Vision-based Global Localization of Large-Scale Indoor Environments with Hierarchical Map

Shih-Huan Tseng[1], Yu-An Peng [1], Jia-Yuan Yu[1], Li-Chen Fu[1,2], Shyh-Roei Wang[3]
Department of Computer Science and Information Engineering, National Taiwan University[1],
Department of Electrical Engineering, National Taiwan University[2],
Mechanical and Systems Research Laboratories, Industrial Technology Research Institute[3]
{shihhuan.tseng, YuAnnPeng, oeoeoeooeo}@gmail.com, lichen@ntu.edu.tw, SeriesWang@itri.org.tw

***Abstract*—In this paper, we propose a method to make the mobile robots building a vision-based hierarchical map and then quickly localizing itself by this map. The top-level map is a topological map which consists of "Places". Topological map is represented through the graph and suitable for the large-scale environments. In this map, the vertices comprise some visual information for efficiently and robustly identifying the "Places" of the robot's environment and the edges denote the spatial relation between these "Places". The visual information consists of the characteristics of segments and SIFT features. The former indexes "Place" coarsely and the latter is with the capability of indentifying the position within the current Place finely. When localizing itself, we merely compute the similarity by voting corresponding SIFT features between current information and the vertices indexed with the similar characterization of segments. This localization is efficient because of the low time complexity of characterization of segments and the few candidates of the sets of SIFT features. In the other aspect, the scale, orientation and illumination invariant of SIFT features make this localization robust.**

***Index Terms*—monocular, vision, global localization, indoor, hierarchical.**

I. INTRODUCTION

In recent years, the mobile robots enter human's life progressively rather than just stay in factories or laboratories [1] [2], hence localization and navigation become the fundamental abilities. In the other words, the robots must have the ability to know its current position over time and find the path to the destination [3].

There are many kinds of sensors with a capability to navigate in the unstructured environment, such as laser scanners, ultrasonic sensors, cameras and etc. The methodology to solve this problem for indoor settings by laser scanners is not an open problem anymore [4] [5] [6], but the laser scanners are expensive. Because of the rich information carried by pictures and the low cost of cameras, many people change their direction on these visual devices and put aside the old ones, laser scanners and ultrasonic sensors [7] [8]. However, solving this problem by visual sensors is still a great challenge.

The two key problems of mobile robot localization are global localization and local position tracking. Global localization is the problem to determine the robot's position in a previously learned map without any other information more than where the robot is. Global localization gives mobile robots capabilities to deal with initialization and recovery from "kidnaps" [9] [10]. Local position tracking is the problem to keep estimating the current position of the robot by the previous path and current sensor readings when it move around the environment [11].

For navigation tasks, the robot constructs a map when wandering the unknown environment in the first. There are two major paradigms of map, topological maps and metric maps. Topological maps [12] [13] are graph-based representation and is naturally inspired from human behaviors [14]. This model encodes the spatial relations between the places. The pro and con are sparse data but difficult to navigate directly. In the other way, metric maps represent [15] [16] environments by grids. The information carried by each cell of the grid is the absence of an obstacle or not. The strengths and weaknesses are more accurate representation and easy to navigate, but the time complexity and space complexity is much higher. In our approach, we combine these two paradigms for taking both advantages of them.

In this paper, we propose an efficient vision-based global localization method. When constructing the map, we take the characterization of segments of every image as the index for speeding up and extract the SIFT features for robust matching. Thus, it is necessary to develop techniques for more efficient localization.

II. Problem Description and Preliminaries

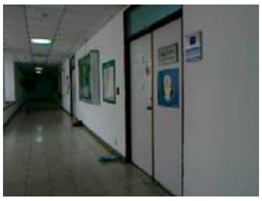
A. Image Segmentation

We use the graph-based segmentation algorithm which is proposed by Felzenszwalb et al. to segment the image efficiently [17]. There are many methods to segment images, such as clustering method, histogram-based method, etc. This method uses the graph-based representation to define a predicate for measuring the evidence of a boundary between two regions by comparing two quantities: one is intensity differences across the boundary, and the other is intensity differences between neighboring pixels attached to one region.

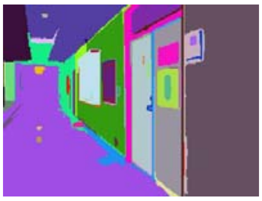
Based on this predicate, this algorithm can produce segmentation that satisfies global properties – running in time $O(\log n)$ for n image pixels. Fig(a) is the original 320x240 pixels image and Fig(b) is the result of image segmentation by the algorithm with parameters σ = 0.5 and k = 500, where σ is the standard deviation of the Gaussian filter to smooth the image and k is used to tune the size of components.

