The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

# Bayesian Robot Localization with Action-Associated Sparse Appearance-Based Map in a Dynamic Indoor Environment

Young-Bin Park, Il Hong Suh and Byung-Uk Choi

***Abstract*— This work considers robot localization with an action-associated sparse appearance-based map, under conditions with dynamic change in the environment. In this case, two significant problems must be solved for robust localization. The first involves variations in the environment caused by dynamic objects and changes in illumination, and the second arises from the nature of sparse appearance-based map. That is, a robot must be able to recognize observations taken at slightly different positions and angles within a certain region as identical. In this paper, we address a possible solution to these problems on the basis of a probabilistic model called the Bayes filter. Here, we propose an observation model based $LeTO^2$ function and an action-associated sparse appearance-based map to be used for prediction, update, and final localization steps. In addition, multiple visual features are used to increase the reliability of the observation model. We performed experiments to demonstrate the validity of the proposed approach under various conditions with regard to dynamic objects, illumination, and viewpoint. The results clearly demonstrated the value of our approach.**

## I. INTRODUCTION

Most of the existing work on visual localization assumes that robot works within a static environment[1],[2],[3]. However, this assumption does not hold for many real-world environments. For example, the appearance of a location is associated with variations in the environment due to dynamic factors, such as people, or changes in illumination. In appearance-based robot localization, a new image is matched with every image in the database. Thus, it is important to reduce the number of images in this database with minimal loss of the ability to accomplish the task.

In the cases mentioned above, there are two significant problems for visual localization by a mobile robot. The first is that the robot must recognize different visual features caused by dynamic changes in the environment as being the same, and the other arises from the nature of the sparse appearance-based map. That is, a robot must be able to recognize observations taken at slightly different positions and angles within a certain region as identical. In this paper, we describe how to achieve localization using sparse appearance-based map with dynamic changes in environment.

Here, we present a possible solutions to these problems in the well known probabilistic framework called the Bayes filter[4]. In this framework, we propose an observation model based on $LeTO^2$(**l**ik**e**lihood of **t**he l**o**cation being **t**rue l**o**cation of the robot based on a visual similarity relative to a certain location) function for update step and an action-associated sparse appearance-based map for prediction, and final localization steps. The $LeTO^2$ function converts a matched result between two images to likelihood the two images are taken from same location, One of the main concept is that the $LeTO^2$ function considers the effects of dynamic factors and viewpoint changes to convert the matched result. Based on the action-associated sparse appearance-based map, an action-based view transition model is constructed. The transition model allows the combination of information over time. For more reliable localization, multiple visual features is fused in the update step and a final localization step is added to the process of the Bayes filter.

We represent environments implicitly as a database of features derived from a set of images in the framework of appearance-based localization[5],[6],[7],[8]. Here, images are collected at each location in a training phase, and scale-invariant feature transform(SIFT) descriptor[9] and homogenous texture descriptor (HTD)[10] are employed as multiple visual cues. For localization, the robot acquires an image at the current location from which features are then extracted. These features are compared with those stored in the database.

This paper is organized as follows. Section II explains the action-associated sparse appearance-based map and visual similarity used in this work. Section III describes the details of the proposed approach. Section IV presents some experimental results. Finally, Section V outlines our conclusions and perspectives for future work.

## II. Action-Associated Sparse Appearance-Based Map and Visual Similarity

This section provides a basis to understand the proposed approach and the experimental results presented later in the paper. The action-associated sparse appearance-based map and visual similarity used in this work are formally described below.

### *A. A. Action Associated Sparse Appearance-Based Map*

Our action-associated sparse appearance-based map is composed of several nodes, each of which consists of a set of views. We consider each node as a circle with a radius

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

Young-Bin Park is with the Division of Electrical and Computer Engineering, Hanyang University, 17 Haengdang-dong, Sungdong-gu, Seoul, 133-791, Korea pa9301@incorl.hanyang.ac.kr

Il Hong Suh is with the College of Information and Communications, Hanyang University, 17 Haengdang-dong, Sungdong-gu, Seoul, 133-791, Korea ihsuh@hanyang.ac.kr

Byung-Uk Choi is with the Division of Electrical and Computer Engineering, Hanyang University, 17 Haengdang-dong, Sungdong-gu, Seoul, 133-791, Korea buchoi@hanyang.ac.kr

978-1-4244-3804-4/09/$25.00 ©2009 IEEE

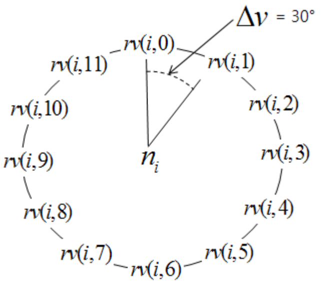

Fig. 1. Example of action-associated sparse appearance-based map, in which there is one node $n_i$ and each reference view is located at interval of 30° within the node.

of about 1m[1] and each view is the basic unit of location for recognition. At each view, only one image is captured by a robot in the sense of a sparse appearance-based map. Specifically, to build the proposed map, a robot is moved to to a manually selected node position, rotates through a certain number of degrees at the center of the node, and then acquires an image. This is repeated until the robot returns to the original orientation.

