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

Self-location Recognition Using Azimuth Invariant Features and Wearable Sensors

Takayuki Katahira and Yoshio Iwai

Abstract— **Self-location capability is a very useful and informative attribute for wearable systems. This paper proposes a method for identifying a user's location from an omnidirectional image sensor, a GPS data source and wireless LAN data. Azimuth-invariant features are extracted from an omnidirectional image by integrating pixel information circumferentially, thus enabling a user to independently recognize his/her location from the omnidirectional image feature, the GPS data and the wireless LAN data projected into a sub-space made from the learning data. We show the effectiveness of our method by experimental results in real data.**

I. INTRODUCTION

In recent years, the potential for developing "wearable" computers has significantly increased because of the progress in science and technology. Typical wearable computers which are small and unobstructive[1] include watches[2], rings[3], and headphones[4]. The concept of "wearable computers" is gaining socially acceptability. Most wearable computers have hands-free capability and can support various tasks. In particular, they are widely used for providing wearer's location information in various applications. In medical practice, a monitoring system for patients has been developed by using wearable computers[5]. Such a system can recognize the patient's states and in an emergency provide the patient's location to a doctor. Thus, self-location is very informative for supporting various tasks. Many different sensors can be used to acquire environmental information for obtaining the self-location information such as range finders[6], image sensors[7], the GPS[8], and the wireless LAN[9]. In this paper, we propose a method for identifying a user's location from an omnidirectional image sensor[10], GPS data and wireless LAN data.

In self-location recognition by using image sensors, 3D reconstruction methods are frequently used[11]. These methods, however, have problems of huge computational cost and other difficulties in 3D reconstructions. To minimize these problems, a memory based navigation approach has been proposed[12]. Memory based navigation keeps the training images as a location database, and recognizes self-location changes by comparing an input image with the location database. This method need not reconstruct a 3D scene from the images, and can improve the location recognition rate even in complex scenes by adding training images into the location database.

Self-location recognition by using the GPS is widely used in car navigation systems, airplanes and ships. The GPS can acquire self-location information directly from satellites, and can recognize the location anywhere where the signal of the satellites can be received. The identification method by using GPS and other sensors can recognize the location even in indoor environments[8]. The main problem with the GPS, however, is that the accuracy of location information varies because of the number of the "visible" satellites. More GPS satellites are required to improve the accuracy, but satellites are costly to launch.

Recently self-location recognition by using wireless LANs has been proposed. This method uses signal strengths and the network names of wireless LANs for location recognition[13]. Self-location recognition by using wireless LANs can recognize self-locations in subways, buildings, and so on. The method can accurately recognize locations in cities because there are many wireless LAN access points; however the recognition rate becomes less effective where wireless LAN access points are sparse.

As describe above, each of the sensors has advantages and disadvantages. We therefore propose a method for identifying a user's location by using a combination of multiple sensors: the omnidirectional image sensor, the GPS, and the wireless LAN to combine the advantages of each of these sensors. We also propose to use the azimuth invariant feature extracted from the omnidirectional images, and we show the effectiveness of our method by experimental results with real data.

II. AZIMUTH INVARIANT FEATURES

In this paper, we use an omnidirectional image sensor as shown in Figs. 1 and 2. The sensor consists of a downward facing camera and an upward facing hyperbolic mirror. The sensor has the same optical characteristics as a common camera, and captures 360 degrees panoramic view at any given time. We can extract azimuth invariant features from the omnidirectional images by using these characteristics. In general, the omnidirectional images captured at the same position differs form each other when the sensor's orientation changes as shown in Fig. 3. This problem is also resolved by using the azimuth invariant features.

