Advanced Robotics
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/tadr20

Visual Recognition of Types of Corridor Segments for Mobile Robots

Young-Bin Park [a] & Il Hong Suh [b]
[a] The Department of Electrical and Computer Engineering, Hanyang University, Seoul, 133-791, Korea
[b] The Division of Computer Sciences and Engineering, Hanyang University, Seoul, 133-791, Korea

Version of record first published: 19 Jul 2012.

To cite this article: Young-Bin Park & Il Hong Suh (2012): Visual Recognition of Types of Corridor Segments for Mobile Robots, Advanced Robotics, 26:16, 1915-1937

To link to this article: http://dx.doi.org/10.1080/01691864.2012.703169

Taylor & Francis
Taylor & Francis Group

Advanced Robotics 26 (2012) 1915–1937

ADVANCED ROBOTICS
tandfonline.com/tadr

Full paper

Visual Recognition of Types of Corridor Segments for Mobile Robots

Young-Bin Park[a] and Il Hong Suh[b]*

[a]The Department of Electrical and Computer Engineering, Hanyang University, Seoul 133-791, Korea;
[b]The Division of Computer Sciences and Engineering, Hanyang University, Seoul 133-791, Korea

Received 2 October 2011; revised 6 January 2012; accepted 1 March 2012

Abstract
This paper presents a visual recognition method to identify types of corridor segments such as T-junctions, L-junctions, and dead ends using vanishing point-based visual features and a two-layer recognition framework. This approach is useful for efficient robot navigation in the sense that a mobile robot is able to recognize the corridor segment type before reaching it, allowing the robot to make navigation decisions in advance. Furthermore, owing to our novel visual features made by using nonvertical vanishing points satisfying Manhattan world assumption, it is more probable for a mobile robot to recognize corridor segment types under partial occlusion by human. Experimental results have also been provided to demonstrate the validity of the proposed approach in real world environments.

Keywords
space recognition, corridor segment recognition, vanishing point, HMM, Manhattan world

1. Introduction

Most corridors contain few distinct features which would allow mobile robots to build their own maps, and because of this reason, the map constructed might not include sufficient information for the robot to correctly determine its own location. Various types of corridor segments, including T-junctions, L-junctions, and dead ends, allow the robot to localize itself more easily using information about the structure of its environment. In this paper, six types of fine-level corridor segments and two types of coarse-level corridor segments are defined, providing alternative landmarks for mobile robots.

*To whom correspondence should be addressed. E-mail: ihsuh@hanyang.ac.kr

DOI: 10.1080/01691864.2012.703169

There are a lot of researches to extract and recognize a landmark in indoor environment [1–7]. However, recognition of corridor segment types has been addressed by few studies to date [8–11]. Beranger et al. [10] considered a gray-scale 360° panoramic view at the center of an intersection and used multilayered neural networks to recognize three-branch intersections. Using this method, a mobile robot identifies an intersection type only at the center of the intersection as the captured panoramic view is used as input to the neural network. In practice, it is important not only to recognize the corridor segment type, but also to do recognition before the robot reaches the structural landmark to enable the robot to plan its navigation in advance. Boada et al. [8] developed recognition of distinctive features such as corridors, halls, narrowings and corridors with open doors in indoor environments using measurements from a laser scanner and hidden Markov models. Aycard et al. [9] defined five distinctive locations, including corridors, T-intersections, start of corridors, etc., and used ultrasonic sensors and second order hidden Markov models (HMMs) for recognition. However, data from ranged sensors such as laser range finders or ultrasonic sensors can pick up noise and other erroneous readings in real environments, especially as corridor segments are frequently occluded by people. Hui Liu et al. [11] developed a method to classify straight corridor segments and junctions based on image brightness values and a genetic algorithm. The experiments conducted within this study were performed in well-controlled environments, in which no human occlusion occurred.

In this study, a recognition system was developed using a two-layer stochastic framework and visual features instead of the range data used in previous works [8,9]. As a result, a mobile robot is able to recognize types of corridor segments, performing this recognition not only on arrival, but also during its approach. Furthermore, it is more probable for a mobile robot to recognize corridor segments under partial occlusion due to human. Specifically, the system exploits vanishing points to characterize the types of corridor segments we defined. Analysis of vanishing points provides strong cues for inferring information about the structure of an indoor scene [12–16].

In robotics, stochastic models are useful for solving decision-making problems such as action selection, simultaneous localization and mapping, sensor estimation, and place recognition [17–20]. This study considers the recognition of several types of corridor segments as a pattern classification problem [21], in which the coarse-level corridor segment type is determined using a simple Bayes classifier [22] and the fine-level corridor segment type is modeled using HMMs [23,8,25,26]. Within this proposed two-layer recognition framework, the identification of the coarse-level corridor types facilitates the fine-level corridor type recognition.

Main contributions of this work are three. Firstly, we recognize types of corridor segments by analyzing orientation of edges on the basis of vanishing points. Secondly, we propose three-dimensional feature vector which effectively

characterize corridor types. Lastly, we develop hierarchical probabilistic inference algorithm to deal with data including incorrectly classified edges.

2. Feature Extraction

This section will introduce the visual feature extraction process, which is based on the detection of vanishing points. The analysis of vanishing points provides strong cues, from which information about the three-dimensional structure of an indoor scene can be inferred. Coughlan and Yuille [12] proposed the existence of three mutually orthogonal vanishing directions in any given scene; this is known as the Manhattan world assumption. This approach to vanishing point detection was used in this study, since the Manhattan world assumption is reasonable in most man-made indoor environments, particularly in corridors.

2.1. Types of Corridor Segments and Three Cameras for a Mobile Robot

As shown in Fig. 1, the types of corridor segment are hierarchically categorized. The two coarse-level corridor segment types are I-corridor (IC) and nonI-corridor (NIC). Within the NIC category, there are six fine-level corridor segment types. These are: T-junction ahead (FT), left-facing T-junction (LT), right-facing T-junction (RT), left-facing L-junction (LL), right-facing L-junction (RL), and dead end (DE).

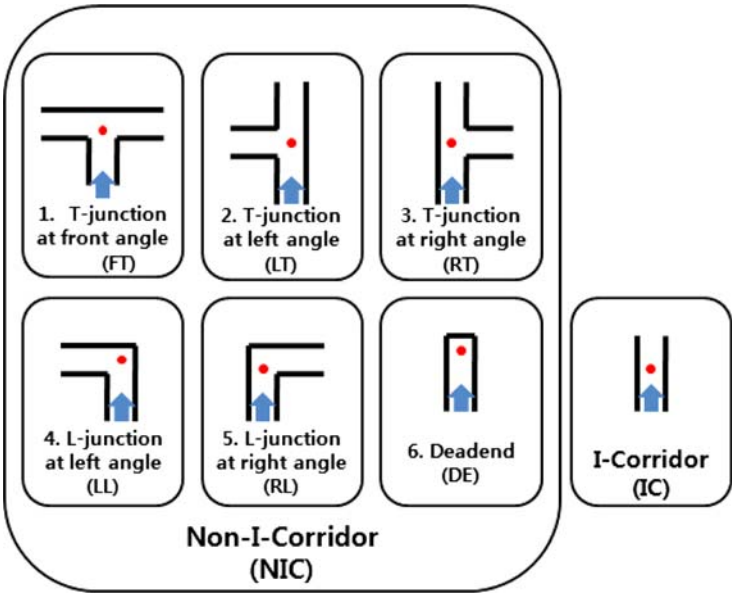

Figure 1. Types of corridor segments. The red circle represents the center of each corridor segment, with the arrow indicating the orientation of the robot.