The proposed map is defined explicitly as follows:

- A node is denoted by $n_i, i \in \{1, ..., N\}$, where $N$ is the number of nodes in the map. The sum of prior probabilities of all nodes equals 1.
- A view is denoted by $v(i,j)$, where i and j are the indices of node and of view, respectively. A node is composed of a set of reference views: $rv(i,j) \in n_i$, $n_i = \{rv(i,0), rv(i,1), ..., rv(i,K)\}$. Here, $K = 360/\Delta v - 1$ and $\Delta v$ denotes the degree of rotation of the robot. The sum of prior probabilities of all views within node $n_i$ equals 1, and the sum of prior probabilities of all views within the map also equals 1.

Fig. 1 shows an example of an action-associated sparse appearance-based map. Note that the map is associated with robot action, and we represent the location as a view and not a node[11],[12]. Based on the map, the metric ranges of a location for localization are formally described as follows:

- Position and orientation of the robot are represented by pose $P = (X, Y, \theta)$. Let $P_{rv} = (X_{rv}, Y_{rv}, \theta_{rv})$ be the pose of the robot when the image was taken at the reference view $rv$ and let $P_v = (X_v, Y_v, \theta_v)$ be the pose of the robot at the actual view $v$, in the case where the robot is located at the view $rv$, which is formally defined as follows:

$$\begin{aligned} P_v \cong P_{rv},\ if\ \ & \theta_{rv} - 15^\circ < \theta_v \leq \theta_{rv} + 15^\circ\ and \\ & distance(P_v, P_{rv}) \leq 1m. \end{aligned} \tag{1}$$

- In the case where the robot is not located at view $v$, which is formally defined as

$$\begin{aligned} P_v \ncong P_{rv},\ if\ \ & \theta_{rv} - 15^\circ \geq \theta_v\ or \\ & \theta_{rv} + 15^\circ < \theta_v\ or \\ & distance(P_v, P_{rv}) > 1m. \end{aligned} \tag{2}$$

### B. Visual Similarity

The appearance-based localization used in this work departs from a set of training images $I = (I(i,0), ..., I(i,K))$ taken at locations $n_i = (v(i,0), ..., v(i,K))$ for all $i \in N$. A set of features $f(i,j) = (f(i,j)^1, ..., f(i,j)^C)$ is extracted from the image $I(i,j)$, for all $i \in N, j \in K$, where $C$ is the number of visual cues. We employ two types of visual cues: global and local features. As a global feature, we use HTD, whereas SIFT descriptor is employed as a local feature. Local feature such as SIFT descriptor is resistant to partial occlusion and is relatively insensitive to changes in viewpoint. On the other hand, global feature based on texture, such as HTD, shows better performance than SIFT under conditions of changing illumination[13].

In the map-building phase, the robot captures an image at each location from which it extracts SIFT descriptors and HTD and then stores the visual features in the database. In the localization phase, the features extracted from an image taken at the tobot's current location are matched with those extracted from each reference image in a pre-built database. In this paper, we define visual similarity as an output value from the match.

Formal description of the visual similarity between actual robot view $v$ and a reference view $rv(i,j)$ is as follows:

- A set of visual similarities relative to the *j-th* view in the *i-th* node is defined as

$$s(i,j) = \{s(i,j)^{SIFT}, s(i,j)^{HTD}\}. \tag{3}$$

- The SIFT descriptor-based visual similarity between actual robot view $v$ and the reference view $rv(i,j)$ is defined as

$$s(i,j)^{SIFT} = match_{\#}(f_v^{SIFT}, f(i,j)^{SIFT}), \tag{4}$$

where $f_v^{SIFT}$ is a set of SIFT descriptors extracted from an image taken at the robot's actual location and $match_{\#}$ denotes the number of matched SIFT keypoints.
- The HTD-based visual similarity relative to the reference view $rv(i,j)$ are defined as

$$s(i,j)^{HTD} = 1 - dist(f_v^{HTD}, f(i,j)^{HTD}), \tag{5}$$

where $f_v^{HTD}$ is HTD extracted from an image taken at the robot's actual location and $dist$ is the distance between two feature vectors. Note that to obtain HTD-base similarity, the difference between two feature vectors is previously calculated.

[1] If the size of a node is larger, variation in sensor data obtained within the node also becomes larger, decreasing the performance of localization.

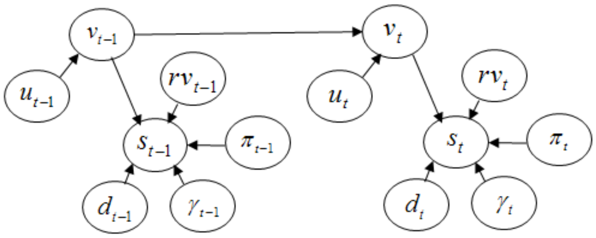

Fig. 2. The Proposed dynamic Bayesian model.