B. Scale Invariant Feature Transform

The scale invariant feature transform (SIFT) has been proposed by D. Lowe [18]. This feature has highly capability to against the change of scale, orientation and partial illumination. The descriptor we get is of dimension 128. The Fig.1 is the corresponding pairs of SIFT features between two images with the different views of a building.



Fig(a) Original image.



Fig(b) Result image.

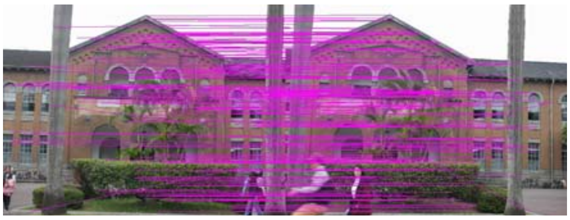


Fig.1 Corresponding SIFT features between two images.

III. Place Identification

In our approach, we construct a hierarchical database map. The top-level map is a topological map which is represented as a graph. Each vertex of this graph indicates a “Place” which is a collection of sequential images and contains the visual information, named “Place Identification”, for efficiently identifying this “Place”. Additionally, a small database map is attached to each vertex for subsequent robust localization.

For achieving efficient and robust localization in the large-scale environment by previous built map, we cluster a series of some successive images into a “Place” and use some salient characteristics as identity of the Place. By going through this section, the reader will understand how we cluster the successive images into Place and extract the visual information as Place Identification.

A. Diagram of Clustering

The procedure of clustering is shown in Figure 2.

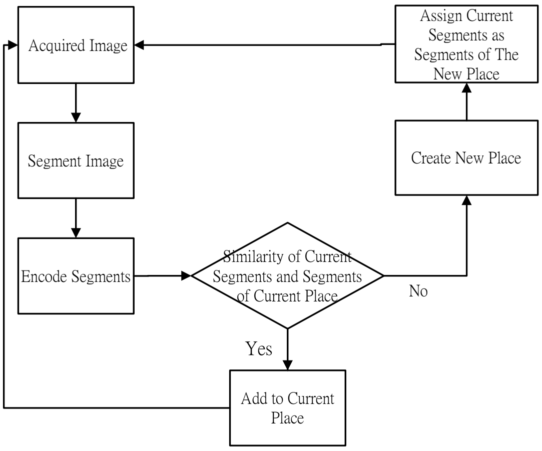

Acquired Image
Assign Current Segments as Segments of The New Place
Segment Image
Create New Place
Encode Segments
Similarity of Current Segments and Segments of Current Place
No
Yes
Add to Current Place

Fig. 2 Diagram of Clustering

After acquiring the image frame, we segment the image into some components using an efficient graph-based image segmentation which is proposed by Felzenszwalb et al. [17]. The details of the algorithm of image segmentation is mentioned in Section II-A. After image segmentation, we encode the segment components by computing simple global visual information; namely, expand color histogram, which would be introduced in Section III-B.

B. Segment Encoding

The procedure of segment encoding is shown as Figure 3. First, we segment a digital image as a few of segments by the method which is mentioned in Section II-A. Then, we encode these segments by expand color histogram. Finally, the input image would be represented as a few of expand color histogram.

After we segment the image using the algorithm as mentioned in Section II-A, we encode the image segments by computing a visual information –color histogram. We compute the expand color histogram of each segmented component.

In the beginning, we transform the RGB color space to HIS color space. The color space conversion equation is:

$$I = \frac{1}{3}(R+G+B)$$

$$S = 1 - \frac{3}{R+G+B}\left[\min(R,G,B)\right]$$

$$H = \cos^{-1}\left[\frac{\frac{1}{2}(R-G)+(R-B)}{\sqrt{(R-G)^2+(R-B)(G-B)}}\right] if B > G, then H = 360^0 - H \qquad (1)$$

Because hue is probably superior for human interpretation, we only take the channel of hue of the HSI color space.

