Sensors and Materials, Vol. 33, No. 1 (2021) 35–51
MYU Tokyo

S & M 2436

Smartphone-based Estimation of Sidewalk Surface Type via Deep Learning

Satoshi Kobayashi* and Tatsuhito Hasegawa

Graduate School of Engineering, University of Fukui, 3-9-1, Bunkyo, Fukui, Fukui 910-8507, Japan

(Received July 13, 2020; accepted October 22, 2020)

Keywords: smartphone, accelerometer, sidewalk surface, deep learning

In this study, we develop a method of estimating the type of sidewalk surface on which a user walks using three-axis acceleration sensor data measured by a user's smartphone's accelerometer. If the shape of the sidewalk surface can be estimated automatically, various sidewalk information, such as walkable sidewalks and sidewalks where pedestrians cannot easily walk, can be collected by simply having many users carry their smartphones. We, therefore, propose a method of estimating the sidewalk surface type by applying a convolutional neural network (CNN) based on the VGG16 architecture to sensor data. In addition, we combined VGG16 with hand-crafted features (HCFs) validated in preliminary experiments. During training, the model was pretrained with the human activity sensing consortium (HASC) dataset, a large benchmark for human activity recognition, as a source domain, and we applied fine-tuning (FT) for the sidewalk surfaces as a target domain. We conducted experiments on seven subjects and evaluated the accuracy of our proposed method using leave-one-subject-out cross-validation (LOSO-CV). The experimental results showed that our proposed method achieved the highest accuracy among all the compared methods. Specifically, our proposed method improved the accuracies of some subjects by more than 20% compared with the baseline method.

1. Introduction

In recent years, owing to the spread of smartphones, various sensors have become readily available, and it has become possible to collect a large amount of sensor data. There are many studies that use sensor data measured by smartphones. For example, there have been studies that estimate user activity[(1)] and the slope of the ground.[(2)] One study also estimated barriers, such as slopes and steps, from acceleration data.[(3)] Barriers in cities refer to mobility obstacles for people with disabilities and the elderly. Knowing the locations of such barriers not only helps these groups avoid them but also encourages road and facility managers to make improvements.

There have been few studies estimating the surface of sidewalks. A sidewalk surface information map, including the locations of gravel sidewalks, can help people with disabilities and the elderly locate barriers. In addition, it is convenient to know the condition of a sidewalk surface in advance when going about daily life. For example, if cyclists know the location of

*Corresponding author: e-mail: mf200471@u-fukui.ac.jp
https://doi.org/10.18494/SAM.2021.2976

ISSN 0914-4935
https://myukk.org/

gravel roads in advance, they can avoid them. Similarly, if sidewalk surface information for walkable sidewalks and sidewalks difficult for pedestrians (e.g., due to snow) can be collected automatically, we can apply the results to navigation systems to automatically choose the most convenient route. Therefore, it is useful to share various types of sidewalk surface information.

In this study, we develop a method that estimates the sidewalk surface type from acceleration data collected from a smartphone's acceleration sensor. This widely used smartphone approach is suitable for collecting sidewalk information in a large area. In addition, motion sensor data, such as acceleration data, are suitable for estimating the condition of sidewalk surfaces. In using this method, sidewalk surface information can be stored by users who carry a smartphone while walking. Rather than classifying the type of sidewalk surface, there are ways to evaluate it on the basis of its walkability. For example, Lee *et al.*[4] used biosensors to classify barriers from stress. There is also a social networking service in which users manually enter and share whether they have been able to pass in a wheelchair.[5] However, the ease of passage varies for pedestrians and cyclists as well as young and old individuals. In addition, asking users to manually enter sidewalk surface information can be a major hassle. By estimating the sidewalk surface type using a smartphone's accelerometer, it is possible to collect information objectively and automatically.

2. Related Work

2.1 Sensor-based sidewalk surface estimation

Fujii *et al.*[6] proposed a method of estimating flat ground, bumps, and slopes by extracting 57-dimensional features from the acceleration data step-by-step using a support vector machine (SVM).[7] Miyata *et al.*[3] proposed a method of estimating stone pavements, crowds, steps, stairs, and open doors in addition to flat ground, bumps, and slopes from three-axis acceleration data measured by a small acceleration sensor as large as a smartphone. Their method employs 33-dimensional features, henceforth referred to as hand-crafted features (HCFs), obtained by an SVM. They also employed a denoising autoencoder (DAE) using acceleration data measured in the subject's daily life and compared the estimation accuracy between the use of classical HCFs and feature maps extracted by a DAE. The result of such a comparison showed that effective features differ depending on the estimation target.