- The vector distance between two HTDs is defined as in [10]

$$dist(f_v^{HTD}, f(i,j)^{HTD}) = \Sigma_k \left| \frac{w(k)[f_v^{HTD}(k) - f(i,j)^{HTD}(k)]}{\alpha(k)} \right|, \quad (6)$$

where $f_v^{HTD}(k)$ and $f(i,j)^{HTD}(k)$ are the kth elements of two HTD feature vectors and $w(k)$ and $\alpha(k)$ indicate the weight and normalization factor for the kth element of the HTD feature vector, respectively.

## III. The Proposed Approach

This section describes the proposed localization approach using an action-associated sparse appearance-based map under a dynamically changing environment. The approach is constructed based on the Bayes filter, a well known probabilistic framework. The following section explains our proposed probabilistic framework in detail.

### A. *Overview of Proposed Probabilistic Framework*

Fig. 2 illustrates the dynamic Bayesian network that characterizes the effects of dynamic factors and viewpoint changes with respect to visual similarity. In addition, the DBN shows that visual similarity relative to a certain reference view is generated stochastically based on whether the robot locates at the reference view or not. In the DBN, random variables for the actual view of the robot, reference view, control data, visual similarity, dynamic factor and viewpoint change at time $t$ are denoted by $v_t$, $rv_t$, $u_t$, $s_t$, $d_t$ and $\gamma_t$, respectively and $\pi_t$ is a boolean random variable to present whether the robot is currently located at the reference view $rv_t$. The formal description of $\pi_t$ is as follows:

$$\pi_t = true, \quad if \quad P_{v_t} \cong P_{rv_t}$$
$$\pi_t = false, \quad if \quad P_{v_t} \ncong P_{rv_t}. \quad (7)$$

All random variables in the DBN can be partitioned into three sets: searched, known, and unknown variables[14]. In this case, a searched variable is the actual location of the robot $v_t$. Known variables are all past reference view $rv_{1:t}$, all past visual similarity $s_{1:t}$, and all past control data $u_{1:t}$. Unknown variables are all past dynamic factor $d_{1:t}$, all past viewpoint change $\gamma_{1:t}$, and actual location of the robot at time $t-1$ $v_{t-1}$, Note that variables $\pi_{1:t}$ are fixed as $true$ and used as evidence, since we estimate a probability that the robot locates at the reference view $rv$, given a certain reference view $rv$ and visual similarity between the reference view and actual robot view $v$. Then, the query is:

$$\begin{aligned} &P(v_t|s_{1:t},\pi_{1:t}\!=\!true,rv_{1:t},u_{1:t}) \\ &\quad = \eta P(s_t|v_t,s_{1:t-1},\pi_{1:t}\!=\!true,rv_{1:t},u_{1:t}) \\ &\qquad P(v_t|s_{1:t-1},\pi_{1:t}\!=\!true,rv_{1:t},u_{1:t}) \\ &\quad = \eta P(s_t|v_t,\pi_t\!=\!true,rv_t) \\ &\qquad P(v_t|s_{1:t-1},\pi_{1:t}\!=\!true,rv_{1:t},u_{1:t}) \end{aligned} \quad (8)$$

where the problem of the first row is reformulated to second and third rows using Bayes' rule and the final step uses conditional independence[15].

Our probabilistic framework consists of two steps: update, including the initial step, and prediction. For the update step, we propose an observation model based on the $LeTO^2$ function and the fusion of multiple visual similarities. For the prediction step, we construct an action-based view transition model based on the action-associated sparse appearance-based map. Specifically, the robot is localized in such a way that it initially has no prior knowledge about its location, and therefore the probabilities in all locations are uniform. The robot then acquires an image from which it extracts features, and it then obtains visual similarity by matching the features with those in the reference model. Our proposed $LeTO^2$ function converts each similarity to a likelihood of each location being the true location of the robot, considering dynamic factors in the environment and viewpoint changes within the location. For more reliable decisions, two resulting likelihoods based on SIFT descriptor and HTD are fused into a single value. After initial processing in the observation model, the robot rotates through the same number of degree as at the time of map building and then predicts the current location using the action-based view transition model. When a new observation is obtained, the process is repeated in the observation model, and then all of the probabilities in the prediction step are updated. This process is typically repeated until a certain hypothesis exceeds the threshold, but we do not determine the final location in this way. Instead, we provide a final localization step.

### B. *Prediction step*

The last row of (8) is manipulated into the form as follows:

$$\begin{aligned} &P(v_t|s_{1:t-1},\pi_{1:t}\!=\!true,rv_{1:t},u_{1:t}) \\ &\quad = \sum_{v_{t-1}} P(v_t,v_{t-1}|s_{1:t-1},\pi_{1:t}\!=\!true,rv_{1:t},u_{1:t}) \\ &\quad = \sum_{v_{t-1}} P(v_t|v_{t-1},s_{1:t-1},\pi_{1:t}\!=\!true,rv_{1:t},u_{1:t}) \\ &\qquad\qquad P(v_{t-1}|s_{1:t-1},\pi_{1:t}\!=\!true,rv_{1:t},u_{1:t}) \\ &\quad = \sum_{v_{t-1}} P(v_t|v_{t-1},u_t) \\ &\qquad\qquad\qquad P(v_{t-1}|s_{1:t-1},\pi_{1:t}\!=\!true,rv_{1:t},u_{1:t}) \end{aligned}$$