Color histogram is one of the common methods to compute the visual information of images. The representation is:

$$A = (a^1, a^2, ..., a^n) \qquad (2)$$

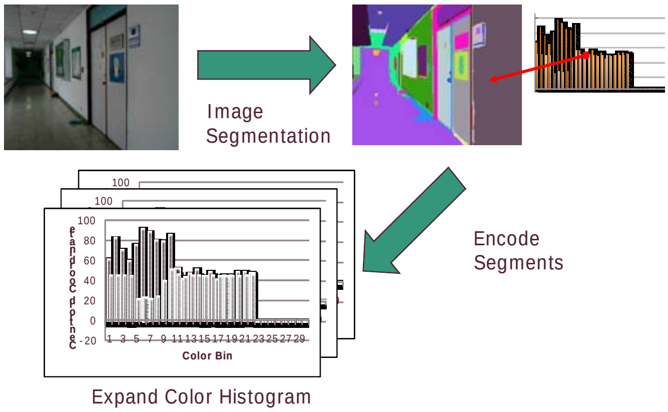


Fig. 3 Diagram of Clustering

, where n is number of color bin, and a^i denotes the number of pixels within color bin i.

For consideration of spatial relation, we choose the improved method of computing color histogram – Expand Color Histogram [19], which is expressed as below:

$$A = (a_x^1, a_y^1, a_x^2, a_y^2, ..., a_x^n, a_y^n) \qquad (3)$$

where n is number of color bin, (a_x^i, a_y^i) denotes the average value of coordinate of pixels within color bin i.

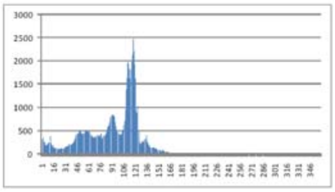


Fig. 4 The color histogram of Fig (a)

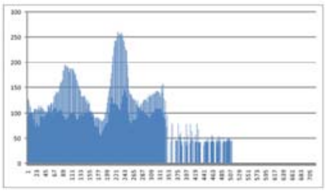


Fig. 5 The expand color histogram of Fig (b)

We use the expand color histogram to compute the visual information of the segment components which have been selected previously. Then, we assign these histograms to the Place Identification.

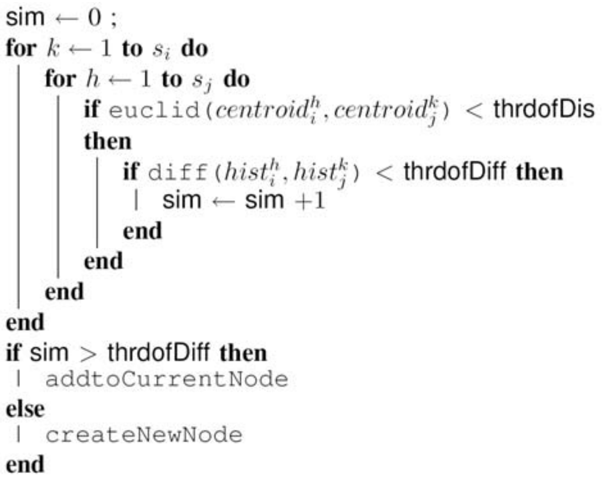
There are many kinds of measurement of difference between two histograms, such as least square, Euclidean distance, etc. Here, we choose the Euclidean distance for our difference measurement. Because the size of each segment component is not equal necessarily, this expand histogram would be normalized by the size of the area of the component. Therefore, the difference between the expand color histograms of A and B is:

$$\text{Dissimilarity(A,B)} = \sqrt{\sum_{i=1}^{n}(a_x^i - b_x^i)^2 + (a_y^i - b_y^i)^2}$$

, where a_x^i is the value of the color bin i in the histogram A.

C. Methodology of Clustering

After segment encoding, each image is represented as some expand color histograms. We define the similarity between two set of expand color histogram as the number of similar histograms. We use the incremental strategy to perform clustering process which is shown as Algorithm 1.



```
sim ← 0 ;
for $k \leftarrow 1$ to $s_i$ do
    for $h \leftarrow 1$ to $s_j$ do
        if euclid($centroid_i^h$, $centroid_j^k$) < thrdofDis
        then
            if diff($hist_i^h$, $hist_j^k$) < thrdofDiff then
                sim ← sim +1
            end
        end
    end
end
if sim > thrdofDiff then
    addtoCurrentNode
else
    createNewNode
end
```

Algorithm 1: Algorithm of Image Clustering

IV. Database Management and Localization

Because we use database map in our approach, we introduce how we manage and access our hierarchical database map. In the top-level map, we use the characteristics of segments of the query image as index to find the Place which is the entry of the database in the bottom level. Then, convert the query image into the inverted index of visual vocabulary using the method which is mentioned in this Section. Finally, assign the inverted index as index of the image which is the image in Place.