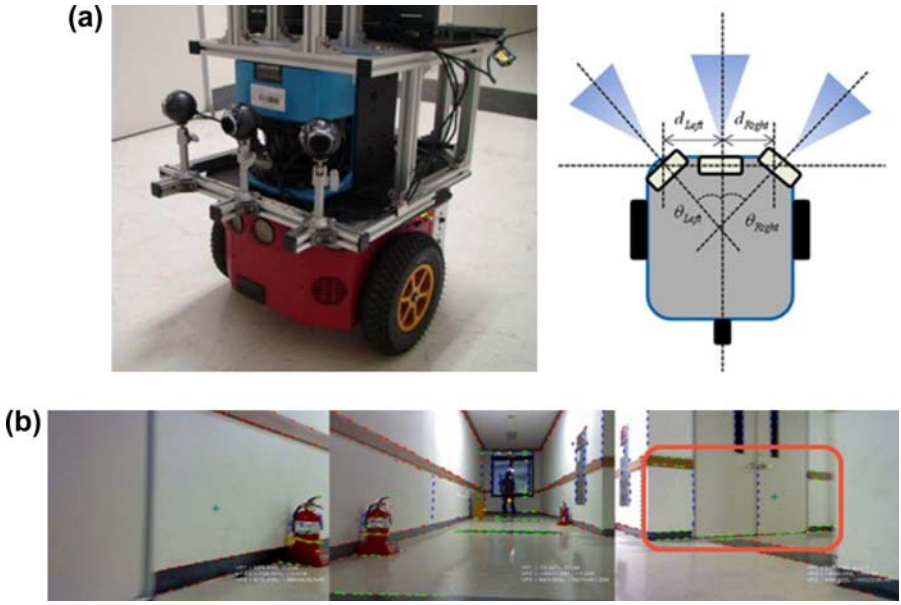

Figure 2. (a) Mobile robot with three front-mounted cameras. (b) Images captured using the three cameras.

Testing of this recognition method was performed with Pioneer 3-DX by MobileRobots Inc., which is most commonly used for mobile robot applications. Three cameras were mounted on the front of the robot, facing left-front, center, and right-front, respectively, as shown in Fig. 2(a). The heights of three cameras are 130 cm and relative directions of the front, left-front and right-front cameras are 0°, −30°, and 30°, respectively. The orientation of the camera was arranged to capture the appearance of a corridor segment better and earlier than if only the center camera was used. Figure 2(b) shows the three images captured by the cameras as the robot approaches an RT. The resolution of camera is 320×240.

2.2. *Assigning Sample Pixels to One of Three Vanishing Points*

According to the Manhattan world assumption, most captured image edges are oriented toward one of three vanishing points. Figure 3 shows how each sample pixel is assigned to either one of three vanishing point groups according to the orientation of the edge or is otherwise not assigned. In this study, sample pixels are determined by first obtaining the image edges using the Canny edge detector [27]. From this, sample pixels are selected at regular intervals along the edges. At most, one sample pixel is selected within each 15×15 region. To detect vanishing points, and assign each sample pixel to one of the three vanishing point groups simultaneously, a simplified version of the method used by Schindler and Dellaert [13] was developed. Schindler and Dellaert use a Bayesian approach and the expectation maximization (EM) algorithm to group edges and search for

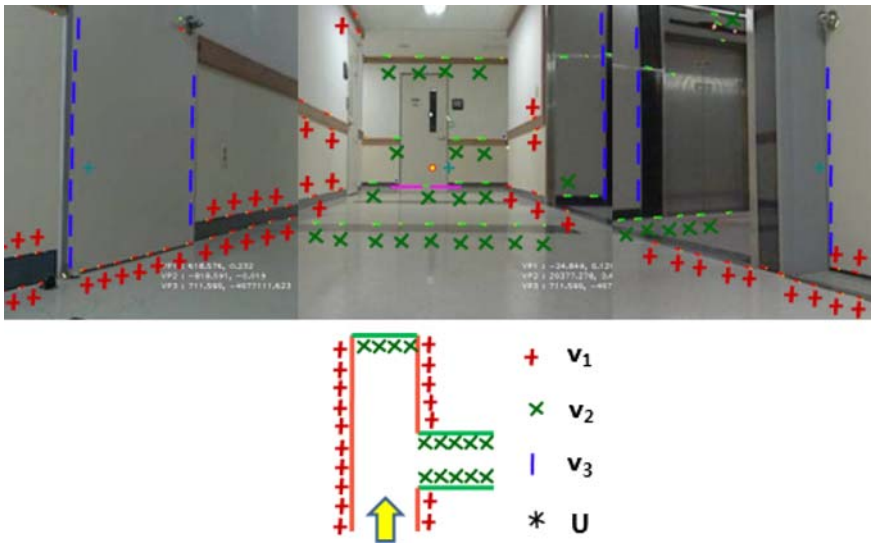

Figure 3. Each pixel is assigned to either one of three groups corresponding to three vanishing points, or placed in an 'unknown' group.

unknown parameters that influence the grouping of the edges. Main difference is that the existence of more than three mutually orthogonal vanishing directions is prohibited in our method in the sense of Manhattan world assumption and thus our approach can reduce computational cost.

Given a gradient image G, the system must estimate a set of unknown parameters Ψ that influence the location of the three vanishing points V. The parameters $\Psi = R$, where R presents the 3×3 rotation matrix of the 3D orientation of the cameras. The parameters Ψ and the vanishing points V have a functional relation $V = f(\Psi)$. Given this, the maximum a posteriori estimate Ψ^* must then be obtained for the parameters Ψ by maximizing the posteriori $P(\Psi|G)$,

$$\Psi^* = \underset{\Psi}{\operatorname{argmax}}\, P(\Psi|G) \propto P(G|f(\Psi))P(\Psi) \tag{1}$$

There are two assumptions in this work. Firstly, the robot should travel along the center of the corridor. It is because that orientations and positions of the left and the right cameras are fixed and the left and the right cameras should capture left and right sides of an intersection simultaneously, where most important features for discriminating intersection types can be observable. Secondly, one of two vanishing points associated with the xz-plane (i.e. plane parallel to floor) should be located in the center region of the middle image, except a case where the robot rotates. This assumption is required for calculating z_t^3 (see detail in Section 2.3). The nearest vanishing point from the center of each image is denoted by vp_1, the vanishing point orthogonal to vp_1 is denoted by vp_2, and the vanishing point at infinity in the vertical direction is depicted by vp_3.