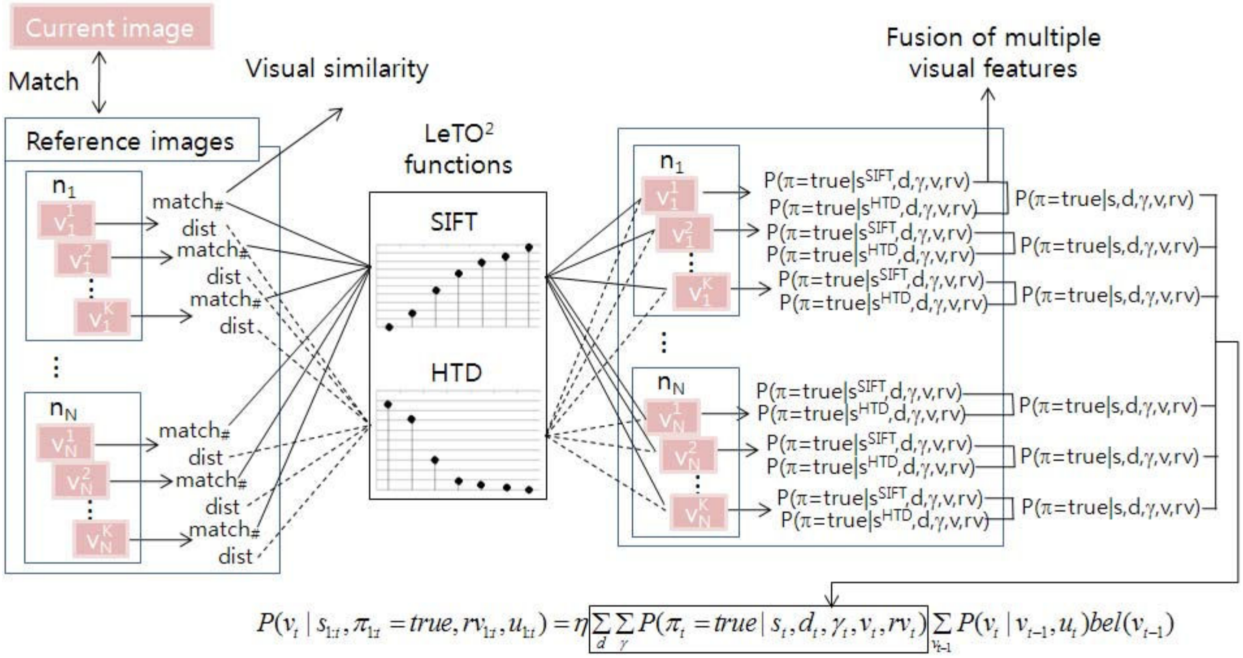
Current image
Match
Reference images
Visual similarity
n1
$v_1^1$
$v_1^2$
$v_1^K$
match#
dist
nN
$v_N^1$
$v_N^2$
$v_N^K$
LeTO² functions
SIFT
HTD
Fusion of multiple visual features
P(π=true|s^SIFT,d,γ,v,rv)
P(π=true|s^HTD,d,γ,v,rv)
P(π=true|s,d,γ,v,rv)
$P(v_t \mid s_{1:t}, \pi_{1:t} = true, rv_{1:t}, u_{1:t}) = \eta \boxed{\sum_d \sum_\gamma P(\pi_t = true \mid s_t, d_t, \gamma_t, v_t, rv_t)} \sum_{v_{t-1}} P(v_t \mid v_{t-1}, u_t) bel(v_{t-1})$

Fig. 3. The proposed observation model.

$$
\begin{aligned}
&= \sum_{v_{t-1}} P(v_t|v_{t-1}, u_t) \\
&\qquad P(v_{t-1}|s_{1:t-1}, \pi_{1:t-1} = true, rv_{1:t-1}, u_{1:t-1}) \\
&= \sum_{v_{t-1}} P(v_t|v_{t-1}, u_t) bel(v_{t-1}), \qquad (9)
\end{aligned}
$$

where the second row uses marginalization rule to take into account all possible values of $v_{t-1}$. Note that the eighth row of (9) is equivalent to posterior probability of the actual location of the robot at time $t-1$. This give us the recursive update equation.

It is important to exploit the opportunity to gain more information about the robot's location in the environment, because decisions based on only a single observation can lead to misclassification. However, if the robot cannot predict the next location, new evidence in the next location may not be important. In this case, all the information that can be used for localization is always obtained from only a single observation. In our framework, the opportunity to obtain more data is provided by the action-based view transition model, constructed based on the action-associated sparse appearance-based map. Whenever the robot rotates, it can predict where it will be located on the map by the transition model and additional evidence at the new orientation can be exploited to correct or increase confidence in the predicted belief.