A. Image Representation

After clustering a series of successive images into a Place, we collect these images to build the local spatial map. We represent the ith image of the jth Place as:

$$I_j^i = \left\{ u_1^{i,j}, u_2^{i,j}, \dots, u_{|v_i|}^{i,j} \right\} \tag{5}$$

, if the word w_k^i is in the image img_j^i, then $u_k^{i,j}$ is the weight of this word; otherwise, $u_k^{i,j}$ is 0.

We extract the SIFT features from the acquired image, and represent the image img_j^i as:

$$S_j^i = \left\{ d_1^{i,j}, d_2^{i,j}, \dots, d_{n_j^i}^{i,j} \right\} \tag{6}$$

, which denotes the set of SIFT features extracted from img_j^i, and $d_k^{i,j}$ is the kth feature from img_j^i.

When building the local spatial map L_i, we construct the “visual vocabulary” v_i by clustering the similar SIFT features into terms for indexing [20]. We use the k-means algorithm which is proposed by Lloyd [21].

Each term of the vocabulary is named “word” w_k^i, $k = 1, 2, \dots, |v_i|$, where $|v_i|$ means the size of v_i. Then, the image img_j^i is represented as

$$\hat{S}_j^i = \left\{ \hat{d}_1^{i,j}, \hat{d}_2^{i,j}, \dots, \hat{d}_{n_j^i}^{i,j} \right\} \tag{7}$$

, where $\hat{d}_k^{i,j}$ is the word of vocabulary and n_j^i is number of the features in the image S_j^i.

More further, the image can be represented as a Boolean vector for indicating the absence or presence of any word w_k^i in the vocabulary v_i,

$$Z_j^i = \left[z_1^{i,j}, z_2^{i,j}, \dots, z_{|v_i|}^{i,j} \right] \tag{8}$$

, where $z_k^{i,j}$ is 0 or 1 which denotes the absence or presence of word w_k^i in the image img_j^i respectively.

The terms for indexing is named “word” . By computing the occurrence frequency of these words, we can get the inverted weight of each word. The weight of word w_k^i is:

$$u_k^i = weight(w_k^i) = \log N_i / N_k^i \tag{9}$$

, where N_i is the number of total times of all of the terms in the local spatial database L_i:

$$N_i = \sum_{j=0}^{n_i} n_j^i \tag{10}$$

, and N_k^i is the number of times that the term w_k^i appears in the Place:

$$N_k^i = \sum_{j=0}^{n_i} z_k^{i,j} \tag{11}$$

Then, the image I_j^i can be represented as:

$$I_j^i = \left[u_1^{i,j}, u_2^{i,j}, \dots, u_{|v|}^{i,j} \right] \tag{12}$$

, where $$u_k^{i,j} = \begin{cases} weight(w_k^i) = \log \dfrac{N_i}{N_k^i} & , if z_k^{i,j} \in Z_j^i \\ 0 & , otherwise \end{cases} \tag{13}$$

B. Database Management

In our approach, we use the hierarchical database with two levels. The top-level is the Place database M which is indexed by the Place Identification m_i as shown in Figure 6. Each vertex of top-level map is an entry of the local spatial database L_i in the bottom-level map. The bottom-level map consists of several databases L_i which contain the

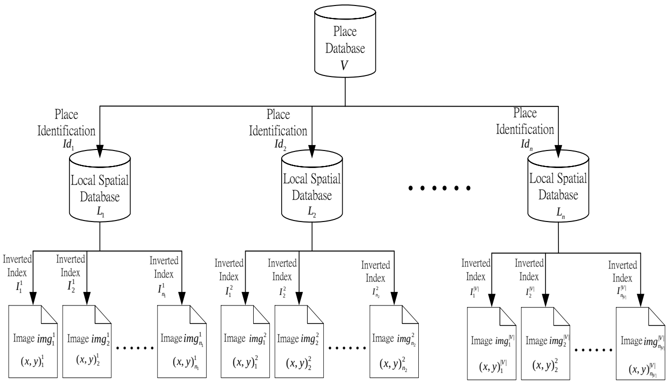


Fig. 6 The diagram of the hierarchical database

local spatial information. The element of local spatial database L_i is indexed by inverted index of vocabulary v_i.

