Vol.2020-HCI-190 No.9
Vol.2020-UBI-68 No.9
2020/12/8

スマートウォッチによるアクティブ・パッシブセンシングを用いた屋内位置ラベル推定

Thilina Dissanayake[1]　前川 卓也[1]　原 隆浩[1]　宮西 大樹[2]　川鍋 一晃[2]

Abstract: This study presents a novel approach for predicting the indoor location class of a smartwatch user, e.g., kitchen, bedroom, bathroom, by discovering location-specific sensor data motifs observed in sensor data collected from the smartwatch sensors. Specifically, we use acceleration data and audio impulse responses from the smartwatch to extract data segments that correspond to actions and acoustic characteristics that are specific to the different location classes using a novel matrix manipulation method. As an example, we can observe waveforms in acceleration data related to brushing actions only in bathrooms and also specific sound features because of their water-resistant walls. Our environment-independent location classifier does not use sensor data collected from the target environment or any handcrafted rules or templates to predict the location class. The proposed method is evaluated using 4 different household environments and achieve state-of-the-art performance.

Keywords: Indoor positioning, location class prediction, frequent pattern mining.

1. Introduction

Due to the increasing popularity of smart devices and advancements in sensors and, context recognition techniques, recognizing daily activities and estimating the indoor location of a user employing sensor data from his smart devices has been widely studied in the ubiquitous computing research community. Previous studies employ onboard inertial sensors of smart devices such as smartwatches and smartphones to recognize activities such as running, walking, and cleaning [1], [2], [3]. Furthermore, signaling techniques based on infrared, active acoustic sensing, Bluetooth, and Wi-Fi have been employed to estimate the indoor coordinates, i.e., the indoor location of a smart device user.

An important component in understanding a user's daily lifestyle is recognizing the room level indoor positioning of the user at a given time. This is because a user's daily activities have a strong correlation to the indoor location class. As an example, if the user's location is estimated as the bedroom, this prior knowledge can then be used to enhance the recognition of activities such as sleeping. Furthermore, recognizing the user's indoor location class can be incorporated into applications such as lifelogging.

This study focuses on recognizing room-level location classes of the user by employing acceleration and acoustic impulse response data recorded by onboard sensors of an off-the-shelf smartwatch. By employing the aforementioned data, we try to automatically capture sensor data inherent to each indoor location class.

2. Related work

Employing sensor data from the smartphone of the user is one of the most common methods that is currently being used to predict the location class of the user. Tarzia et al. [4] employed passive sound sensing to extract acoustic fingerprints from the background noise and attempted to locate a smartphone user. Azizyan et al. [5] employed multi-model sensor data (acceleration, image, Wi-Fi, light features, and acoustic features) to estimate the location labels to different stores such as Starbucks and Walmart. However, these methods require training data from the target environment. In contrast, our method can extract environment independent location-specific sensor data motifs from the training environments, hence not re-

[1] 大阪大学大学院情報科学研究科
[2] 株式会社 国際電気通信基礎技術研究所

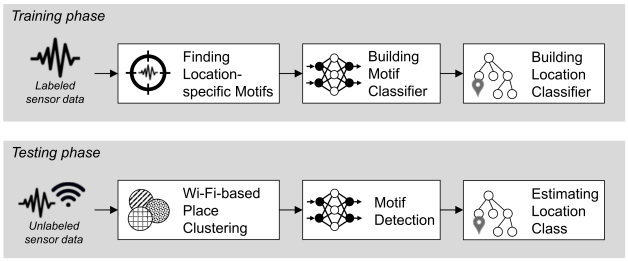


図 1: Overview of the proposed method

quiring training data from the target environment.

Tachikawa et al. [6] employed a modified random forest classifier to extract inherent sensor data from each location class and estimated room-level location class of the user. They also employed magnetometer, barometer, and microphone of the smartphone to extract location-specific features in six location classes in laboratory/office environments. In contrast to their method, our method does not assume that location-specific sensor data can be continuously observed in a target environment. Also, we employ inaudible sound waves to acquire location-specific acoustic data.

Elhamshary et al. [7] conducted a study that employed microphone, accelerometer, gyroscope, magnetometer, and barometer of a smartphone to extract location-specific features in nine assumed classes of train stations. However, this method relies on handcrafted recognition rules and sensor data templates tailored for each location class. In contrast, our method automatically extracts location-specific sensor data motifs and automatically constructs the location classifier.