Algorithm I outlines our action-based view transition model, in which $r_t$ denotes the actual degree of rotation and $u_t$ indicates the degree of rotation estimated by odometry. We assume that $u_t$ follows a Gaussian distribution and is given as

$$u_t \sim N(r_t, \sigma_t). \qquad (10)$$

In the algorithm, $bel(v_{t-1})$ and $\overline{bel}(v_t = v_j^i)$ represent posterior probability at time $t-1$ and predicted belief of $jth$ view in the $ith$ node at time $t$, respectively. $P(v_t = v_j^i | v_{t-1} = v_{j+K}^i, u_t)$ is the transition probability from the $j+Kth$ view to $jth$ view in the $ith$ node, given an estimated degree of rotation. If the $jth$ view is oriented farther from the addition of orientation of $j+Kth$ view and the estimated rotation degree, less transition probability is assigned. The algorithm eventually calculates the predictive belief for a certain view by integrating the product of estimated transition probabilities from all views in the node $n_i$ and the posterior probabilities at all views in node $n_i$ at time $t-1$,

Algorithm I Action-based View Transition Model

```
**view_transition_model**$(bel(v_{t-1}), u_t)$ :
    $\overline{bel}(v_t = v_j^i) \leftarrow 0$
    for $k \leftarrow 0$ to $K$
        $\beta \leftarrow k\Delta v - r_t$
        if $\beta > 180$ then
            $\beta \leftarrow 360 - \beta$
        $u_t(r_t + \beta) \leftarrow \frac{1}{\sqrt{2\pi\sigma}} \exp[-\frac{1}{2}(-\frac{\beta}{\sigma})]$
        $P(v_t = v_j^i | v_{t-1} = v_{j+K}^i, u_t) \leftarrow u_t(r_t + \beta)$
        $\overline{bel}(v_t = v_j^i) \leftarrow \overline{bel}(v_t = v_j^i) + bel(v_{t-1} = v_{j+K}^i)$
                                        $P(v_t = v_j^i | v_{t-1} = v_{j+K}^i, u_t)$
    end for
return $\overline{bel}(v_t = v_j^i)$
```

### C. Update step

The second step of the Bayes filter is called the measurement update, in which the predicted belief $\overline{bel}(v_t)$ is multiplied by the probability of sensor measurement to yield posterior probability. Based on the fourth row of (8), the

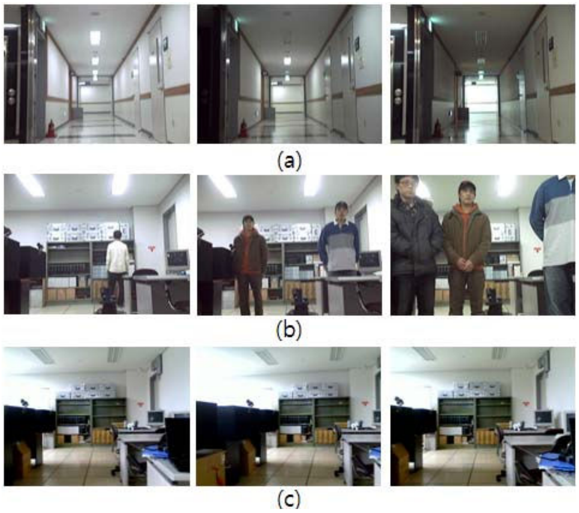

Fig. 4. (a) Various changes in illumination. (b) Variations of dynamic objects in environment. (c) Various changes in viewpoint within a view.

proposed update step is formally defined as follows:

$$
\begin{aligned}
&bel(v_t)\\
&= \eta P(s_t|v_t, \pi_t{=}true, rv_t)\overline{bel}(v_t)\\
&= \eta\sum_d\sum_\gamma P(s_t,d_t,\gamma_t|v_t, \pi_t{=}true, rv_t)\overline{bel}(v_t)\\
&= \eta\sum_d\sum_\gamma P(s_t|d_t,\gamma_t, v_t, \pi_t{=}true, rv_t)\\
&\qquad\qquad P(d_t,\gamma_t|v_t, \pi_t{=}true, rv_t)\overline{bel}(v_t)\\
&= \eta\sum_d\sum_\gamma P(s_t|d_t,\gamma_t, v_t,\pi_t{=}true, rv_t)P(d_t,\gamma_t)\overline{bel}(v_t)\\
&= \tilde{\eta}\sum_d\sum_\gamma P(s_t|d_t,\gamma_t, v_t, \pi_t{=}true, rv_t)\overline{bel}(v_t)\\
&= \tilde{\tilde{\eta}}\sum_d\sum_\gamma P(\pi_t{=}true|s_t, d_t,\gamma_t, v_t, rv_t)\\
&\qquad\qquad P(s_t|d_t,\gamma_t, v_t, rv_t)\overline{bel}(v_t)\\
&= \tilde{\tilde{\tilde{\eta}}}\sum_d\sum_\gamma P(\pi_t{=}true|s_t, d_t,\gamma_t, v_t, rv_t)\overline{bel}(v_t),
\end{aligned}
\tag{11}
$$

where it is noted that the third row uses marginalization rule to take into account all possible values of variation of dynamic factors and viewpoint. The probabilities of $P(d_t,\gamma_t)$ and $P(s_t|d_t,\gamma_t, v_t, rv_t)$ of row 6 and 9 can safely be omitted from (11) since we assume the probabilities are uniform. The last row shows multiplication of the observation model and predicted belief $\overline{bel}(v_t)$. One of the main contributions of this work is the observation model based on the proposed $LeTO^2$ function, considering environmental dynamic factors and viewpoint changes within a location.