As shown in Fig. 4, we consider the polar coordinate system (r,θ) with the origin fixed at the center of the image. For any feature $f(r,\theta)$ extracted at the position (r,θ), we can calculate the azimuth invariant feature $X(r)$ by the following integral transform:

$$X(r) = \int_0^{2\pi} f(r,\theta)d\theta = \int_0^{2\pi} f(r,\theta+\Delta)d\theta, \qquad (1)$$

Graduate School of Engineering Scence, Osaka Unieristy, 1–3 Machikaneyama, Toyonaka, Osaka 560–8531 Japan
iwai@sys.es.osaka-u.ac.jp

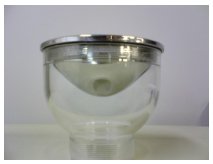

Fig. 1. Hyperbolic mirror

Fig. 2. Omnidirectional image sensor

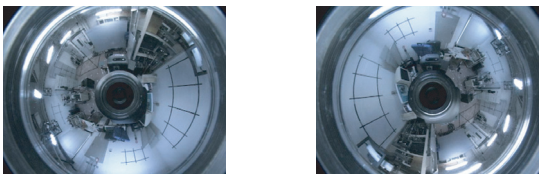

Fig. 3. Omnidirectional images captured at the same point with different orientations

where Δ is the perturbation of the azimuth θ. For any Δ, the above equation is always satisfied. This integral transform is difficult to calculate for a common camera because a common camera cannot capture a 360 degrees panoramic image at any one time. However, by using an omnidirectional image sensor, we can easily calculate such an integral transform. Therefore, the azimuth invariant feature is a suitable feature for omnidirectional image sensors. We can use autocorrelation or differentiation as the function f. In the following section, we describe the self-location recognition system as an application of invariant features and show experimental results.

III. Self-location Recognition System

In this section, we describe a self-location recognition system using the azimuth invariant features, the GPS data, and the wireless LAN data.

A. Overview of the system

Fig. 5 shows the process flow of the self-location recognition system from capturing the wearable sensors' positional information to outputting the recognition result. We use an omnidirectional image sensor[10] for capturing image data, and use the GPSMAP60CSx (Garmin) for acquiring the GPS data. For the wireless LAN data, we use a notebook computer to get signal strengths and network names. Fig. 6 shows a user wearing a prototype sensor system.

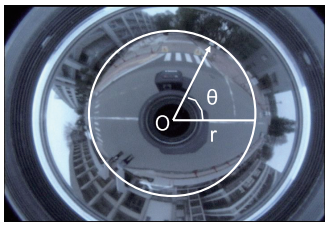

Fig. 4. Polar coordinate system

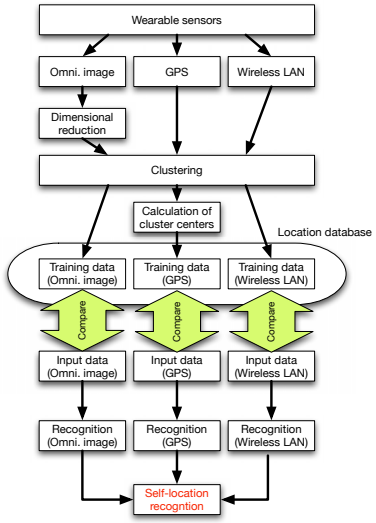

Fig. 5. Process flow of the system

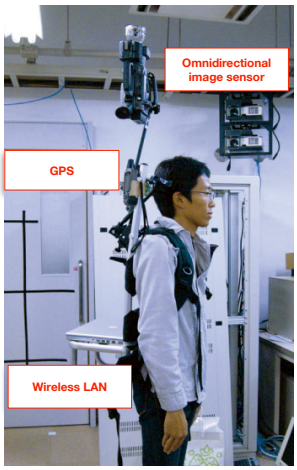

Fig. 6. A user wearing a prototype sensor system

After data acquisition, image features are extracted from the captured omnidirectional images, and extracted image features are compressed by the KL transform. The GPS data contain elevation, latitude, longitude and time, and the wireless LAN data contain signal strengths and network names. After feature extraction, the training data are clustered by the k-mean clustering algorithms and then, each cluster is labeled. The center of each cluster is calculated after the clustering. At the clustering phase, recently acquired training data can be considered as positional near data, so the GPS data are clustered with corresponding time information. Finally, all labeled clusters and training data are separately stored into a location database. Self-location recognition is independently achieved by comparing input data with data in the location database. When the recognition results differ, the system checks the confidence of the recognition results, and then makes the final decision on any positional changes.

The recognition and discrimination methods are described in section III-D.