3. Location class estimation method

3.1 Preliminaries

For this study, we assume that the user is wearing a smartwatch that is paired with a smartphone which is also carried by the user in his pocket. We use the smartphone to observe Wi-Fi Received Signal Strength Indicator (RSSI) data in the target environment at the same time collect acceleration and audio impulse response data from the smartwatch. Therefore, for each time slice, there exists a Wi-Fi scan, an acceleration data segment, and audio impulse response. We cluster the Wi-Fi scans to divide the locations into room level units. Furthermore, we employ acceleration data from the smartphone of the user to detect movement in between locations.

3.2 Overview

Figure 1 shows the overview of the proposed method. Our proposed method consists of two main phases; training phase and testing phase. During the training phase, we employ the data from the training environments, i.e., acceleration data and audio impulse responses to extract location-specific sensor data motifs using labeled training data. However, the training data from each environment and user is different from each other, because actions performed by each user is different from each other.

In order to address such problems related to environmental dependencies in the training data, we employ a domain-adversarial motif classifier to detect the occurrences of extracted motifs in the training data. Next, we use the detected motifs to train a location classifier.

In the testing phase, we collect sensor data as well as Wi-Fi RSSI data from a target user in a target environment. First, we cluster the Wi-Fi RSSI data into several place clusters, each corresponding to a different location the user visited in the target environment. Next, we detect the occurrences of the motifs in each Wi-Fi place cluster and then estimate a class label for each cluster using the trained location classifier.

3.3 Preprocessing

3.3.1 Acceleration data

In order to reduce the noise contained in acceleration data while preserving the most significant and unique waveforms corresponding to hand motions in data, we apply Principle Component Analysis (PCA) [8] and reduce the dimensionality of acceleration data from three dimensions to a single dimension. We then employ an overlapping time window to separate acceleration data into segments.

3.3.2 Impulse responses

An impulse is a signal that equals to one at time zero and is zero otherwise. Audio impulse responses can be employed to capture the acoustic characteristics of different environments. Impulse responses contain information related to sound propagation, hence, impulse responses can be used to capture information related to various environmental factors such as construction materials, abundant objects, and shape and the size of the space.

Our goal is to extract acoustic features that are inherent to each location class. As an example, when the user is in the bedroom, we can observe sound features related to mattresses, pillows, and other bedding while we observe features related to outside noise when the user is on the balcony. Therefore, we can expect to observe similar acoustic fingerprints in impulse responses recorded in the same location class.

In our method, we employ an impulse that sweeps the inaudible frequency range of 18 kHz and 20 kHz within 1 sec. Using the recorded signal, 12^{th} degree Mel Frequency Cepstral Coefficient (MFCC) features are extracted using a sliding time window. MFCC algorithm employs a scale that is more discriminative at lower frequencies and less discriminative at higher frequencies. Therefore, we modify and optimize the algorithm to fit our desired frequency range of 18 kHz - 20 kHz. In order to further enhance the hidden patterns in the acoustic features, we extract local binary patterns (LBP) from the MFCC features [9], [10].

3.4 Finding location-specific motifs

The process of discovering location-specific motifs consists of two main procedures: (i) similarity matrix calculation and (ii) calculation of location specificity measure (LSM) of each sensor data segment.

3.4.1 Similarity matrix calculation

In this step, the distance between each data segment collect by the same sensor modality is calculated. This allows us to detect similar data segments contained in the recorded sensor data. We first calculate the Euclidean distance between each pair of sensor data segments s_i and s_j, and arrange the distances into a distance matrix. Before the distance calculation, we standardize data within each data segment by subtracting the mean and dividing by the standard deviation. This makes the distance between data segments with low amplitude static noise to have larger values and the data segments with a similar waveform to have small distances.

Next, we normalize the distance matrix by dividing each element by the maximum value of the matrix and then we subtract each element from 1 to create a normalized similarity matrix. As we are mostly concerned with elements with higher similarity, we replace the elements in the similarity matrix that are less than a threshold with 0.0.