### D. Construction of $LeTO^2$ Function

Fig. 3 illustrates observation model in our probabilistic framework, in which each visual similarity is converted into a probabilistic value based on the proposed $LeTO^2$ function. By the last row of (11), the $LeTO^2$ function must present the likelihood that the two images being taken at same location, given a visual similarity between the two images, with dynamic changes in environment and viewpoint changes within a view. For this purpose, we constructed the $LeTO^2$ function based on a number of experimental observations under various conditions. The experiment was conducted within one room and one corridor, in each of which we selected two nodes. At each node, we obtained a reference image at every 30° of rotation of a robot, and then the SIFT descriptors and HTD were extracted from each image. During image acquisition, there were no dynamic objects or variations in illumination. In this way, we constructed a reference model composed of 48 views, in which each view was considered a particular location. After generating the model, we gathered 400 test images at one of the locations of the reference model under the following varied conditions:

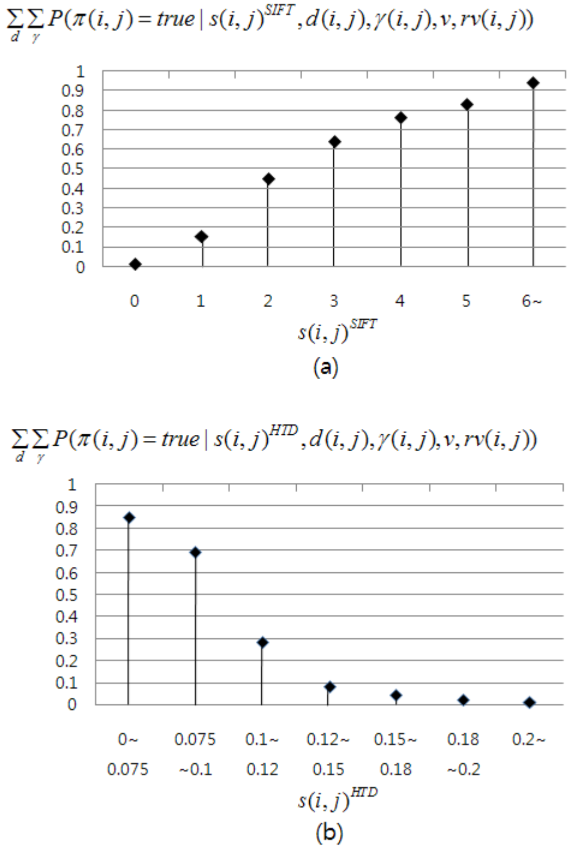

Fig. 5. (a) $LeTO^2$ function for SIFT descriptor-based similarity. (b) $LETO^2$ function for HTD-based similarity.

- The differences in mean gray level between test and reference images at same location were up to 50, at which level of illumination was varied between three or four different conditions by switching the ceiling light on or off and opening or closing the curtains on the windows.
- The percentage of regions of people within an image was changed from 0% to 50%, with a maximum of three people standing in front of the robot, with variation in the number of people and their position relative to the

Algorithm II Final Localization Filter

```
$\chi[1, ..., m]$ : array of $\chi$,
$m$ : number of items in array of $\chi$
r : number of rotation for localization
$\theta_p$ : threshold probability for candidate view
$\theta_c$ : threshold number of correct localization for final decision

localization($\chi, r, m$):
  $correct\_localization \leftarrow 0$
  if $\chi.p < \theta_p$ then
    if $r > K$ then
      return LOCALIZATION_FAILURE
    else then
      return REQUEST_ROTATION

  $\chi[m].n \leftarrow \chi.n$, $\chi[m].v \leftarrow \chi.v$
  $\chi[m].r \leftarrow \chi.r$, $m \leftarrow m + 1$

  for $k \leftarrow 0$ to $m - 1$
    if $\chi[m].n = \chi[k].n$ then
      $rot\_diff \leftarrow \chi[m].r - \chi[k].r$
      if $rot\_diff > K$ then
        $rot\_diff \leftarrow rot\_diff - K$
        if $\chi[m].v = \chi[k].v + rot\_diff$ then
          $correct\_localization \leftarrow correct\_localization + 1$
  end for

  if $correct\_localization \geq \theta_c$ then
    return LOCALIZATION_SUCCESS
  else then
    return REQUEST_ROTATION
```

robot.

- The pose of the robot was changed within a location.

Fig. 4 shows test images under various conditions as mentioned above, in which we attempted to marginalize all possible variations in dynamic factors and viewpoint changes. To construct the $LeTO^2$ function for SIFT descriptor, we calculated the ratio of total number of particular similarity according to each number of matched SIFT key points and the number of cases in which similarity was obtained from the same location. To construct the $LeTO^2$ function for HTD, each visual difference was mapped into one of seven quantization values, and then we calculated the ratio of the total number of a certain range of differences according to the quantization value and number of the certain range of differences obtained from the same location. The experimental results are shown in Fig. 5.