B. Features

In this section, we describe features used in the system. We use normalized RGB features and differentiation features as image features and these features are considered azimuth invariant. We also use elevation, latitude, and longitude in the GPS data, and signal strengths and network names in the wireless LAN data.

1) The normalized RGB feature: The normalized RGB features are autocorrelations of the RGB pixel values on the circle $C(r,\theta)$ as shown in Fig. 4. To avoid effects of illumination changes, we use a color vector $(R',G',B')^T$ that normalizes the RGB values so that its length is 1. Autocorrelation of $(R',G',B')^T$ on the circle $C(r,\theta)$ is approximated by the Monte Carlo method and calculated by the following equation:

$$\begin{aligned} X(r)_{RGB} &= \int_C \begin{pmatrix} R'(r,\theta) \\ G'(r,\theta) \\ B'(r,\theta) \end{pmatrix} \begin{pmatrix} R'(r,\theta) \\ G'(r,\theta) \\ B'(r,\theta) \end{pmatrix}^T d\theta \\ &\simeq \sum_{i=0}^{k} \begin{pmatrix} R'(r,2\pi i/k) \\ G'(r,\pi i/k) \\ B'(r,\pi i/k) \end{pmatrix} \begin{pmatrix} R'(r,2\pi i/k) \\ G'(r,\pi i/k) \\ B'(r,\pi i/k) \end{pmatrix}^T \\ &= \sum_{i=0}^{k} \begin{pmatrix} R'R'_i & R'G'_i & R'B'_i \\ G'B'_i & G'G'_i & G'B'_i \\ B'R'_i & B'G'_i & B'B'_i \end{pmatrix} \\ &= \begin{pmatrix} RR_{RGB} & RG_{RGB} & RB_{RGB} \\ GR_{RGB} & GG_{RGB} & GB_{RGB} \\ BR_{RGB} & BG_{RGB} & BB_{RGB} \end{pmatrix}, \end{aligned} \tag{2}$$

where k is the number of samples. The autocorrelation matrix $X(r)$ is a symmetric matrix, so $X(r)$ is vectorized to a feature vector $\phi(r)_{RGB}$ by using 6 components as follows:

$$\begin{aligned} &\phi_{RGB}(r) = \\ &\quad (RR_{RGB}, RG_{RGB}, RB_{RGB}, GG_{RGB}, GB_{RGB}, BB_{RGB})^T. \end{aligned} \tag{3}$$

For a single omnidirectional image, the normalized RGB feature Φ_{RGB} is generated by varying the radius from r_1 to r_n as follows:

$$\Phi_{RGB} = (\phi_{RGB}(r_1), \phi_{RGB}(r_2), \cdots, \phi_{RGB}(r_n))^T. \tag{4}$$

2) The spatial differentiation feature: The spatial differentiation feature is calculated in the same way as the normalized RGB feature by using differential values instead of pixel values. The differential value of the RGB pixel is calculated as follows:

$$X(r)_{dif} = \int_C \begin{pmatrix} \frac{\partial R(r,\theta)}{\partial r} \\ \frac{\partial G(r,\theta)}{\partial r} \\ \frac{\partial B(r,\theta)}{\partial r} \end{pmatrix} \begin{pmatrix} \frac{\partial R(r,\theta)}{\partial r} \\ \frac{\partial G(r,\theta)}{\partial r} \\ \frac{\partial B(r,\theta)}{\partial r} \end{pmatrix}^T d\theta. \tag{5}$$

The above calculation is performed by a Monte Carlo approximation and finally, by varying the radius from r_1 to r_n, we get:

$$\Phi_{dif} = (\phi_{dif}(r_1), \phi_{dif}(r_2), \cdots, \phi_{dif}(r_n))^T, \tag{6}$$

where $\phi_{dif}(r_i)$ is a spatial differentiation feature vector on the circle with radius r_i.