3.4.2 Calculating location specificity measure (LSM)

Next, we use the concept of Gini impurity to calculate the degree of specificity of each data segment s_i with respect to a location class. Gini impurity calculates the statistical dispersion using the following formula:

$$G = 1 - \sum_{i=1}^{C} p_i^2.$$

Here, p_i is the proportion of instances belonging to the i-th class and C is the number of classes. A lower Gini impurity means a smaller dispersion in instances. This idea can be used to calculate the specificity of each data setment to each location class.

First, we calculate p_i for a data segment, which represents the ratio of the segment occurring in i^{th} class, i.e., location. By using p_i, we can calculate the location specificity of each data segment based on the idea of Gini impurity.

Here, we assume that a similarity matrix $\mathbf{S} \in \mathbb{R}^{n \times n}$ and binary time-series of location labels with length n (i.e., ground truth labels) are given. The binary time-series $\mathbf{b}_c$ is prepared for each location class c where its element value at time t is 1 when the training user is at the location class at time t, which is defined as follows:

$$b_{c,t} = \begin{cases} 1 & \text{(the training user is at } c \text{ at time } t\text{)} \\ 0 & \text{(otherwise)} \end{cases}$$

For a row vector of $\mathbf{S}$ at each time slice t, i.e., $\mathbf{S}_{(t)}$, we compute $s_{c,t}$ as follows:

$$s_{c,t} = \frac{\mathbf{b}_c \cdot \mathbf{S}_{(t)}^{\mathrm{T}}}{\sum_{b_{c,t'} \in \mathbf{b}_c} b_{c,t'}}.$$

Because $\mathbf{S}_{(t)}$ is the similarity time-series for a time window (segment) at time t, $s_{c,t}$ shows the frequency of the occurrences of segments similar to the segment at place c. Note that we normalize the frequency by the duration of staying at place c, i.e., $\sum_{b_{c,t'} \in \mathbf{b}_c} b_{c,t'}$. Then, we compute $p_{c,t}$ as follows:

$$p_{c,t} = \frac{s_{c,t}}{\sum_{i=1}^{C} s_{i,t}}.$$

Therefore, $p_{c,t}$ shows the ratio of the occurrences of the data segment at time t for place c. Using $p_{c,t}$, we compute the LSM of the segment based on the idea of Gini impurity as follows:

$$\mathrm{LSM}_t = \sum_{i=1}^{C} p_{i,t}^2,$$

Note that a larger LSM value shows greater location specificity of the data segment. Figure 2 shows an example time-series of LSM values. As shown in the example, location-specific actions (e.g., eating in a dining room) have high LSM values. In addition, walking actions, which are observed in multiple locations, have low LSM values.

3.5 Building motif classifier

By using the aforementioned method, we obtain location-specific motifs from the training environments. Next, we build a classifier that detects the occurrences of location-specific motifs in the target environment for each

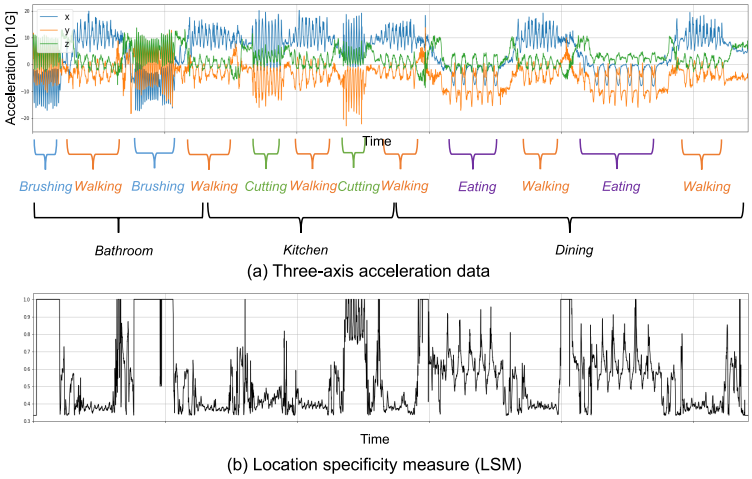


図 2: Example acceleration data and time-series of LSMs computed from the data

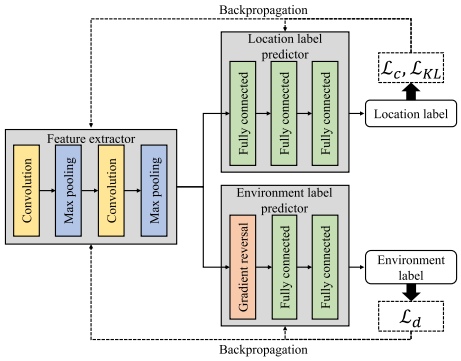


図 3: Structure of motif classifier based on domain-adversarial learning. $\mathcal{L}_c$, $\mathcal{L}_d$, and $\mathcal{L}_{KL}$ show the loss functions for location classification, domain classification, and distributions of output probabilities, respectively.