The model at a sample pixel p is denoted as m_p. Each sample pixel is either assigned to one of the three vanishing points in the set V (i.e. $m_p \in \{vp_1, vp_2, vp_3\}$) according to the orientation of the edge at each pixel, or remain unassigned to V (i.e. $m_p \in \Phi$), in which Φ denotes a random edge. As shown in Fig. 3, we denote $v1$, $v2$, $v3$, and U as sets of sample pixels corresponding to vp_1, vp_2, vp_3, Φ, respectively. The posterior $P(\Psi|G)$ is then defined as the sum over all possible model configurations M,

$$P(\Psi|G) = P(\Psi) \sum_{M} P(G|M, f(\Psi))P(M) \tag{2}$$

where the prior $P(M)$ is assumed not to depend on Ψ. The summation over all model configurations M is intractable for all but small images. However, if conditional independence between g_p (the gradient measurement at the sample pixel p) given a model at the sample pixel m_p and the set of vanishing points V, Equation (2) becomes somewhat more tractable:

$$P(\Psi|G) = P(\Psi) \prod_{p} \sum_{m_p} P(g_p|m_p, f(\Psi))P(m_p) \tag{3}$$

To determine the parameters, EM is used, allowing efficient nonlinear optimization to be performed over Ψ .

An EM formulation leads to an algorithm where the following two steps are iterated until convergence:

- In the E-step, the current estimate Ψ^{old} for the parameters is fixed and a conditional posterior $P(M|G, \Psi^{old})$ is calculated over the model configuration M:

$$P(M|G, \Psi^{\text{old}}) \propto \prod_{p} P(g_p|m_p, f(\Psi^{\text{old}}))P(m_p) \tag{4}$$

A set of weights w_{pm} can then be obtained for each sample pixel p in the E-step:

$$w_{pm} \triangleq P(m_p|g_p, \Psi^{old}) \tag{5}$$

- In the M-step, the expected log-posterior $Q(\Psi; \Psi^{\text{old}})$ is maximized with respect to Ψ to obtain a new parameter Ψ^{new}. The reestimated parameters Ψ^{new} are calculated by minimizing the weighted least-squares error obtained by summing only over the cases $m_p = \{vp_1, vp_2, vp_3\}$. It is important to note $m_p = R$ does not depend on Ψ, since there is no predicted orientation in this case:

$$Q(\Psi;\Psi^{\text{old}}) \triangleq \sum_{p}\sum_{m_p\in V} w_{pm} log(P(g_p|m_p,f(\Psi))) + log(\Psi) \tag{6}$$

$$\Psi^{new} = \underset{\Psi}{\text{argmax}}\, Q(\Psi;\Psi^{old}) \tag{7}$$

The likelihood model $P(G|M,f(\Psi))$ is shared between the E-step and the M-step. Here, gradient measurements M are decomposed into gradient magnitude E and gradient orientation ϕ. Using this, the following likelihood model is obtained:

$$P(G|M,f(\Psi)) = \prod_{p} P(e_p|m_p)P(\phi_p|m_p,f(\Psi)), \tag{8}$$

where e_p is the gradient magnitude and ϕ_p denotes the gradient orientation. The likelihood of the gradient orientation $P(\phi_p|m_p,f(\Psi))$ models the distribution over orientation given the edge model m and the set of vanishing points V, and is defined as

$$\begin{cases} \mathcal{N}(\phi_p - \theta(vp_i,l_p),\sigma) & \text{if} \quad m_p = vp_1, vp_2, vp_3 \\ 1/2\pi & m_p = \Phi \end{cases}$$

where $l_p = [u,v,1]^T$ is the location of the sample pixel p in homogeneous coordinates. The predicted edge orientation $\theta(vp_i,l_p)$ is computed by taking the cross product.

$$vp_i \times u_p = [x,y,1]^T$$

From this, θ is computed as $arctan(y/x)$. A zero-mean Gaussian is performed on the difference between the measured and predicted edge orientations $(\phi - \theta)$, with no exploitation of the likelihood of gradient magnitude. The right term of Equation 6 can then be rewritten as:

$$\sum_{p}\sum_{m_p\in V} w_{pm}(\phi_p - \theta(vp_i,l_p))^2 + log(\Psi) \tag{9}$$

The algorithmic detail of detection of the three vanishing points and allocation of sample pixels to each vanishing point is described by Schindler and Dellaert [13].

2.3. Construction of the 3D Feature Vector Using Groups of Assigned Sample Pixels

When the robot moves through an IC, most of the sample pixels are assigned to v_1. When the robot approaches an NIC, the number of pixels assigned to v_2 increase. The pixels in v_3 are not used to identify the type of corridor segment.

We define three-dimensional feature vector at time t for the quantitative evaluation of structural characteristics.

$$Z_t = \{z_t^1, z_t^2, z_t^3\} \tag{10}$$

In the above equation, z_t^1 is defined as $n(v_{2_t}^l)/(n(v_{1_t}^l) + n(v_{2_t}^l))$, where n denotes the number of pixels in each group, the subscript t indicates the time t, and the superscript l denotes the left image. For example, $n(v_{1_t}^l)$ indicates the number of pixels in group v_1 of the left image at time t. Similarly, z_t^2 is identical to z_t^1 but z_t^2 extracted from the right image. Finally, z_t^3 indicates the normalized distance between the robot and the wall in front of the robot. We do not directly measure this distance instead, we measure distance from the vanishing point in the center region of the middle image and nearest horizontal line under the vanishing point. Here, we assume the line to be the boundary between the floor and the wall. We consider this measure as the distance from the robot and the wall in front of the robot. We refer the measure as "normalized distance" because we choose the longest distance in experiments as denominator. z_t^3 is formally defined as $(p_v - p_l)/d_{\max}$, in which p_v is y position of vanishing point, p_l denotes y position of the horizontal line, and $d_{\max}$ is the longest distance. As Fig. 4(a) and (b) show, when the robot is far from the wall, the distance between the vanishing point and the horizontal line is short, but when the robot is near the wall, the distance between the vanishing point and the horizontal line is relatively long. This distance increases as the robot approaches the wall.

Figure 5 shows four sequences of three-dimensional vectors as the robot moves from IC to FT, DE, LT, and LL. The x-axis indicates the index of the frame and the y-axis shows values from 0 to 1. In the case of FT, the values of z_1 and z_2 increase from 0 to 1. As the robot moves through IC, the majority of sample pixels from the right and left images are assigned to v_1. As the robot approaches FT, the number of sample pixels assigned to v_2 increases, and the z_3 values increase from near 0 to near 1. This is because the distance between the robot and the wall is very far while the robot is in IC, which decrease as the robot reaches the end of FT. In the case of DE, the sequence of three-dimensional vectors is similar to the sequence found in FT, except the maximum values of z_1 and z_2 are smaller than the maximum values in FT case. The values of z_1 during the LT and LL cases are similar, increasing to over .8. Conversely, the maximum values of z_2 and z_3 during the LT case are smaller than those found during an LL. This is because the structural appearance of the left sides

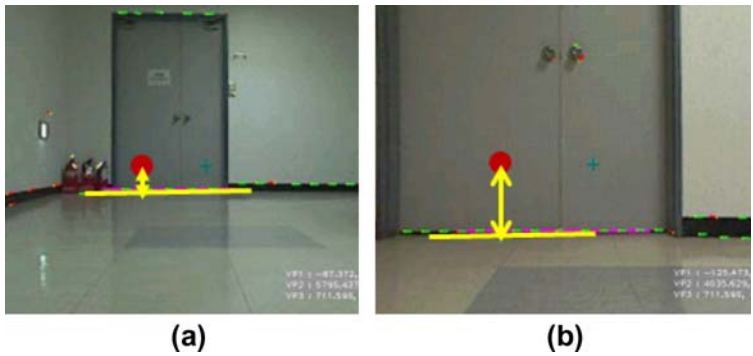

Figure 4. Normalized distance between the robot and the wall it is facing is calculated by considering the distance between the vanishing point in the center region of the middle image and the nearest horizontal line beneath.