Takahashi *et al.*[8] proposed a method of detecting bumps from three-axis acceleration data measured by cyclists' smartphones in their pants pockets or bags. They used independent component analysis (ICA) to separate cycling motion and road signals from acceleration data and estimate road bumps using a read-signal mother wavelet (RMW). Hoffman *et al.*[9] proposed a method of detecting bumpy roads using acceleration data obtained from a smartphone mounted on a bicycle.

Jain *et al.*[10] developed a system that alerts pedestrians entering an intersection by detecting small slopes and steps using three-axis acceleration data and three-axis gyroscope data (by the rule-based method they defined) from accelerometers and gyroscopes mounted on shoes. Ohashi *et al.*[11,12] proposed a method of estimating tactile tiles from pressure on a foot pressure sensor.

Nunes and Mota[13] developed a system that collects asphalt conditions from acceleration data from a user's smartphone in a car. They estimated road conditions using machine learning (such as random forest, SVM, k-nearest neighbor algorithm, and J48) with HCFs extracted from acceleration data. Nomura and Shiraishi[14] proposed a method of estimating road surface conditions from acceleration data from a smartphone placed on the dashboard of a car. Their method detected changes in road conditions by comparing the latest estimation results with past results.

In our previous study,[15] we proposed a method using a random forest[16] as a preliminary study. We targeted six types of sidewalk: asphalt, gravel, lawn, grass, sand, and a mat. Asphalt is a common sidewalk on which pedestrians find it easy to walk. A lawn is a sidewalk with short grass that is maintained. Grass refers to a sidewalk with long grass that is difficult to walk through. The mat called Evermat[17] is the type used in artistic gymnastics. We adopted the mat to imitate a firmly packed snowy sidewalk. The reason for choosing these six sidewalks is that they are surfaces that are likely to make a difference in walking and are likely to be encountered by pedestrians in real life. We used a random forest with 177-dimensional HCFs extracted from acceleration data.

2.2 Sensor-based human activity recognition via deep learning

Sensor-based human activity recognition is a research field similar to sensor-based sidewalk surface estimation. There are many studies that apply a convolutional neural network (CNN) for human activity recognition using motion sensor data such as acceleration data.[18–20] Hasegawa and Koshino[21] verified the effectiveness of various CNN models using the human activity sensing consortium (HASC) dataset,[22] which is the benchmark for human activity recognition. Owing to the availability of such large datasets, there have been many studies on the application of deep learning in human activity recognition using sensor data. Although there are many studies in the field of activity recognition, few studies have addressed sidewalk surface estimation, and there are no examples of deep learning applied to sidewalk surface estimation tasks. In addition, CNNs applied to the field of activity recognition often have a shallow architecture.

2.3 Our contributions

From the above information, the following are the main contributions of this study:

- Propose a CNN-based sidewalk surface type estimation method. Conventional methods employ machine learning methods with HCFs. In this study, we propose a new architecture model based on VGG16 combined with HCFs validated in preliminary experiments. In addition, we clarify the effectiveness of fine-tuning (FT), which is a type of transfer learning method, using the HASC human activity recognition dataset.
- Investigate the limits of the sidewalk surface type that can be estimated from smartphone acceleration sensor data. We conducted an experiment to collect the sensor data measured while subjects are walking on various sidewalk surface types, including roads not easily

detected using acceleration sensor data. We classify sidewalk surfaces that are not targeted in related studies and clarify how and to what extent a sidewalk surface type can be estimated from the data measured by a smartphone.

- Verify the effectiveness of the proposed method. We conducted an experiment with seven subjects to measure the sensor data of six types of sidewalk, as shown in Fig. 1: asphalt, gravel, lawn, grass, sand, and a mat. As a result, we found that VGG16 with FT and HCFs significantly improves the estimation accuracy compared with baseline methods (including those proposed in related studies). We also considered the amount of detail our method can detect from the surface type.

3. Materials and Methods

3.1 Outline of our proposed method

Figure 2 shows the outline of our proposed method. The user, who provides the sensor data, only needs to put the sensor-activated smartphone in his or her pocket before starting to walk. The measured three-axis acceleration data is divided into fixed windows. The obtained window of the raw acceleration data is inputted to the CNN. In addition, we use HCFs extracted for each window. For the HCFs, we adopted those used as random forest inputs in our previous study[15] (details described in Sect. 4.3) We concatenate the extracted HCFs and the output of the CNN feature extractor (encoder) to input a fully connected layer of the classifier to estimate the six types of sidewalk surface studied here.