sensor modality. Here, a sensor data segment is an input to the motif classifier and the output is the location-specific motif class. Here, we are only interested in the sensor data segments that are specific to a certain location class, not recognizing fine-grained actions and gestures. We also define a class called "other" into which the none location-specific, i.e., the data segments with low LSM values are assigned to. As a result, the trained motif classifiers can ignore the outliers contained in the test data and mainly focus on the location-specific motifs.

Next, we train this classifier using the location-specific motifs extracted from the training environments. Note that the variability in sensor data segments in different environments has to be taken into consideration when training the classifier. In order to address environmental dependencies in data, we build the motif classifier based on domain-adversarial neural networks [11].

During the domain-adversarial training, we train the classifier (neural network) in such a way that it can classify the input instances into appropriate location classes but cannot classify the instances into their appropriate domain (environment). As our classifier is incapable of distinguishing features of different domains, we can consider the features as environment-independent.

Figure 3 shows the structure of the neural network used in this study. The neural network has three sections; the feature extractor, location label predictor, and the environment label predictor. Feature extractor is consisted of 1D convolution layers and is responsible for extracting domain-independent features from the sensor data segments. The location label predictor predicts the location class of each data segment based on the features extracted by the feature extractor. The environment label predictor predicts the relevant environment. i.e., source domain or target domain, of the data segment.

In order to make the motif classifier incapable of distinguishing between the domains, we introduce the gradient reversal layer that multiplies the gradient with a negative constant, making the feature extractor to extract features that can only recognize the classes, not the domain [11].

We train the network to minimize the following loss function using backpropagation based on Adam [12].

$$E(\theta_f, \theta_c, \theta_d) = \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}_c^i(\theta_f, \theta_c) - \lambda_1 \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}_d^i(\theta_f, \theta_d) - \lambda_2 \mathcal{L}_{KL}(\theta_f, \theta_c),$$

Here, $\theta_f, \theta_c, \theta_d$ are the network parameters of the feature extractor, location label predictor, and environment label predictor. n is the number of training instances. $\mathcal{L}_c^i(\theta_f, \theta_c)$, and $\mathcal{L}_d^i(\theta_f, \theta_d)$ are calculated cross entropy loss for the location label predictor and environment label predictor. $\mathcal{L}_{KL}(\theta_f, \theta_c)$ is the loss calculated based on the divergence of the classes output by the location label predictor. The input of our location label predictor is the output location probabilities averaged over each location class. Hence, when the distribution of these averaged vectors is distinguishable for each class, our location label predictor achieves higher accuracy. Therefore, we calculate the Kullback-Leibler divergence between the averaged vectors between each class and train the network in a way such that $\mathcal{L}_{KL}(\theta_f, \theta_c)$ is maximized.

3.6 Location classifier

Here, we construct a classifier that can predict the location class of the place cluster in a target environment. We base the estimation of location class of the place clusters on the occurrence frequencies of location-specific motifs,

acquired by the motif classifier. The input of the location classifier is the occurrence ratios of location-specific motif classes and the output is the location class. Here, we explain the process of calculating the location cluster-wise motif occurrence ratios. First, using the motif classifier, we acquire the output probabilities of each segment in each location cluster. Note that the output of the motif classifier is a probability vector with a dimension of $C + 1$, C being the number of location classes. Next, we take the average of the output of the vectors by the motif classifier over the data segments in each location cluster. When employing multi-model sensor data, we calculate the probability vectors and average them for each vector modality separately. Next, we concatenate the probability values of each sensor modality for each location class to form a single probability vector. In the case of employing s sensor modalities, the dimension of this vector becomes $s \times (C + 1)$. Note that the training data for the location classifier is limited as we are averaging the class probabilities of each data segment in each location cluster. Due to this limitation, we employ the nearest centroid classifier [13] as the location classifier.

