. 152, no. 1, pp. 47–104, 2004.

[12] T. Cover and J. Thomas, *Elements of Information Theory*. Wiley-Interscience New York, 2006.

[13] http://incorl.hanyang.ac.kr/.