The number of entries is the number of vertices in the top-level map $|V|$, which also means the number of Places. Each m_i, $i = 1, 2,\ldots, |V|$, consists of a Place Identification Id_i, local spatial database L_i, and a collection of some successive images img^i_j, $j = 1, 2,\ldots, n_i$.

The bottom-level maps of our hierarchical map are some databases. Each local spatial map is stored such a database. Each element of the local database l^i_j, $j = 1, 2,\ldots, n_i$, consists of inverted index I^i_j and coordinate $(x, y)^i_j$.

C. Robust Localization

The localization is achieved by two stages. The first is to identify the Place in the topological map by finding the entry of the database into the bottom-level map according to Place Identification. The second is to locate the position of the local spatial map by computing the dissimilarity between the query image which is represented as inverted index and the images within the Place using the equation (13).

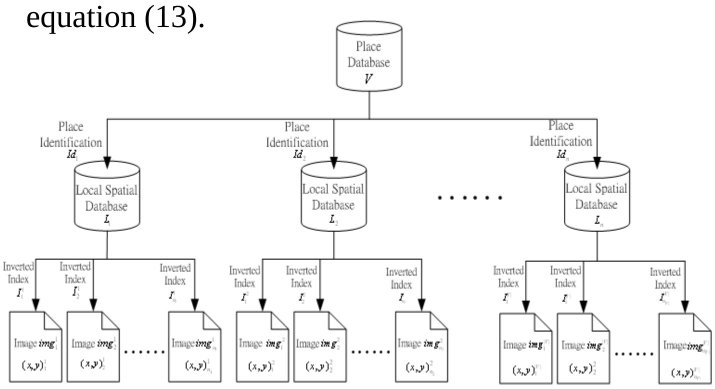
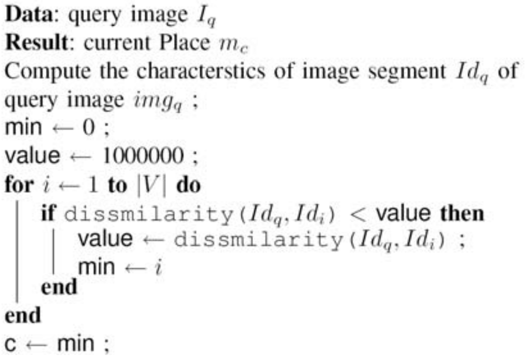


Fig. 7 The diagram of the localization

In Figure 7, we compute the index graph by encoding the segments of the query image. Simultaneously, we extract the SIFT features from query image. Then, we use the characteristics of segments as index to find the current Place and the entry of the database in the bottom-level map. Additionally, we convert the SIFT features into inverted index using the visual vocabulary which is attached to the current Place. Finally, we locate the current position using inverted index as an index.

In the first stage, we must find the Place m_c where the query image img_q indicates. First, we segment the query image using the method which is mentioned in Section II-A. Then, we encode the segment of the image after segmentation using the method which is mentioned in Section III-B. Finally, we identify the current Place using Algorithm 2.



```
Data: query image $I_q$
Result: current Place $m_c$
Compute the characterstics of image segment $Id_q$ of
query image $img_q$ ;
min ← 0 ;
value ← 1000000 ;
for $i \leftarrow 1$ to $|V|$ do
    if dissmilarity($Id_q, Id_i$) < value then
        value ← dissmilarity($Id_q, Id_i$) ;
        min ← $i$
    end
end
c ← min ;
```

Algorithm 2: Algorithm of Place Identify

In Algorithm 2, dissimilarity is the function which computes the measurement of the dissimilarity between two characteristics of segments and $|V|$ is the number of Places.

After identifying the Place, we can represent query image img_q as the inverted index I^c_q of the visual vocabulary v_c which is attached to the Place. Then, we perform localization by finding the minimum dissimilarity of the current image and all of the images of this place.

The similarity score between image I^c_q and image I^c_j is:

$$\text{Simillarity}(I^c_q, I^c_j) = \frac{\sum_{k=0}^{|v_c|} u_k^{c,q} u_k^{c,j}}{\sqrt{\sum_{k=0}^{|v_c|}\left(u_i^{c,q}\right)^2}\sqrt{\sum_{k=0}^{|v_c|}\left(u_i^{c,j}\right)^2}} \tag{14}$$

V. Experiments

A. Hardware of the Experimental System

Experiments are held on our newly designed robot, Home Robot. Home Robot is equipped with a laser rangefinder, ultrasonic sensing system. Furthermore, with an on-board dome camera, the Home Robot has an even wider range of view which allows self-localization algorithms based on vision to be run more efficiently. The Home Robot is designed to have a friendly appearance, the picture below shows what it looks like.