3.7 Wi-Fi based place clustering

During the testing phase, we cluster the Wi-Fi RSSI data collected using the smartphone of the user. Each Wi-Fi scan consists of the MAC address of the access points (APs) that are detected by the device and the received signal strengths from them. As the scans of the same place are similar, we can cluster the Wi-Fi scans into place clusters [6].

3.8 Estimating location class

Finally, we estimate the location label of each location cluster in the target environment. Here, we feed the data segments acquired from the target environment into the motif classifier and obtain the probability vectors. Next, we average the vectors within each Wi-Fi location cluster. These averaged vectors are then fed into the location classifier and the output is the estimated location class of the location cluster.

4. Evaluation

4.1 Dataset

In order to evaluate our method, we collected data from four different environments (Figure 4), where a different participant collected data from each environment. ASUS ZenWatch3 smartwatch was attached to their dominant wrists. We collected data from seven location classes (kitchen, dining room, restroom, washstand, den, smoking area (outdoor resting place), and bedroom), hence, this problem can be considered as a seven-class classification problem. We observed a semi-naturalistic collection protocol [1] that serves a greater variability in participant behavior compared to the laboratory data. Table 1 shows a list of the activities, where each participant performed in a random sequence, at each location. The smoking areas are outdoor places, e.g., balcony, veranda, therefore outside noises are captured by the microphone of the smartwatch. Furthermore, noises of running water were captured by the microphone when the participant was washing hands, dishes, and face.

表 1: List of locations and specific actions performed

Location	Activity
kitchen	chop, wash dishes, wash hands
washstand	brush teeth, wash face, wash hands
bedroom	lie
toilet	sit, use toilet paper, wash hands
dining room	eat, drink, sit
den	use PC, write on a notebook, sit
smoking area	smoke

表 2: Physical features of participants

	dominant hand	height	sex	age	sessions
1	right	175cm	male	26	6
2	right	177cm	male	27	7
3	right	168cm	male	30	8
4	right	n/a	female	39	7

Table 2 shows the list of physical features of the participants and the number of data collection sessions done by each participant. The acceleration data was collected at the sampling rate of 30 Hz and the audio data was collected at 44.1 kHz sampling rate. We also collected video data to obtain ground truth.

4.2 Evaluation methodology

We evaluated our method using "leave-one-environment-out" cross-validation where sensor data from one environment is employed as the test data and the data from the remaining environments were treated as the training data. We predicted the location class of each session. In order to compare the effectiveness of the proposed method we also prepared the following methods.

- **Proposed**: This is our proposed method that employs acceleration data and impulse responses col-

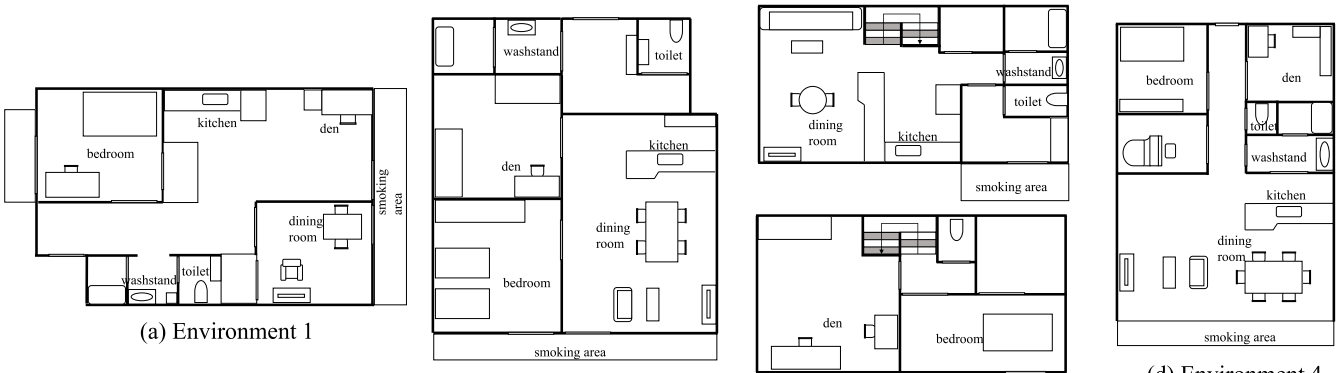


図 4: Experimental environments

lected using active probing.