3) Data compression: The number of dimensions of the azimuth invariant features extracted from an omnidirectional image is still high, and it takes considerable computation time to recognize the self-location position. Therefore, the azimuth invariant features are compressed by using the KL transform. By means of the transformation matrix Q calculated by the KL transform, all azimuth invariant features are compressed as follows:

$$\Psi_x = Q^T \Phi_x, (x = RGB, dif). \tag{7}$$

We use the compressed vectors Φ_{RGB}, and Φ_{dif} as the image features extracted from an omnidirectional image, and store them in the location database. The distance function in the image feature space, which is used in the clustering and classification processes, is defined as follows:

$$d_{img}(\Phi_i, \Phi_j) = \sum_{k=0}^{M} \left| \psi_{i,k} - \phi_{j,k} \right|, \tag{8}$$

where M is the number of dimensions of the compressed features, and satisfies $M < N$, i.e. L^1-norm is used for the metric in the image feature space.

4) The GPS feature: We use the elevation, latitude, longitude and time outputs. The units of latitude and longitude are in degrees, and the unit of elevation is in meters. The GPS time is the atomic time scale implemented by the atomic clocks. The GPS time was zero at 0000 hours on 6th January 1980 and since it is not perturbed by leap seconds, GPS is now ahead of UTC by 15 seconds. We, therefore, compensate for the time difference in advance. The number of dimensions of the GPS data is only 4, so we need not compress it. The distance function between two points $\mathbf{X}_{gps}, \mathbf{Y}_{gps}$ is defined as follows:

$$\begin{aligned} d_{gps}(\mathbf{X}_{gps}, \mathbf{Y}_{gps}) &= ||\mathbf{X}_{gps} - \mathbf{Y}_{gps}|| \\ &= \sqrt{{\Delta_1}^2 + {\Delta_2}^2 + (x_{ele} - y_{ele})^2}, \\ \Delta_1 &= A\Delta\alpha \cos x_{lon}, \\ \Delta_2 &= A\Delta\beta, \\ \Delta\alpha &= y_{lat} - x_{lat}, \\ \Delta\beta &= y_{lon} - x_{lon}, \end{aligned} \tag{9}$$

where $x_{lat}, x_{lon}, x_{ele}, A$ are latitude, longitude, elevation, and equatorial radius, respectively.

5) The wireless LAN feature: We use the signal strengths and network names of the wireless LAN as the wireless LAN features. Network names are expressed by character strings, and signal strength is expressed by percentage. The distance function between two wireless LAN features $\mathbf{X}_{lan}, \mathbf{Y}_{lan}$ is

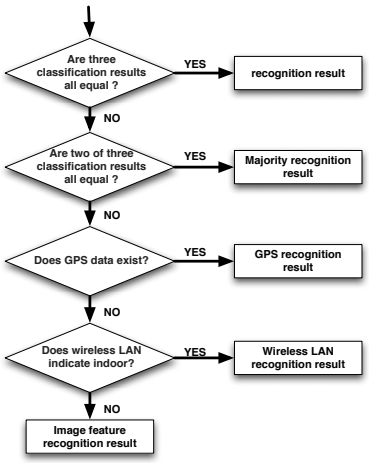

Fig. 7. Decision Flow

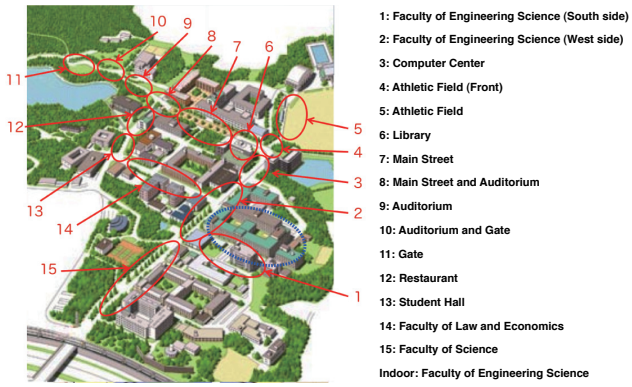

Fig. 8. Experimental area (outdoor)

defined by the following equation:

$$d_{lan}(\mathbf{X}_{lan},\mathbf{Y}_{lan}) = \sum_{i=0}^{n}\sum_{j=0}^{m}\delta(x_{name,i},y_{name,j})g(x_{sig,i},y_{sig,j}) \quad (10)$$

$$\delta(x,y) = \begin{cases} 1 & x=y, \\ 0 & otherwise \end{cases},$$

$$g(x,y) = \begin{cases} 0 & x\cdot y=0, \\ x/y & x\leqq y \cap x\cdot y\neq 0, \\ y/x & otherwise \end{cases},$$

where $x_{name,i}, x_{sig,i}$ are i-th network name and signal strength of the wireless LAN, respectively. n,m are the numbers of received signals.