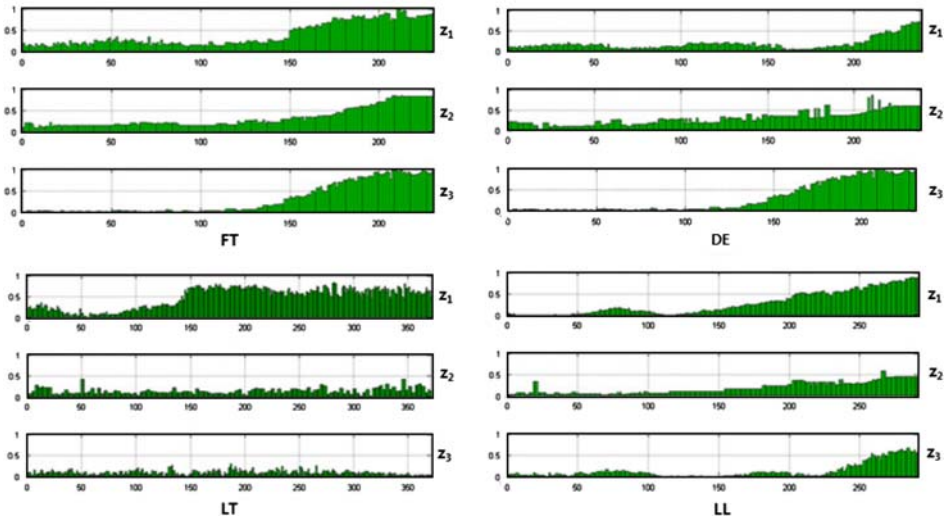

Figure 5. Four sequences of three-dimensional feature vectors as the robot moves from IC to FT, DE, LT, and LL.

view of an LT and LR are similar, whereas the structural appearance of the right and front sides differ.

3. Recognition of Types of Corridor Segment

IC and NIC, defined as two types of coarse-level corridor segments, are classified using a simple Bayes classifier. The identification of corridor type at a coarse-level facilitates the recognition of six fine-level corridor types, by trigger-

ing the following fine-level stochastic procedure. Here, the characteristics of a type of corridor segment at fine-level are modeled using a HMM.

3.1. Observation

As described in the previous section, the extracted feature vector from the three images at time t is a value in three-dimensional continuous space. This vector cannot be used directly as the observation for HMM or Bayes classifier methods due to its huge possible feature space. To achieve a better estimation for the observation likelihood functions, 10 sequences of feature vectors are collected for each corridor segment during the training phase. For each sequence, range of the number of frames is from 220 to 390 and traveling distance is in the range 15–30 m. The feature vectors are then all clustered using the k-means algorithm. k is the number of clusters and it should be determined before performing k-means clustering. k is determined based on experimental evaluation. Specifically, the value of k is increased from 1 to a sufficiently large value k and k maximizing the recognition performance is then chosen. We set $k = 13$. The cluster indices are exploited as observations. The observation likelihoods for the two coarse-level and six fine-level corridor types are obtained by performing this k-means clustering on the three-dimensional feature vector during the learning phase. Figure 6 shows the example of the observation likelihoods of IC and NIC, where k is 13. During the recognition phase, an observation at time t is formally defined as follows:

$$o_t = \underset{i=1,\ldots,k}{\operatorname{argmin}}\, d(\mu_i, Z_t), \tag{11}$$

where μ_i is the mean of the $i-th$ cluster, d denotes the Euclidean distance function, and k is the number of clusters.

3.2. Overview

To better understand the two-layer recognition procedure, the learning and recognition algorithms are described in the following subsections.

3.2.1. Learning Algorithm

Before the recognition phase is executed, it is necessary for the system to learn the characteristics of each type of corridor segment in terms of the model parameters such as initial distribution, state transition probability distribution, and observation probability distribution. The learning phase consists of five steps summarized as follows:

- 1. Select a type of a corridor segment to learn and navigate the segment.
- 2. Detect three vanishing points from each of the three cameras.

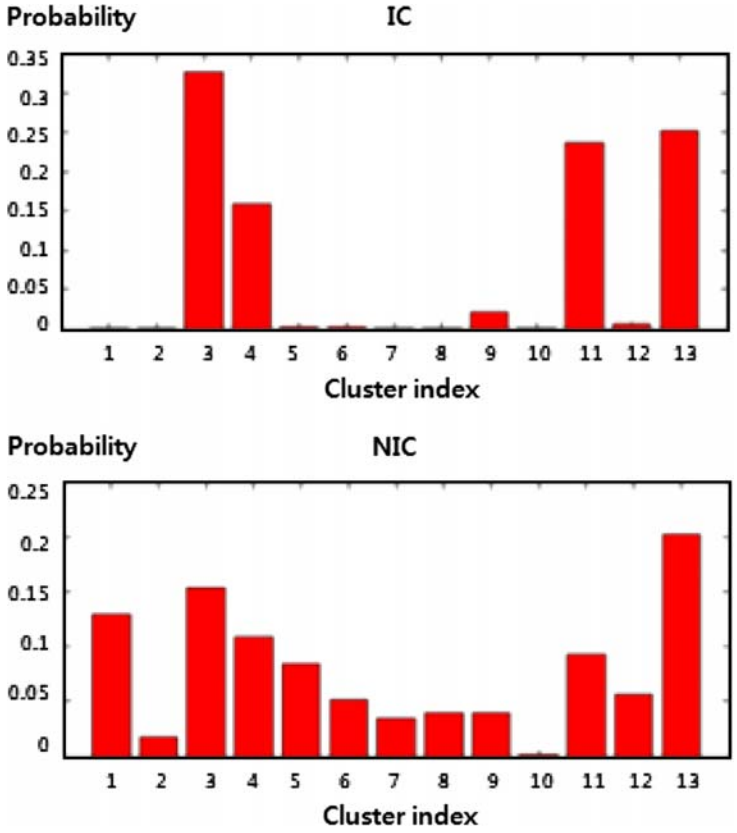

Figure 6. Observation likelihoods for IC and NIC cases, the x-axis shows the observation symbol index and y-axis represents the probability of each observation occurring.

- 3. Extract three-dimensional visual features based on the detected vanishing points.
- 4. Transform the feature vector into the cluster index using Equation 11.
- 5-1. Return to Step 2 and repeat until the robot reaches the corridor segment.
- 5-2. When the robot reaches the corridor segment, return to Step 1 and select another corridor segment of the same type and repeat until sufficient observations have been obtained to learn the corridor type.
- 5-3a. For fine-level learning, when sufficient observations have been obtained, estimate the parameters using the Baum–Welch algorithm and terminate the learning process.
- 5-3b. For coarse-level learning, when sufficient observations have been obtained, generate the observation likelihood function and terminate the learning.

This learning procedure must be repeated for all corridor segment types.