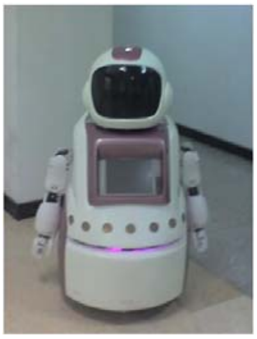


Fig. 8 Home Robot

B. Experimental Environment

The one of our experimental environment is a home environment in Open Lab. This environment contains five rooms, such as living room, bath room, shown in Figure 9.

C. Performance of Clustering

In the open Lab, we took about 400 pictures by the camera which is equipped with Home Robot. The average number of clusters is 63. The average number of images within each cluster is 6.34. The sequences of images in Figure 10 are some clusters within open Lab.

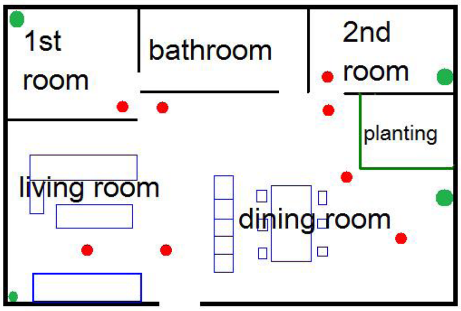


Fig. 9 The home environment in Open Lab

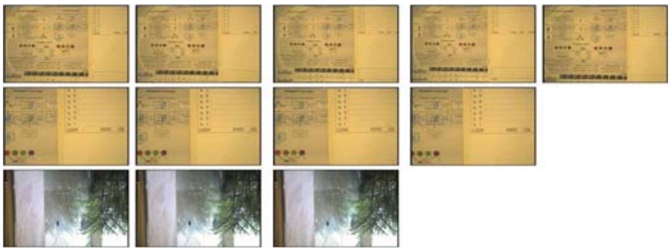


Fig. 10 Some clusters within open Lab

D. Performance of Global Localization

In the open Lab, we simultaneously take pictures by the camera equipped with Home robot and record the world coordinates. Although the numbers of total images are about 2000, we only construct database by 400 images. Because the moving speed of Home Robot is slow, we merely select one from five images to construct out database.

In this experiment, the output is the world coordinate. To avoiding returning the wrong coordinate, we set a threshold of the similarity of inverted index. If similarity between inverted index of the most two similar images is lower than the threshold, we return "insufficient information" and the robot would wander and take other pictures for localization.

There are some correct results in Figure 11(a). The text of the above textbox is the world coordinates of test image. The text of the below textbox is the information of result. The first item of the below textbox is current cluster. The second item denotes the similarity of inverted index between these images. The last item is world coordinate of result image. If the similarity of inverted index is smaller than 0.1, we would return "insufficient information" and make the robot take other pictures for performing localization.

In certain condition, we would get incorrect results, such as complex scenes, similar scenes, and etc. We list some incorrect results in Figure 11(b). In the first image of Figure 11(b), the scene of test image is very complex, so that the image segmentation would be meaningless for classify and identify.

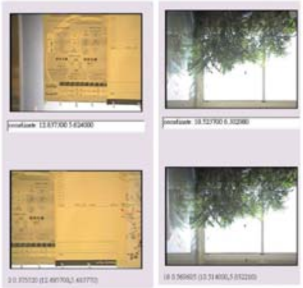


Fig. 11(a) Correct results.

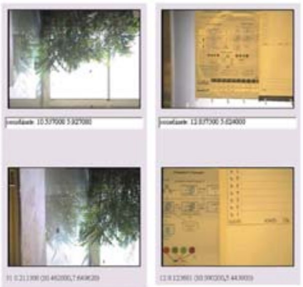


Fig. 11(b) Incorrect results.

In the future, we would improve the rate of recognition by pre-processing these images.

In this experiment, if the threshold of inverted index is higher, the number of cases of “insufficient information” would be more. Because the wrong coordinate would cause the robot damage, we set a higher threshold. There are 235 correct results within 300 test images. The rate of correct recognition is 0.783. On the other part, there are 9 false positive within 300 test images. The rate of false positive is 0.03. The average distance between test image and result image is 0.67m.