C. Clustering

For self-location recognition, we label clusters of training data, and the minimum area of location recognition corresponds to these clusters. Input data taken by wearable sensors are classified into the clusters, which provide the self-location information. We use the k-mean algorithm for clustering, and the distance functions described above are used for the k-mean algorithm. We denote $W_j(j=1,2,\cdots,c)$ as a cluster obtained by the algorithm.

D. The Classification Method

In the proposed system, self-location recognition is independently performed for each feature: the azimuth invariant feature, the GPS feature and the wireless LAN feature. When three classification results are same, the system outputs the same result, but when the classification results are different, the final decision is made by the procedure shown in Fig. 7. We describe each classification method in the following sections.

1) Classification with the azimuth invariant feature: Classification with the azimuth invariant feature is performed by the k-nearest neighbor method which uses image features compressed by the KL transform. The decision rule is expressed by the following equation:

$$W = \max_{j=1,\cdots,c} \#W_j \rightarrow \mathbf{X}_{img} \in W, \quad (11)$$

where $\mathbf{X}_{img}$ is an input image feature, $\#W_j$ is the number of samples among the k-th nearest neighbors in the location database. W is the most frequent class label among the k nearest samples, and an input image feature $\mathbf{X}_{img}$ is assigned to the class W.

2) Classification with the GPS feature: Classification with the GPS feature is performed by the nearest-neighbor method. The distance between the input GPS feature $\mathbf{X}_{gps}$ and the center $\mathbf{C}(W_j)$ of the cluster W_j in the location database is calculated by (9). The decision rule is expressed by the following equation:

$$W = \arg\min_{j=1,\cdots,c} d_{gps}(\mathbf{C}(W_j)-\mathbf{X}_{gps}) \Rightarrow \mathbf{X}_{gps} \in W. \quad (12)$$

3) Classification with the wireless LAN feature: Classification with the wireless LAN feature is performed by the k-nearest neighbor method. The distance between the input wireless LAN feature $\mathbf{X}_{lan}$ and the cluster center $\mathbf{C}(W_j)$ in the location database is calculated by (10). The decision rule is expressed by the following equation:

$$W = \max_{j=1,\cdots,c} \#W_j \rightarrow \mathbf{X}_{lan} \in W. \quad (13)$$

W is the most frequent class label among the k nearest samples, and an input image feature X_{lan} is assigned to the class W.

IV. EXPERIMENTS

We conducted self-location recognition experiments to show the effectiveness of our method. Omnidirectional video sequences were captured by a walking person, and a number of selected still images per second. The still images and the corresponding GPS and wireless LAN data were used as training data. The training data were captured on 24th September, 2008 at our University, and the verification data were captured on 4th October, 2008. 500 items of training data and associated verification data were recorded. 25 clusters of training data are shown in Figs. 8 and 9. Numbers 1 to 15 are outdoor scenes, and Numbers 16 to 25 are indoor scenes. We also compared the recognition accuracy of the normalized RGB feature with that of the spatial differentiation feature.

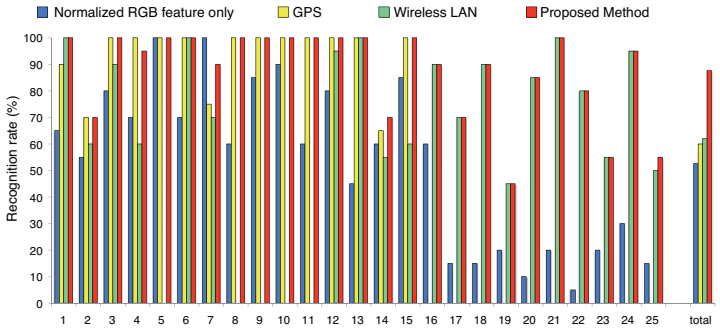

Fig. 11. Recognition rates of the proposed system using normalized RGB features and other sensor data