- **RF-ACC**: This method only employs acceleration data and is designed based on a previous study that assumed location-specific features are always observed at a location of interest. Therefore, this method assumes a sliding time window and extracts sensor data features within that window. Then, this method forms a feature vector concatenating the feature values, which is input to the random forest (RF) classifier, which classifies a feature vector into a location class. The extracted features are the minimum, maximum, mean, standard deviation, etc. using the tsfresh library (v 0.12.0).
- **RF-MIC**: This method only employs impulse responses. The procedures of this method are identical to those of RF-ACC.
- **RF-C**: This method employs acceleration data and impulse responses. This method first clusters Wi-Fi scans in the same manner as the proposed method. This method then aggregates the location class estimation results for time windows in each location cluster by majority vote to determine the location class of the cluster.
- **Only-ACC**: This is a variant of the proposed method that only employs acceleration data.
- **Only-MIC**: This is a variant of the proposed method that only employs impulse response.
- **DANN**: This is a variant of the proposed method that does not employ the loss function for distributions of output probabilities in the motif classifier (i.e., using only $\mathcal{L}_c$ and $\mathcal{L}_d$).
- **CNN**: This is a variant of the proposed method that does not employ domain-adversarial training and the loss function for distributions of output probabilities in the motif classifier (i.e., using only $\mathcal{L}_c$).

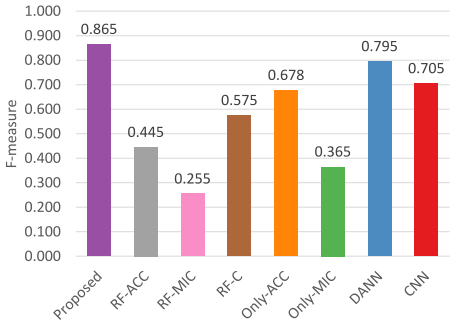


図 5: Average F-measures of the methods

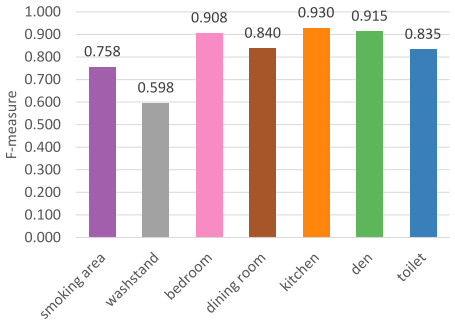


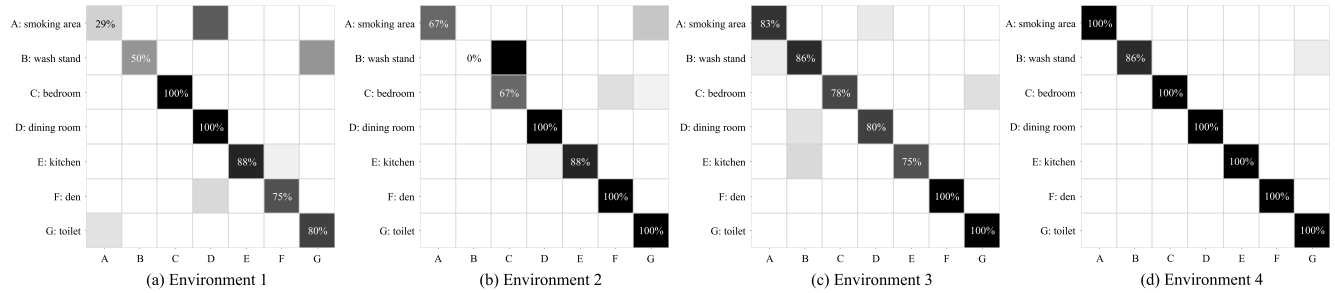
図 6: Average F-measures of Proposed for each location class

The classification accuracy was evaluated by micro-averaged F-measure of the location class predictions made for each location cluster. For RF-ACC and RF-MIC methods, these metrics were calculated based on the classification results of the time windows.