3.2.2. Recognition Algorithm

This procedure consists of six main steps summarized as follows:

- 1. Select a type of corridor segment for the robot to recognize and navigate the corridor segment.
- 2. Detect three vanishing points from each of the three cameras.
- 3. Extract three-dimensional visual features based on the detected vanishing points.
- 4. Transform the feature vector into the cluster index using Equation 11.
- 5. Perform recognition of the coarse-level corridor segment type based on the Bayes decision rule.
- 6-1. If the result of the recognition is IC, return to Step 2.
- 6-2. If the result of the recognition is NIC, perform an HMM-based inference at the fine-level to identify the corridor type at a more detailed level.

3.3. Two Classification Methods

A Bayes classifier is a simple probabilistic classifier based on the application of Bayes' theorem using naive independence assumptions. HMM is a robust approach for handling complex temporal signals and is a useful technique for classification based on learned model parameters and a sequence of temporal observations. Because of these advantages, the Bayes classifier and HMM are suited to both the simple classification task of recognizing coarse-level corridor types, and the relatively complicated problem of classifying fine-level corridor types, respectively.

3.3.1. Bayes Classifier for Coarse-Level Recognition

Bayes' formula shows that by observing the value of o, the prior probability $P(\omega_j)$ can be converted into a posteriori probability $P(\omega_j|o)$. The posterior represents the probability of the state ω_j, given an observation value o. In the case of two-category classification, a simple Bayes decision rule can be considered:

$$\text{Decide} \quad \omega_1 \quad \text{if} \quad P(\omega_1|o) \geq P(\omega_2|o) \tag{12}$$

This form of the decision rule emphasizes the role of posterior probabilities. Using this rule, the posterior probabilities can instead be expressed in terms of the likelihood and prior probabilities:

$$\text{Decide} \quad \omega_1 \quad \text{if} \quad P(o|\omega_1)P(\omega_1) \geq P(o|\omega_2)P(\omega_2) \tag{13}$$

Some additional in sight can be obtained by considering a few special cases such as $P(\omega_1) = P(\omega_2)$. In this case, the decision is based entirely on the likelihood

$P(o|\omega_1)$. In this study, $P(\text{IC})$ and $P(\text{NIC})$ are assumed to be equal, therefore coarse-level corridor segments are decided using the likelihood values for each corridor type. Both observation likelihood functions for IC and NIC are learned using training data.

To classify the coarse-level corridor type through comparison of likelihood values, the joint density function for all observations must first be specified. In this case, each observation was assumed to be conditionally independent of each other given the observation likelihood function θ. Therefore, the joint density function can be expressed as:

$$f(o_1, ..., o_t|\theta) = f(o_1|\theta) \cdot f(o_2|\theta) \cdots f(o_t|\theta) = \prod_{i=1}^{t} f(o_i|\theta). \tag{14}$$

3.3.2. *Six HMMs for Fine-Level Recognition*

An HMM is characterized the following variables:

- N: Number of states in the model.
- M: Number of observation symbols.
- The initial distribution π_i
- The state transition probability distribution $A = \{a_{ij}\}$
- The observation probability distribution

$$B = \{b_j(o)\}$$

- Observation sequence $O = o_1, o_2, ..., o_t$

For each type of coarse-level corridor segment, a HMM is generated, representing a particular characteristic of the place. The chosen model has two states ($N = 2$) and three transitions, as shown in Fig. 7. Thirteen observation symbols ($M = 13$) are used as mentioned before. x_{inter} indicates the intermediate section, in which the characteristics of a type of corridor segment begin to appear, and $x_{\text{dist.}}$

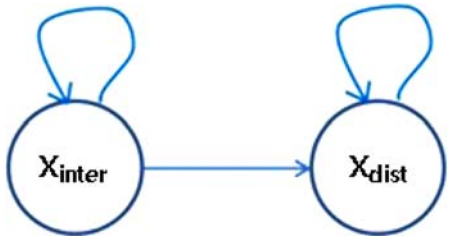

Figure 7. HMM model for a corridor segment.

represents the distinctive section, in which the visual pattern of one type of corridor segment is distinguishable. In the case of a fine-level corridor type, there are two observation likelihoods for $x_{\text{inter.}}$ and $x_{\text{dist.}}$ states.

And there are three problems associated with HMM:

- Evaluation: Finding the probability of a set of observations occurring given a particular HMM. These probabilities can be used to calculate a score via the forward–backward algorithm to show how applicable the given model is to the sequence.
- Decoding: The determination of the optimal order of hidden states to generate the observed sequence. The Viterbi algorithm is commonly employed to solve this problem.
- Learning: The estimation of the model parameters such as π_i, $A = \{a_{ij}\}$, and $B = \{b_j(o)\}$. There exist no particular algorithm to solve this optimally, however most common methodology is to adapt the Baum–Welch.

The coarse-level recognition procedure is carried out using the forward–backward algorithm. As a result, a good recognition result entails finding the model with the highest probability given a sequence of observations.

3.4. *Two-layer Recognition Framework*

Within this study, a two-layer recognition framework was proposed. According to this framework, the robot is only required to evaluate two coarse-level corridor types in the upper layer by means of a simple Bayes classifier. The upper-layer inference prevents the robot from wasting computational resources at every measurement otherwise, the proposed system estimates probabilities of all fine-level corridor type at all frames. The six fine-level corridor types are only considered when the robot strongly believes that an NIC condition is approaching. In addition, observations relevant to fine-level corridor types can be segmented in this framework from observations relevant to IC.

The probabilities of a corridor segment being IC or NIC at time t can be defined as $P(C_t^u = \text{IC}) \equiv f(O^{t-n:t}|\theta_{\text{IC}})$ and $P(C_t^u = \text{NIC}) \equiv f(O^{t-n:t}|\theta_{\text{NIC}})$, respectively. Here, the superscript u represents the upper-layer, θ_{IC} and θ_{NIC} indicate the learned observation likelihood functions, and O represents a sequence of observations. It is noted that only observations ranging from time $t - n$ to time t are considered, instead of total observations from time 0, for the following reason: if all observations were used as evidence for the Bayes classifier, travel through a long IC before reaching an NIC, could cause $P(C_t^u = \text{IC})$ to be higher than $P(C_t^u = \text{NIC})$ due to the accumulation of observations. This would cause the fine-level recognition to be omitted, as $P(C_t^u = \text{IC}) \leq P(C_t^u = \text{NIC})$ to trigger HMM-based recognition. Thus, a moving time window is employed to ensure only recent observations are considered.

For the same reason as in coarse-level classification, the robot does not need to use all previous observations as inputs for HMM-based estimation. A sequence of observations must be truncated to use only the signals associated with the current type of corridor segment. To do this, a sequence of observations taken after the trigger point are used as inputs for the HMM-based inference.

The probability of the current position being a corridor type l_i at time t can be defined as $P(C_t^l = l_i) \equiv P(O^{t-\widetilde{n}:t}|\theta_{l_i})$, where θ indicates the HMM parameters and superscript l represents the lower-layer. Similar to the upper-layer, only observations ranging from time $t - \widetilde{n}$ to time t are considered. $t - \widetilde{n}$ represents the start time of the moving window, in which $\widetilde{n}$ indicates the trigger time. The recognition procedure at fine-level is formally defined as:

$$C_t^l = \underset{l_i=l_1,\ldots,l_6}{\operatorname{argmax}} \quad P(O^{t-\widetilde{n}:t}|\theta_{l_i}), \tag{15}$$

where, as mentioned in the previous section, $P(O^{t-\widetilde{n}:t}|\theta_{l_i})$ is estimated using the forward–backward algorithm to solve the evaluation problem. The numbers of frames used for upper-layer and lower-layer classifications are different. n is 15 and $\widetilde{n}$ is 10. The numbers are empirically determined for obtaining best performance.