Area No.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	70	30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	5	95	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	10	90	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0
14	0	15	0	0	0	0	0	0	0	0	0	0	15	70	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	90	0	0	0	10	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	70	25	5	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	90	0	0	0	0	0	0	10
19	0	0	0	0	0	0	30	0	0	0	0	0	0	0	0	0	15	5	0	45	0	0	0	5	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	5	5	85	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	80	0	15
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	40	0	0	55	0	0
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	95	0
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	40	0	0	0	0	0	0	55

TABLE I

CONFUSION MATRIX OF THE PROPOSED SYSTEM USING NORMALIZED RGB FEATURES

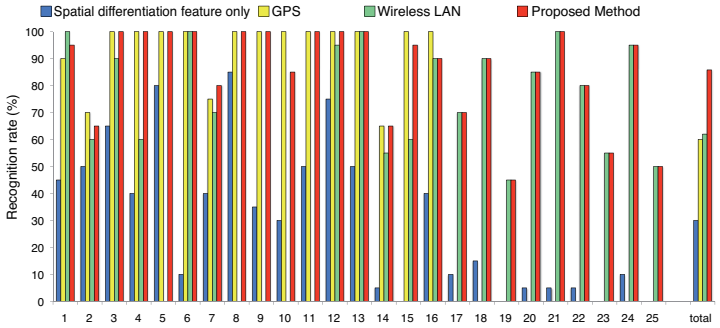

Fig. 12. Recognition rates of the proposed system using spatial differentiation features and other sensor data

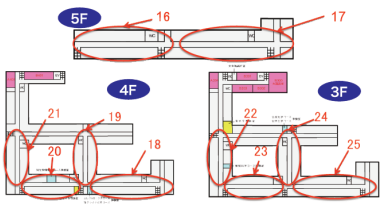

Fig. 9. Experimental area (indoor)

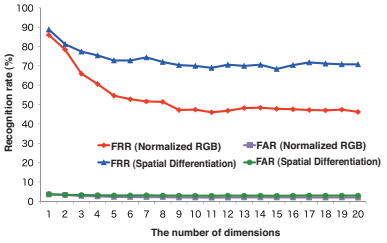

Fig. 10. Recognition rate by image features

A. *Effects of feature dimension*

We compressed the azimuth invariant features, because of the high dimensionality of the feature vector. The high dimensionality results in a sparseness of the feature space, and affects the accuracy of the clustering and the recognition rate. Therefore, we conducted an experiment to investigate the accuracy change against the change of the dimensionality

Area No.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1	95	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0
2	10	65	20	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	20	80	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	85	0	15	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0
14	0	15	0	0	0	0	0	0	0	0	0	0	20	65	0	0	0	0	0	0	0	0	0	0	0
15	5	0	0	0	0	0	0	0	0	0	0	0	0	0	95	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	90	0	0	0	10	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	70	25	5	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	90	0	0	0	0	0	0	10
19	0	0	0	0	0	0	30	0	0	0	0	0	0	0	0	15	5	0	45	0	0	0	5	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	5	85	0	5	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	5	80	5	0	5
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	40	0	0	55	0	0
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	95	0
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	45	0	0	0	5	0	0	50

TABLE II

CONFUSION MATRIX OF THE PROPOSED SYSTEM USING SPATIAL DIFFERENTIATION FEATURES

of the image features. The recognition in this experiment is performed by the image features only. The experimental result is shown in Fig. 10. According to the result, recognition rates of both features increase when the dimensionality of features is low, but when the number of dimensions of feature is greater than 10, the recognition rate does not increase and deteriorates. Therefore, we decided to use 10 dimensional image features in all future experiments on balancing useful results versus computation cost.

B. Self-location recognition

We have conducted self-location recognition experiments using two systems: one uses the normalized RGB features and the other uses spatial differentiation features. The recognition rates of both systems are shown in Figs. 11 and 12, and the confusion matrices are shown in Tables I and II. The i-th row and the j-th column element expresses the percentage that the i-th area datum is recognized as the j-th area.