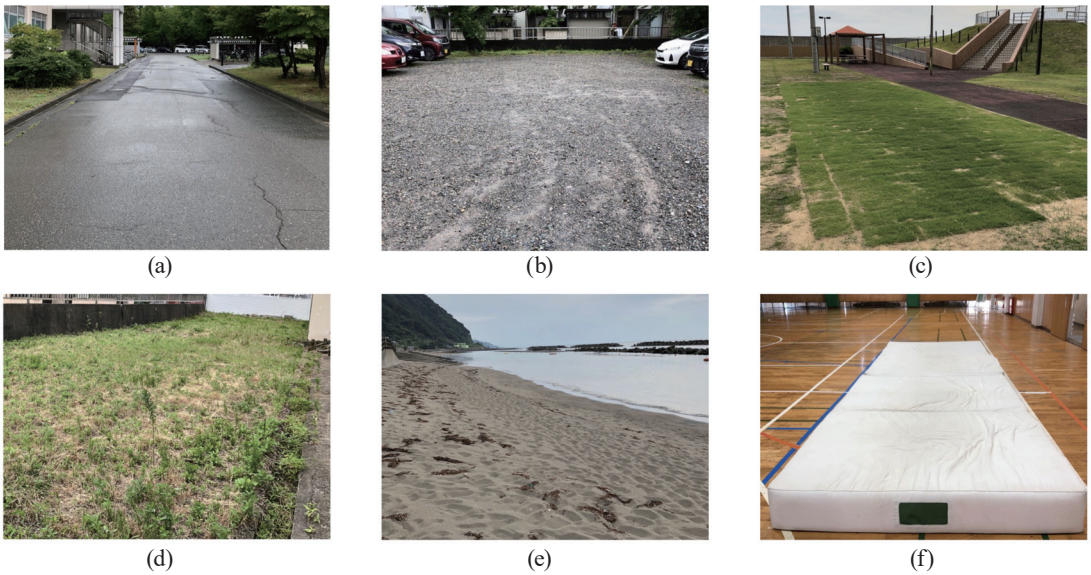

Fig. 1. (Color online) Six types of estimation target: (a) asphalt, (b) gravel, (c) lawn, (d) grass, (e) sand, and (f) mat.

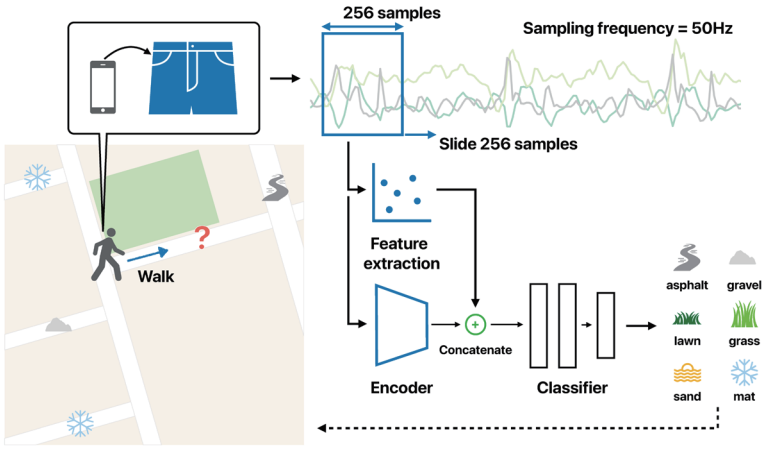

Fig. 2. (Color online) Outline of our proposed method.

Our proposed method can visualize the condition of a sidewalk surface on a map using GPS data, as shown on the left side of Fig. 2. Particularly in this paper, we describe a method of estimating the sidewalk surface type using acceleration sensor data measured by a smartphone while the user is walking.

The following sections are structured as follows: we describe the model architecture of the encoder in Sect. 3.2, how to combine the HCFs with the encoder in Sect. 3.3, and the training method of the whole model with transfer learning in Sect. 3.4.

3.2 CNN model

Table 1 shows the model architecture of the encoder part of our method, and Table 2 shows that of the classifier part. Conv1D indicates the convolutional layer, and MaxPooling1D indicates the pooling layer. Flatten is a layer that converts 2D vectors from the convolutional layer into 1D vectors. FC indicates the fully connected layer. The Softmax function is used as the activation function of the output layer. All the dropout rates of each dropout layer are set to 0.5. Concatenate is a layer that combines the output of the encoder with HCFs. *N* in the "Output Channels" column indicates the number of HCFs to be combined. Hasegawa and Koshino[21] found that the VGG16[23] model is easy to implement yet achieves a high estimation accuracy in human activity recognition. Therefore, we adjusted the parameters of this model based on VGG16. VGG is a well-known model in the image recognition field that placed second in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2014. The 16 in VGG16 means that there are 16 convolutional and fully connected layers in total.

3.3 How to combine the encoder output with HCFs

In this study, we propose a method of combining HCFs designed on the basis of human knowledge with the CNN-based feature representations from the VGG16 described in Sect. 3.2.

Table 1
Architecture of encoder.