4. Experimental results

Experiments were performed in four buildings at the Hanyang University. In each experiment, the robot moved forward in an IC condition and then approached one of six corridor types until the wall or end of the corridor type was reached. This experiment was repeated approximately 15 times for each corridor type. For each sequence, range of the number of frames is from 200 to 400 and traveling distance is in the range 15–30 m. Moving speed of the robot is 25 cm/sec and width of corridors is in the range 250–350 cm. All of the experiments are performed by using the computer with an Intel(R) Core(TM)2 2.0 GHz CPU and 1.0 GB RAM. d_{left}, d_{right}, θ_{left}, and θ_{right} in Fig. 2(a) are set to 20, 20 cm, $-30°$, and 30°, respectively.

As mentioned in Section 2, there are two assumptions in this work. First, the robot should travel along center of the corridor. However, this assumption is not strongly required to our experiments. Specifically, we experimentally observed that the robot could be allowed to travel along near center of the corridor (e.g. 50 cm on the left and right sides from the center), and moreover, larger margin could be allowed if it happens occasionally through an image sequence. We believe that such a near center navigation can be easily achieved by employing usual sonar-based strategy that follows the center of the corridor. Such flexibility can be arisen owing to two reasons. First, observation symbols used in both

HMM and simple Bayes classifier are generated by clustering our proposed visual features. Thus, different visual features having a certain level of variation can be grouped into same observation symbol. In other words, visual features obtained at exact middle or near middle of corridor might be considered as same observations by proposed model. Second, a corridor type is not recognized by a static image, but a set of sequential images based on temporal probabilistic inference model. Thus, some frames obtained far from the middle of corridor do not have significant effect on classification if most frames are captured at near middle of the corridor. Second assumption is that one of two vanishing points associated with the xz-plane (i.e. plane parallel to floor) should be located in the center region of the middle image, except a case where the robot rotates. We also believe that such navigation can be also achieved without practical difficulty by following vanishing point.

The performance of this method was evaluated in two ways. Firstly, the region of the ground truth for NIC was defined as rectangles as shown in Fig. 8. Following this, the recognition results obtained as the robot traversed from the beginning to end of the region of the ground truth (approximately 2-frame/sec) were compared to the ground truth. Secondly, the ground truth region was extended to the dotted lines shown in Fig. 8. Again, recognition results obtained as the robot traversed the extended ground truth rectangle were compared with the extended ground truth. This second experiment was conducted primarily to evaluate the performance of the predictive recognition characteristic of the system. The position of the dotted line was determined manually through analysis of the left and right images. When the number of sample pixels assigned to v_2, shown in rounded rectangles in Fig. 9, exceed the threshold $T_n = 5$, the position

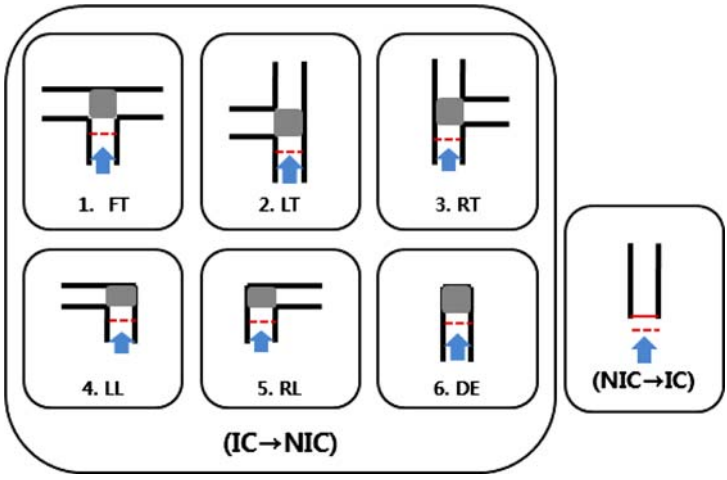

Figure 8. The gray rectangle denotes the region of ground truth for NIC. The extended region of ground truth for NIC expands to the dotted line.

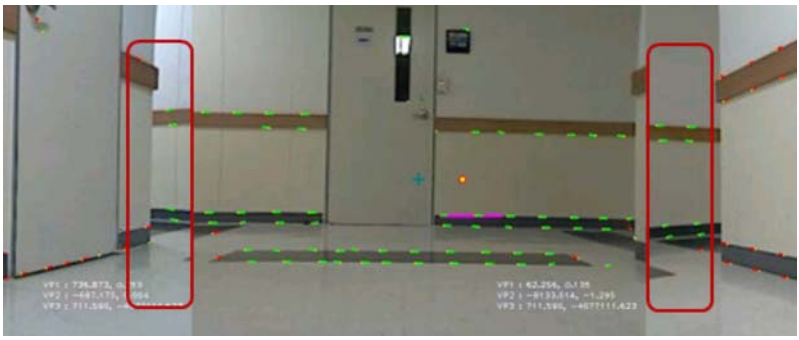

Figure 9. The location of each dotted line shown in Fig. 8 is determined when the number of sample pixels assigned to v2 in the left or right image, (as shown within the rounded rectangles above), exceeds the threshold value Tn.

Table 1.
Ratios of correct and incorrect recognitions for each corridor type

	Ground Truth					
	FT	LT	RT	LL	RL	DE
FT	**.9206**	–	–	.0032	.0096	.0741
LT	–	**.9957**	–	.1317	–	–
RT	–	–	**.9776**	–	–	–
LL	–	.0043	–	**.8369**	–	–
RL	–	–	.0224	–	**.9904**	–
DE	.0794	–	–	.0282	–	**.9259**

of the robot is determined to be the position of the dotted line. These lines were usually located 2 or 3 m ahead of each original rectangle.

The results of the first performance evaluation are shown in Table 1, in which the ground truths are arranged in columns and the recognition results are shown in rows. The ratio of correct and incorrect recognitions for each corridor type is shown in Table 1. It is observable that all corridor types except LL were correctly recognized over 90%. In several LL cases, straight lines were present on the floor. These straight lines were oriented in the same direction as the edges of the pixels in v_1 increasing the potential for confusion between LL and LT. Table 2 gives the ratios of correct and incorrect recognitions for IC and NIC.