4.3 Results

4.3.1 Classification accuracy

Figure 5 shows the F-measures for the proposed method averaged over all environments. As can be seen, our proposed method achieved an average F-measure of 86.5% even the test data from the target environment is not used. Figure 6 shows the average accuracy of the proposed method for all the location classes. Figure 7 shows



(a) Environment 1 (b) Environment 2 (c) Environment 3 (d) Environment 4

図 7: Confusion matrices of classification results of Proposed

the visual confusion matrices for the proposed for all the environments. Note that the accuracy of the washstand class is poorer than that of the other classes. As the user washed the hands at both the washstand and the toilet, several washstand instances were incorrectly classified into the toilet class. The tooth brushing action was somewhat different between each participant. This can be the reason for some washstand instances being misclassified into the bedroom class where there are only little distinguishable hand motions.

The accuracy of the smoking area class is also somewhat poor. Several smoking area instances were incorrectly classified into the dining room class (Figure 7). This is because our method could not distinguish between location-specific actions such as smoking and eating in these locations. Furthermore, acquiring meaningful acoustic characteristics from the balcony class was difficult in some environments due to ambient noises.

4.3.2 Comparison with prior methods

Figure 5 also shows the average F-measures of RF-ACC, RF-MIC, RF-C methods. Note that the RF-ACC and RF-MIC methods are based on the assumption that location-specific motifs can continuously be observed at the location of interest. However, even though the participants performed the location-specific actions at each location, these methods could not accurately recognize the location based on those actions as they only lasted for a very short amount of time. Furthermore, some activities such as “wash hand” were performed in multiple locations (Table 1), and these methods could not distinguish between those locations accurately.

Moreover, the RF-C method which aggregates the window-wise location class estimations within each place cluster and predicts the location class of the cluster based on majority vote, could not recognize the location classes with high accuracy as the results of RF-C is based on the poor results of RF-ACC and RF-C methods.

表 3: Classification F-measures in each environment

	Env.1	**Env.2**	**Env.3**	**Env.4**
Proposed	0.770	0.830	0.880	0.980
RF-ACC	0.490	0.380	0.440	0.470
RF-MIC	0.240	0.200	0.220	0.360
RF-C	0.640	0.510	0.570	0.580
Only-ACC	0.590	0.760	0.710	0.650
Only-MIC	0.400	0.200	0.430	0.430
DANN	0.720	0.820	0.670	0.970
CNN	0.670	0.600	0.720	0.830

4.3.3 Classification accuracy in each environment

Table 3 shows the F-measure values of the methods for each environment. Our method achieves good results in all the environments. However, the result of Environment 1 is marginally poor compared to the other Environments. As can be seen in Figure 7, several smoking area instances, and washstand instances are misclassified into the dining room and toilet classes respectively.

4.3.4 Sensor contribution

Figure 5 also shows the F-measures of the Only-ACC and Only-MIC methods. Between the accelerometer and the microphone sensor modalities, the accelerometer was the best contributor to the accuracy. Only-ACC achieved an average F-measure around 68%. However, the accelerometer alone could not achieve good results because the actions of different participants are different from each other. Furthermore, locations such as a bedroom that has very little location-specific actions, were specifically hard to predict using Only-ACC method.

4.3.5 Contribution of motif classifier

Figure 5 also shows the average F-measure of DANN and CNN methods. The proposed method outperforms DANN by 7% and DANN outperforms CNN by 9%. Especially, the F-measures of smoking area and toilet classes were improved by introducing the loss function that was calculated for the distributions of output probabilities, i.e., $\mathcal{L}_{KL}$. By introducing the domain adversarial train-

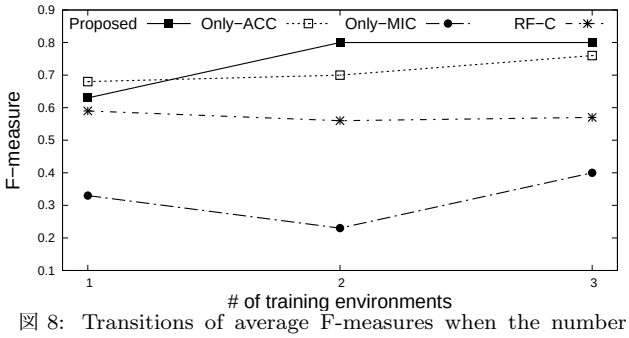


図 8: Transitions of average F-measures when the number of training sessions is varied

ing, accuracies of the toilet and bedroom classes were improved because they contained the most environmental-dependent acoustic features.