From Figs. 11 and 12, the proposed system achieves a high accuracy rate by using multiple sensors throughout. Comparing the results using normalized RGB features, with the results using spatial differentiation features, the former system has better performance than the latter system. Normalized RGB features can express the characteristics of places from the omnidirectional images irrespective of illumination changes. The recognition performance of indoor scenes is not good using the azimuth invariant features, because of the lack of variety in indoor scenes. The azimuth invariant feature is a feature extracted by integral transform, so slight changes of indoor scenes cannot affect the transform. Therefore, in order to recognize indoor scenes more accurately, we need to extract image features sensitive to local image changes, but this is a trade-off problem that is affected by the noise.

V. CONCLUSION

This paper proposes a method for self-location recognition using an omnidirectional image sensor and other wearable sensors. The proposed method uses azimuth invariant features extracted from the omnidirectional images, the GPS data and the wireless signals from the wearable sensors. The results demonstrate that through wearable sensors the method can recognize self-location indoors and outdoors with a high accuracy rate. In future work, we will improve the recognition method so that it is more accurate and robust against noisy data.

REFERENCES

[1] K. Tsukada, and M. Yasumura, "ActiveBelt: Belt-type Wearable Tactile Display for Directional Navigation," *Proc. of UbiComp2004*, pp. 384–399, 2004.

[2] N. Kern, S. Antifakos, B. Schiele, and A. Schwaninger, "A Model for Human Interruptability: Experimental Evaluation and Automatic Estimation from Wearable Sensors," *Proc. of the Eighth IEEE Intl. Symp. on Wearable Computers*, Vol. 1, pp. 158–176, 2004.

[3] M.Fukumoto, and Y.Tonomura, "Body Coupled FingeRing: Wireless Wearable Keyboard," *Proc. of the ACM Conference on Human Factors in Computing Systems*, pp. 147–154, 1997.

[4] K. Hirota and M. Hirose, "An Implementation of Wearable Auditory Interface," *Proc. of MOVIC 2002*, pp. 570–575, 2002.

[5] P. Lukowicz et al., "AMON: a wearable medical computer for high risk patients," *Proc. Sixth International Symposium on Wearable Computers*, Seattle, Washington, pp. 133–134, 2002.

[6] H. Surmann, A. Nuchter, and J. Hertzberg, "An autonomous mobile robot with a 3D laser range finder for 3D exploration and digitalization of indoor environments," *Robotics and Autonomous Systems*, Vol. 45, pp. 181-198, 2003.

[7] C. Zhou, Y.Wei, and T. Tan, "Mobile Robot Self-Localization Based on Global Visual Appearance Features," *Proc. IEEE Int. Conf. Robotics & Automation*, pp. 1271–1276, 2003.

[8] H.-S. LEE, et al., "Pedestrian Tracking Using GPS, Pedometer and Magnetic Compass," *Trans. of IEICE B*, Vol. J84-B, No. 12, pp. 2254–2263, 2001.

[9] T. Ishihara, N. Nisio, "A Positioning System Exclusively Utilizing GPS and Wireless Network Detecter," *IPSJ Tech. Report*, No. 2004-UBI-6-14, pp. 91–96, 2004.

[10] K. Yamazawa, Y. Yagi, M. Yachida, "Omnidirectional image sensor — HyperOmni Vision—," *Proc. 3rd Int. Conf. on Automation Technology*, Vol. 5, pp. 127–132, 1994.

[11] K. T. Simasarian, T. J. Olson, and N. Nandhakumar, "View-Invariant Regions and Mobile Robot Self-Localization," *IEEE Trans. on Robotics and Automation*, Vol. 12, No. 5, pp. 810–816, 1996.

[12] Y. Yagi, K. Imai, K. Tsuji, M. Yachida, "Iconic Memory-Based Omnidirectional Route Panorama Navigation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 27, No. 1, pp. 78–87, 2005.

[13] J. Rekimoto, A. Shionozaki, T. Sueyoshi, T. Miyaki, "PlaceEngine: A WiFi location platform based on realworld-folksonomy," *Internet Conference 2006*, pp. 95–104, 2006.