Tables 3 shows the performance of predictive recognition for fine-level corridor types. The results of predictive recognition were mostly accurate, except for in the case of FT. These were misidentified as DE several times since, as shown in Fig. 10(a) and (b), the extracted features from the dotted line to the start line of the rectangle in an FT are very similar to those found in an DE (recall

Table 2.
Ratios of correct and incorrect recognitions for IC and NIC

	Ground Truth	
	IC	NIC
IC	**.9871**	**.1379**
NIC	**.0129**	**.8621**

Table 3.
Ratios of correct and incorrect recognitions for corridor type

	Ground Truth					
	FT	LT	RT	LL	RL	DE
FT	**.5333**	–	–	.0021	.0062	.0687
LT	–	**.9657**	–	.1833	–	–
RT	–	–	**.9643**	–	**.0613**	–
LL	–	**.0181**	–	**.7505**	–	–
RL	**.1373**	**.0144**	**.0279**	**.0172**	**.8988**	–
DE	**.3294**	**.0018**	**.0077**	**.0469**	**.0337**	**.9313**

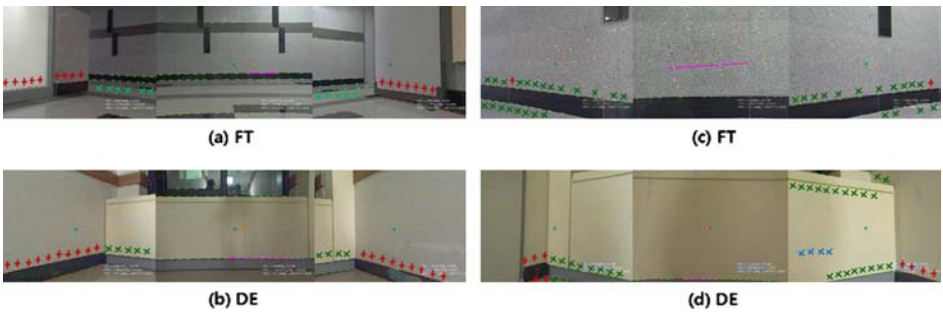

Figure 10. (a) An image captured from the dotted line to FT rectangle start line. (b) An image captured from the dotted line to the DE rectangle start line. (c) An image captured after entering the FT rectangle start line. (d) An image captured after entering the DE rectangle start line (recall Fig. 8).

Fig. 8). The differences between the FT and DE cases begin to manifest after the robot entered the ground truth rectangle, as shown in Fig. 10(c) and (d). However, the system does not misidentify DE as FT as shown in Table 3. This is because the three-dimensional feature vectors obtained from an FT have a larger variation than those taken from an DE as shown in Fig. 5. This means the

number of observation types for an FT is greater than those for an DE. Furthermore, the probabilities of each observation type within the FT observation likelihood function are relatively small compared to those found for DE. Therefore, when an observation both in FT and DE is given, the system is likely to select DE.

We note that sequences used in our experiments include frames having partial occlusion mostly caused by human. Ratio of such frames is about 10% of all frames. When a corridor is partially occluded, performance of the recognition depends heavily on the size of the occluded region. For example, if the size of the occluded region is small, recognition accuracy might be still good and otherwise, recognition accuracy might become bad. The performance evaluation is thus valuable only when the size of the occluded region is considered. However, it is extremely hard to evaluate performance according to different size of occluded region.

Instead, it is worth qualitatively observing why the proposed system has been relatively a little affected by partial occlusion due to human. First, three-dimensional feature vector is not significantly corrupted by such occlusion, because most edges obtained from human are not assigned to v_1 or v_2 that contribute to build up three-dimensional feature vector. In other words, orientations of edges coming from human are almost random and usually do not correspond to any of vanishing directions in xz-plane (i.e nonvertical vanishing directions). Second, proposed method employs HMM, a temporal probabilistic inference model. This implies that a corridor type is not recognized by a static image but a set of sequential images. In this case, even though some of the feature vectors obtained from the sequence are distorted by human occlusion, it is highly probable to recognize corridor type correctly.

Figure 11 shows both an LT and LL, in which the corridor segments are partially occluded by humans. Several sample pixels were extracted from the edges of these humans, however most were assigned to U which represents the unknown set as described previously. Table 4 shows the distribution of pixels in

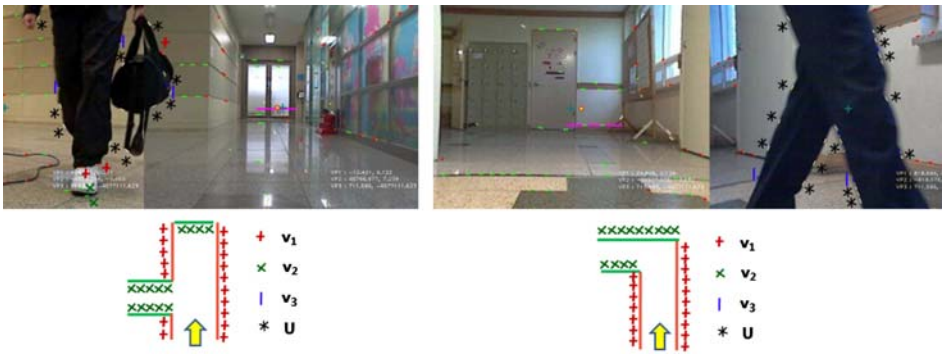

Figure 11. (a) A human and bag occlude an LT. (b) A human occludes an LL.

Table 4.
Ratios of the numbers of pixels in v_1, v_2, v_3 and U

v_1	v_2	v_3	U
.1533	**.1824**	**.1561**	**.508**

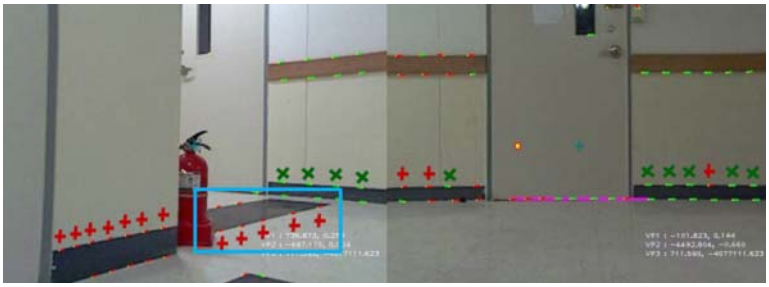

Figure 12. Left and middle images, which are captured as the robot approaches LL.

v_1, v_2, v_3, and the unknown set U, which contain the pixels extracted from the occluding objects. This table was obtained by using all experimental images including human occlusion. It is important to note that the sum of v_3 and U is .6641. In this case, partial occlusion of a corridor segment is not crucial for recognition performance, since the v_3 set and the unknown set are not used to extract the three-dimensional feature vector.

Figure 12 shows limitation of our method. Arbitrary patterns on the floor can cause misclassification as if the patterns are accidentally oriented toward one of vanishing directions in xz-plane. For example, edges in blue rectangle are assigned to v_1. This can cause misidentification, i.e. the robot recognizes LL as LT. Similarly, an open door also can be recognized as a junction in some cases where edges obtained from the open door are accidentally oriented toward one of vanishing directions in xz-plane.

5. Conclusions and Future Works

In this paper, a visual recognition method was presented to identify several types of corridor segments. By detecting vanishing points in three images and extracting three-dimensional features based on the vanishing points, the type of corridor segment can be recognized by a mobile robot before it reaches the center of the corridor. This allows efficient robot navigation since the robot is able to make decisions in advance about its next navigable path. Furthermore, it was shown by experiments that a mobile robot would recognize the corridor segment types

under partial occlusion due to human. It is remarked that laser range finder-based approaches may suffer from erroneous readings in such occluded situation.

In proposed method, a two-layer recognition framework is used to enable the recognition to be performed efficiently. Here, a simple Bayes classifier identifies coarse-level corridor types such as IC and NIC, and then use the result of this recognition to trigger the lower-layer inference procedure. In this fine-level recognition, HMMs-based estimation classifies six types of corridor segments including T-junctions, L-junctions, and dead end.