IV. Conclusion

In this paper, we propose a method which uses the combination of two types of features to denote a location. One type of features is the texture characterization which is simple and time-conserving. However, its toleration of illumination and viewpoint changes is low. The other type of feature is SIFT feature which robust against scale, orientation and partial illumination. At the same time, the time consumption of computing this feature is huge. We take the respective advantages of both. This method of solving global localization problem is robust and efficient.

Reference

[1] “Aibo remote framework(rfw),” available at http://openr.aibo.com.

[2] H. Asoh, Y. Motomura, F. Asano, I. Hara, S. Hayamizu, K. Itou, T. Kurita, T. Matsui, N. Vlassis, R. Bunschoten, et al., “Jijo-2: An Office Robot That Communicates and Learns,” 2001.

[3] E. Motard, E. Motard, B. Raducanu, V. Cadenat, and J. Vitria, “Incremental on-line topological map learning for a visual homing application,” in Proc. IEEE International Conference on Robotics and Automation, B. Raducanu, Ed., 2007, pp. 2049–2054.

[4] L. Armesto, G. Ippoliti, S. Longhi, and J. Tornero, “FastSLAM 2.0: Least-Squares Approach,” IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 5013–5018, 2006.

[5] C.-C. Wang, “Simultaneous localization, mapping and moving object tracking,” Ph.D. dissertation, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, April 2004.

[6] T. Jost and H. Hugli, “Fast icp algorithms for shape registration,” in Proceedings of the 24th DAGM Symposium on Pattern Recognition. London, UK: Springer-Verlag, 2002, pp. 91–99.

[7] P. Newman, D. Cole, and K. Ho, “Outdoor slam using visual appearance and laser ranging,” in Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on, 2006, pp. 1180– 1187.

[8] J. Wang, H. Zha, and R. Cipolla, “Coarse-to-fine vision-based localization by indexing scale-invariant features,” Systems, Man, and Cybernetics, Part B, IEEE Transactions on, vol. 36, no. 2, pp. 413–422, April 2006.

[9] J. Wang, J. Wang, R. Cipolla, and H. Zha, “Vision-based global localization using a visual vocabulary,” in Proc. IEEE International Conference on Robotics and Automation ICRA 2005, R. Cipolla, Ed., 2005, pp. 4230–4235.

[10] R. Sim and G. Dudek, “Learning environmental features for pose estimation,” Image Vision Comput., vol. 19, no. 11, pp. 733–739, 2001.

[11] P. Besl and H. McKay, “A method for registration of 3-d shapes,” Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 14, no. 2, pp. 239–256, Feb 1992.

[12] B. K. Emilio Remolina, “Towards a general theory of topological maps,” Artificial Intelligence, 2004.

[13] H. Choset, H. Choset, and K. Nagatani, “Topological simultaneous localization and mapping (slam): toward exact localization without explicit localization,” vol. 17, no. 2, pp. 125–137, 2001.

[14] D. Crisan and A. Doucet, “A survey of convergence results on particle filtering for practitioners,” 2002.

[15] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, “Fastslam: A factored solution to the simultaneous localization and mapping problem,” in Proceedings of the AAAI National Conference on Artificial Intelligence. Edmonton, Canada: AAAI, 2002.

[16] H. P. Moravec, “Sensor fusion in certainty grids for mobile robots,” AI Magazine, 1988.

[17] P. Felzenszwalb and D. Huttenlocher, “Efficient Graph-Based Image Segmentation,” International Journal of Computer Vision, vol. 59, no. 2, pp. 167–181, 2004.

[18] D. Lowe, “Distinctive image features from scale-invariant keypoints,” in International Journal of Computer Vision, vol. 20, 2003, pp. 91–110.

[19] H. Greenspan, J. Goldberger, and L. Ridel, “A continuous probabilistic framework for image matching,” Computer Vision and Image Understanding, vol. 84, no. 3, pp. 384–406, 2001.

[20] J. Sivic and A. Zisserman, “Video Google: a text retrieval approach to object matching in videos,” Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on, pp. 1470–1477, 2003.

[21] T. Kanungo, D. Mount, N. Netanyahu, C. Piatko, R. Silverman, and A. Wu, “An Efficient k-Means Clustering Algorithm: Analysis and Implementation,” IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, pp. 881–892, 2002.