The probabilistic outputs from SIFT descriptors-based and HTD-based similarities were multiplied to obtain the final likelihood value as shown in Fig. 3. The fusion of multiple visual features is formally defined as

$$\sum_{d}\sum_{\gamma} P(\pi_t = true|s_t, d_t, \gamma_t, v_t, rv_t) = \eta \Pi_f \sum_{d}\sum_{\gamma} P(\pi_t = true|s_{f_t}, d_t, \gamma_t, v_t, rv_t) \tag{12}$$

where $f = \{SIFT, HTD\}$.

### *E. Final localization step*

To determine the reliable location of the robot, we provide an additional localization step called the final localization filter. The main concept underlying this process is to match the sequence of highest probability views with the sequence of the views in the action-associated sparse appearance-based map. This process is conducted in such a way that when the update step is finished, the view with the highest probability is selected, In addition, if the probability exceeds a certain threshold, the view is considered a candidate for final decision. The candidate view is considered a four-dimensional vector, which is formally defined as

$$\chi = < n, v, p, r >, \tag{13}$$

where $n$ and $v$ denote the indices of node and view, respectively, $p$ is the probability of the candidate view, and $r$ indicates the time of rotation. The vector is saved into an array of candidate views and never changed; then, the robot rotates and localizes again. Whenever a candidate view is added to the array of candidate views, the next process is conducted. In the next process, if the difference in the time of rotation between the current and previous candidate view is the same as the difference in the view index between the current and previous candidate views and the current and previous candidate view are located in the same node, localization is considered to have been executed correctly[2]. This process is repeated with respect to all further previous candidate views in the same manner. After the process is finished, if the number of correct localizations is above a certain threshold, the last candidate view is considered the final location of the robot; if not, the robot rotates again. the other hand, if the time of rotations is grater than the number of views in a node, localization fails. All of these processes are outlined in Algorithm II. In our experiment, $\theta_p$ and $\theta_c$ were 0.85, 3, respectively. These parameter were estimated to maximize recall and precision simultaneously, based on a number of experimental observations.

## IV. Experiment

We implemented our approach on a notebook PC (Intel. Pentium 2.0 GHz) and performed thorough testing using a Pioneer AT3 mobile robot equipped with a Logitech QuickCam Pro 4000. All the images were acquired at a resolution of 320x240 pixels using the web camera.

We placed special emphasis on three core ideas: (1)assigning probabilities to all the locations considering changes in environmental dynamics and viewpoint; (2)using transition probabilities based on the action-associated view transition model; and (3)fusing multiple visual features. We will show that integration of three core ideas described as above yields synergistic improvement of localization of a robot. For the sake of evaluation of our proposed approach, we implement three other localization approaches, each of which employs two of three core ideas and one different method. TABLE

[2] In this case, we assume that the robot rotates through the same angle as it did at the time of map-building.

TABLE I

ONE DIFFERENT METHOD OF EACH COMPARING APPROACH

| Approach | Different method | Proposed method |
|---|---|---|
| Approach I | Hard decision based on a certain threshold | Probability based on $LeTO^2$ function |
| Approach II | uniform transition probability | Transition probabilities based on view transition model |
| Approach III | Only HTD-based similarity | Fusion of multiple visual features |

I briefly explains how we implement the different methods with respect to three core ideas. In Approach I, if a matched similarity does not reach a certain threshold, a very low likelihood is assigned to the result; otherwise, a very high value is assigned. In Approach II, transition probabilities in the prediction step are uniform. We implemented Approach III using only HTD-based similarity but not SIFT descriptor-based similarity. As SIFT descriptor-based similarity is not discriminative in the case of usual illumination changes, most of the number of matched SIFT keypoints were zero.

The action-associated sparse appearance-based map was built in an office building. The map consisted of 15 nodes, of which six, six and three nodes were selected from two rooms, one corridor, and one hall, respectively. An image was obtained whenever the robot rotated through 30° in a node, and consequently each node was composed of 12 views.

First, We compared our approach and three other approaches at locations within the map, under various condition of dynamic factors, and various viewpoints as described in Section III-D. This experiment was conducted 100 times for each approach, and the experimental results are shown in TABLE II. The correct localization rates of Approach III and our approach were higher than those of the other approaches. Approach I uses a hard decision based on a certain threshold, but most of the similarities did not reach the threshold, due to various change in the conditions. Approach II decides its location using only a single observation, as the transition probabilities for all locations are always uniform.

Next, we compared Approach III and our approach, in which localization was conducted at a location that did not exist in the map. Various conditions were also changed. This experiment was performed 50 times for each approach. In this experiment, if the robot did not decide on one of the locations in the map until it reached the initial orientation, we considered the localization to have been conducted correctly. TABLE III compares the incorrect localization rate, given negative data. Approach III had poor performance, while our approach showed good performance. Approach III employed only HTD as visual cues, but in general texture-based features such as HTD can lead to much higher false-positive detection rates with respect to SIFT. In our approach, false positives caused by the matched result of HTD can be corrected by the matched results of SIFT descriptors. These experimental results indicated that all three of our methods are essential for reliable localization.