Through experimentation, the recognition and predictive recognition results of the proposed method were compared with ground truth in a real environment. These results demonstrate the value of this approach.

We assume Manhattan world where there are three mutually orthogonal vanishing directions and restrict the scope of recognition to two upper-layer corridor segment types and six lower-layer corridor segment types. In future work, we plan to recognize various structural landmarks such as hall, curving wall, and cross intersection corridor.

Acknowledgments

This work was supported by the Global Frontier R&D Program on <Human-centered Interaction for Coexistence> funded by the National Research Foundation of Korea grant funded by the Korean Government (MEST) (NRF-M1AXA003-2011-0028353).

References

1. P. Beeson, N. K. Jong and B. Kuipers, Towards autonomous topological place detection using the extended voronoi graph, in: *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, Barcelona, pp. 4373–4379 (2005).
2. G. Dedeoglu, M. J. Mataric, and G. S. Sukhatme, Incremental, on-line topological map building with a mobile robot, In Proceedings of SPIE, Mobile Robots XIV, Boston, pp. 129–139, 1999.
3. S. Marsland, U. Nehmzowb and T. Duckett, Learning to select distinctive landmarks for mobile robot navigation, *J. Robot. Auton. Syst.* **37**, 241–260 (2001).
4. V. A. Ramirez, J. B. Hayet, F. Lerasle and M. Devy, Visual localization of a mobile robot in indoor environments using planar landmarks, in: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Takamatsu, pp. 275–280 (2000).
5. A. Ranganathan and F. Dellaert, Bayesian surprise and landmark detection, in: *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, Kobe, pp. 2017–2023 (2009).
6. P. Ranganathan, J. B. Hayet, M. Devy, S. Hutchinson and Lerasle, Topological navigation and qualitative localization for indoor environment using multisensory perception, in: *9th International Symposium on Intelligent Robotic Systems*, Toulouse, pp. 497–506 (2001).
7. S. Thrun, Bayesian landmark learning for mobile robot localization, *Machine Learn. J.* **33**(1), 41–76 (1998).

8. B. L. Boada, D. Blanco and L. Moreno, Symbolic place recognition in voronoi-based maps by using hidden Markov models, *J. Intell. Robotic Syst.* **39**(2), 173–197 (2004).
9. O. Aycard, J.-F. Mari and F. Charpillet, Second order hidden Markov models for place recognition: new results, in: *Proc. of IEEE/ICTAI*, Taipei, pp. 408–415 (1998).
10. V. Beranger and J. Y. Herve, Recognition of intersections in corridor environments, in: *IEEE Intl. Conf. on Pattern Recognition (ICPR)*, Vienna, p. 133 (1996).
11. H. Liu, Z. Zhang, W. Song, Y. Mae, M. Minami and A. Seiji, Evolutionary recognition of corridor and turning using adaptive model with 3D structure, in: *SICE-ICASE Int. Joint Conf.* Busan, pp. 2840–2845 (2006).
12. J. M. Coughlan and A. L. Yuille, Manhattan world: orientation and outlier detection by Bayesian inference, in: *IEEE Intl. Conf. on Computer Vision (ICCV)*, Corfu, pp. 941–947 (1999).
13. G. Schindler and F. Dellaert, Atlanta world: an expectation maximization framework for simultaneous low-level edge grouping and camera calibration in complex man-made environments, in: *IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, Washington, pp. 203–209 (2004).
14. J. A. Shufelt, Performance evaluation and analysis of vanishing point detection techniques, *IEEE Trans. Pattern Anal. Machine Intell. (PAMI)* **21**(3), 282–288 (1999).
15. H. Wildenauer and M. Vincze, Vanishing point detection in complex man-made worlds, in: *IEEE Intl. Conf. on Image Analysis and Processing (ICIAP)*, Modena, pp. 615–622 (2007).
16. H. Kong, J. Y. Audibert and J. Ponce, Vanishing point detection for road detection, in: *IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, Miami, pp. 96–103 (2009).
17. D. Schröter, T. Weber, M. Beetz and B. Radig, Detection and classification of gateways for the acquisition of structured robot maps, *Lecture Notes in Computer Science (LNCS)*, Springer, pp. 553–561 (2004).
18. D. Schröter, M. Beetz and J. Gutmann, RG mapping: learning compact and structured 2D line maps of indoor environments, in: *IEEE Intl. Workshop on Robot and Human Interactive Communication (ROMAN)*, Tokyo, pp. 282–287 (2002).
19. D. Kortenkamp and T. Weymouth, Topological mapping for mobile robots using a combination of sonar and vision sensing, in: *AAAI Nat. Conf. on Artificial Intelligence*, CA, pp. 979–984 (1994).
20. J. B. Hayet, F. Lerasle and M. Devy, A visual landmark framework for mobile robot navigation, *J. Image Vision Comput.* **25**, 1341–1351 (2007).
21. C.M. Bishop, *Pattern Recognition and Machine Learning*, Springer (2006).
22. R. O. Duda, P. E. Hart and D. G. Stork, *Pattern Classification*, 2nd edn, Wiley (2000).
23. D. Zhang, D. G. Perez, S. Bengio, I. McCowan and G. Lathoud, Modeling individual and group actions in meetings with layered HMMs, in: *IEEE Trans. on Multimedia*, pp. 509–520 (2004).
24. R. J. Elliott, J. L. Aggoun and J. B. Moore, *Hidden Markov Models. Estimation and Control*, Springer (1995).
25. O. Cappé, E. Moulines and T. Rydén, *Inference in Hidden Markov Models*. Springer (2005).
26. C. John, A computational approach to edge detection, *IEEE Trans. Pattern Anal. Machine Intell. (PAMI)* **8**(6), 679–698 (1986).

About the Authors

Young-Bin Park received his B.S. degree in 2001 in Sociology at Hanyang University. He received his M.S. degree from the Department of Electronics and Computer Engineering at the same University in 2007. Currently, he is a PhD student in the Department of Electronics and Computer Engineering at Hanyang University. His research interests include computer vision, pattern recognition, artificial intelligence, machine learning, and probabilistic modeling/inference.

Il Hong Suh received his PhD degree in electrical engineering from the Korea Institute of Science and Technology 2000, he was with the Department of EECS, Ansan Campus, Hanyang University, where he was a full professor. From 2000 to the present, he has been with the college of Information and Communications, Hanyang University, Seoul, Korea, where he is a full professor. From August 2004 to July 2006, he also served as dean for the College of Information and Communications, Hanyang University, Seoul, Korea. He has served as the president of the Systems and Control Society of Korea Institute of Telecommunications Engineers (2005–2007) and president of the Korea Robotics Society in 2008. He has been involved in a number of Korea National Projects such as the Intelligent Robotics Frontier Research Program for 21st Century and URC (Ubiquitous Robotics Companion). He served as chair of the steering committee for RUPI (Robot Unified Platform Initiative) organized by former Korean Ministry of Information and Communications from 2006–2007. His research interests lie in the area of communications and intelligence for robots including web-based control of robots, ontology-based robot intelligence, context-adaptive action-coupled perception and learning, and robot software platforms. He has published more than 170 contributions on robotics and control.