4.4 Discussion

4.4.1 Amount of training data

Figure 8 shows the transitions of average F-measure for several methods when the number of training environments was changed. As shown in the figure, even though the number of training environments is increased, the F-measure for RF-C method did not improve, showing the inability of capturing location-specific features using this method. In contrast, with only two training environments, the proposed method exceeds the average F-measure of 80%.

4.4.2 Energy consumption

In order for our method to work, the accelerometer, speaker, and microphone of the smartwatch should always be on. Under this condition, ASUS ZenWatch3 could function up to approximately 3 hours When only the accelerometer was switched on, the battery life was approximately 4.5 hours. To reduce the battery consumption, we propose to enable sound sensing only when the hand movement of the user is minimum as we do not use the impulse responses recorded when the hand of the participant was moving.

5. Conclusion

In this study, we employed multi-model sensor data from the smartwatch of a user to extract location-specific sensor data to predict his indoor location class. We also proposed a novel matrix manipulation method that can automatically extract location-specific time-series sensor data by calculating a score known as Location Specificity Measure based on the idea of Gini impurity. To the best of our knowledge, this is the first study that introduces Gini impurity to extract class-specific sensor data templates. We evaluated our method in real household environments and achieved state-of-the-art-performance. As a part of our future work, we intend to evaluate our method in working environments such as factories.

謝辞 This work is partially supported by JST CREST JPMJCR15E2, JSPS KAKENHI Grant Number JP16H06539 and JP17H04679.

参考文献

[1] Bao, L. and Intille, S. S.: Activity recognition from user-annotated acceleration data, *Pervasive 2004*, pp. 1–17 (2004).
[2] Lu, H., Pan, W., Lane, N. D., Choudhury, T. and Campbell, A. T.: SoundSense: scalable sound sensing for people-centric applications on mobile phones, *MobiSys 2009*, pp. 165–178 (2009).
[3] Lukowicz, P., Ward, J. A., Junker, H., Stäger, M., Tröster, G., Atrash, A. and Starner, T.: Recognizing workshop activity using body worn microphones and accelerometers, *Pervasive 2004*, pp. 18–32 (2004).
[4] Tarzia, S. P., Dinda, P. A., Dick, R. P. and Memik, G.: Indoor localization without infrastructure using the acoustic background spectrum, *MobiSys 2011*, pp. 155–168 (2011).
[5] Azizyan, M., Constandache, I. and Roy Choudhury, R.: SurroundSense: mobile phone localization via ambience fingerprinting, *MobiCom 2009*, pp. 261–272 (2009).
[6] Tachikawa, M., Maekawa, T. and Matsushita, Y.: Predicting location semantics combining active and passive sensing with environment-independent classifier, *UbiComp 2016*, pp. 220–231 (2016).
[7] Elhamshary, M., Youssef, M., Uchiyama, A., Yamaguchi, H. and Higashino, T.: TransitLabel: A crowd-sensing system for automatic labeling of transit stations semantics, *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*, ACM, pp. 193–206 (2016).
[8] Pearson, K.: Principal components analysis, *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, Vol. 6, No. 2, p. 559 (1901).
[9] Ojala, T., Pietikainen, M. and Harwood, D.: Performance evaluation of texture measures with classification based on Kullback discrimination of distributions, *Proceedings of 12th International Conference on Pattern Recognition*, Vol. 1, IEEE, pp. 582–585 (1994).
[10] Yang, W. and Krishnan, S.: Combining temporal features by local binary pattern for acoustic scene classification, *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, Vol. 25, No. 6, pp. 1315–1321 (2017).
[11] Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M. and Lempitsky, V.: Domain-adversarial training of neural networks, *The Journal of Machine Learning Research*, Vol. 17, No. 1, pp. 2096–2030 (2016).
[12] Kingma, D. and Ba, J.: Adam: A method for stochastic optimization, *arXiv preprint arXiv:1412.6980* (2014).
[13] McIntyre, R. M. and Blashfield, R. K.: A nearest-centroid technique for evaluating the minimum-variance clustering procedure, *Multivariate Behavioral Research*, Vol. 15, No. 2, pp. 225–238 (1980).