TABLE II

LOCALIZATION AT ONE OF THE LOCATION IN THE MAP

| Approach | Number of correct localization | Percentage |
|---|---|---|
| Approach I | 15 | 15% |
| Approach II | 24 | 24% |
| Approach III | 72 | 72% |
| Proposed approach | 86 | 86% |

TABLE III

LOCALIZATION AT UNKNOWN LOCATION

| Approach | Number of incorrect localization | Percentage |
|---|---|---|
| Approach III | 29 | 58% |
| Proposed approach | 4 | 8% |

## V. CONCLUSIONS AND REMARKS

We proposed a localization approach with an action-associated sparse appearance-based map, under dynamically changing environment. Our approach is based on the Bayes filter, a well known probabilistic framework. In this framework, we propose an observation model based on $LeTO^2$ function and an action-associated sparse appearance-based map for prediction, update and, final localization steps. In addition, multiple visual features are employed to provide greater reliability of the observation model.

Our probabilistic framework was evaluated experimentally, and was compared with three other approaches in Table 1. In an experiment with various changes in the environment and in viewpoint, the probabilities of correct and incorrect localizations were measured for each approach, and the results demonstrated the value of our approach.

In future work, we plan to enhance the mathematical aspects of the approach and experimentally confirm the generality of the proposed localization method. The quantization interval of the $LeTO^2$ function for HTD-based similarity is selected heuristically. Therefore, we will improve our framework by substituting the $LeTO^2$ function as a mathematically well defined classification technique. However, in this case, the classifier should provide probabilistic classification for use in the Bayes-filter framework. We consider the relevance vector machine(RVM)[16] to be such a classifier because it does not only achieves comparable recognition accuracy to the support vector machine(SVM) but also provides a full predictive distribution. In addition, we will apply our approach to many different indoor environments to confirm its generality.

## REFERENCES

[1] T. Goedeme, M. Nuttin, T. Tuytelaars, and L. Van Gool, "Omnidirectional Vision Based Topological Navigation," *International Journal of Computer Vision*, 74(3):219.236, 2007

[2] H.M. Gross, A. Koenig, C. Schroeter, and H.J. Boehme, "Omnivision-based probabilistic self-localization for a mobile shopping assistant continued," In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS03)*, 2003.

[3] I. Ulrich and I. Nourbakhsh, ”Appearance-based place recognition for topological localization,” In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA00)*, 2000.

[4] S. Thrun, W. Burgard and D. Fox, ”Probabilistic Robotics,” MIT Press, 2005.

[5] O. Booij, Z. Zivkovic and B. Krose, ”Sparse appearance based modeling for robot localization,” In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS06)*, 2006.

[6] B.J.A. Krose, N. Vlassis, R. Bunschoten and Y. Motomura, ”A Probabilistic Model for Appearancebased Robot Localization,” *Image and Ksion Computing*, 19(6), 381-391, April, 2001.

[7] A. Pronobis, B. Caputo and P, ”Confidence-based Cue Integration for Visual Place Recognition,” In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS07)*, 2007.

[8] H. Tamimi and A. Zell, ”Vision based localization of mobile robots using kernel approaches,” In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS04)*, 2004.

[9] Lowe, David G, ”Distinctive Image Features from Scale-Invariant Keypoints,” *International Journal of Computer Vision*, 60 (2): 91-110, 2004.

[10] Yong Man Ro, Munchurl Kim, Ho Kyung Kang, B.S. Manjunath, and Jinwoong Kim, ”MPEG-7 Homogeneous Texture Descriptor,” *ETRI Journal*, vol.23, no.2, pp.41-51, 2001.

[11] Tapus A., Tomatis N. and Siegwart R, ”Topological Global Localization and Mapping with Fingerprints and Uncertainty,” In *Proceedings of the International Symposium on Experimental Robotics*, Singapore, June, 2004.

[12] A. Rottmann, O. M. Mozos, C. Stachniss, and W. Burgard, ”Semantic place classification of indoor environments with mobile robots using boosting,” In *Proceedings of the National Conference on Artificial Intelligence American Association for Artificial Intelligence(AAAI05)*, Pittsburgh, PA, USA, 2005.

[13] A. Pronobis, ”Indoor place recognition using support vector machines,” Masters thesis, NADA/CVAP, KTH, 2005.

[14] O. Lebeltel, P. Bessiere, J. Diard, and E. Mazer, ”Bayesian Robot Programming,” *Autonomous Robots*, Volume 16, pp.49-79, 2004.

[15] S. Russell and P. Norvig, ”Artificial Intelligence: A Modern Approach, Second Edition,” Prentice-Hall, 2003 .

[16] Michael E. Tipping, ”Sparse Bayesian Learning and the Relevance Vector Machine,” *Journal of Machine Learning Research 1*, 211-244, 2001.