Layer type	Number of filters	Shape of output	Output channels
Conv1D	16	768	16
Conv1D	16	768	16
MaxPooling1D	2	384	16
Conv1D	32	384	32
Conv1D	32	384	32
MaxPooling1D	2	192	32
Conv1D	64	192	64
Conv1D	64	192	64
Conv1D	64	192	64
MaxPooling1D	2	96	64
Conv1D	128	96	128
Conv1D	128	96	128
Conv1D	128	96	128
MaxPooling1D	2	48	128
Conv1D	128	48	128
Conv1D	128	48	128
Conv1D	128	48	128
MaxPooling1D	2	24	128
Flatten	—	—	3072
FC	—	—	1024

Table 2
Architecture of classifier.

Layer type	Number of filters	Shape of output	Output channels
Concatenate	—	—	1024 + *N*
FC	—	—	1024
Dropout	—	—	1024
FC	—	—	1024
Dropout	—	—	1024
Softmax	—	—	6

We use the 177-dimensional HCFs from our previous study[15] (details described in Sect 4.3). Decision tree algorithms, such as the random forest used in our previous study, can calculate the importance of each feature computed by considering the Gini impurity. In this study, we adopted *N* selected HCFs in order of importance. We then compared the estimation accuracy of the additional HCFs with one to 15 features.

The output of Flatten, which converts the final pooling layer of VGG16 into a 1D vector, is 3072 dimensions. Since the HCFs are at most 177 dimensions, the effect of the VGG16 feature map is likely to be greater than that of the HCFs. Therefore, the Flatten output is transformed to 1024 dimensions by adding a fully connected layer before combining the HCFs.

3.4 Transfer learning

In our proposed method, we apply transfer learning with human activity recognition as the source domain to train the deep learning model described above. Transfer learning is a technique of adapting a model trained in one domain to another domain. In the field of image recognition, many studies have applied transfer learning based on ImageNet,[24] a large dataset used in ILSVRC as a source domain. Some studies have tried to apply transfer learning to tasks using sensor data.[25,26]

In this study, we verify the effectiveness of transfer learning with human activity recognition as the source domain to estimate the sidewalk surface type as the target domain (see Fig. 3 for

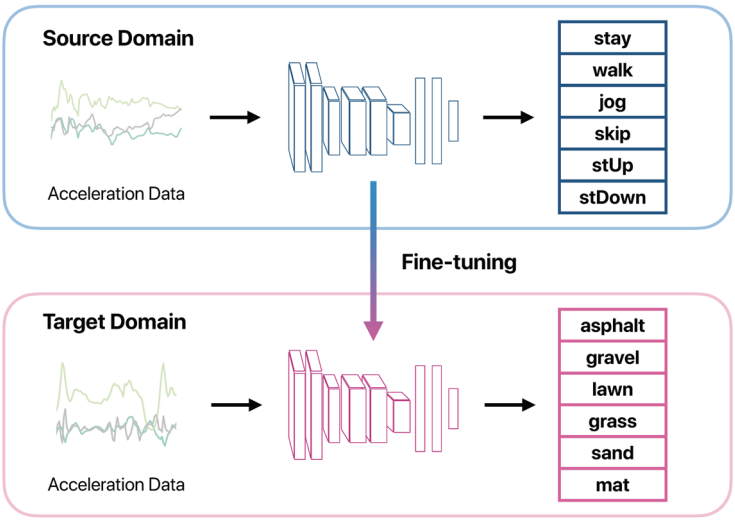

Fig. 3. (Color online) Outline of transfer learning in this study.

an outline of transfer learning). Since the input of both domains is three-axis acceleration data, it is possible to acquire feature representations useful in improving the accuracy of sidewalk surface estimation, which is the target domain. There are two types of transfer learning method: feature extraction and FT. Feature extraction reuses the feature maps of the pretrained models in a target domain. In addition, the weights of the pretrained models are frozen, and a classifier is newly trained. FT is a method in which the weights of the pretrained model are used as the initial weights of a target domain. In this study, we adopt FT as the transfer learning method. Additionally, we transfer weights from the human activity recognition model to the encoder part only, and the classifier part is newly trained.

We used the HASC dataset[22] to train the human activity recognition model, and we used 151 subjects' acceleration data collected at a sampling frequency of 100 Hz in the HASC dataset. We divided each measurement data into a window width of 256 and a stride width of 256. The estimation targets were six types of basic activity: standing (stay), walking (walk), jogging (jog), skipping (skip), going up stairs (stUp), and going down stairs (stDown).

4. Experiments

4.1 Data collection

We used seven subjects' acceleration data collected in our previous study[15] and labeled with each type of sidewalk surface. The acceleration data was collected at a sampling rate of 50 Hz. The collected acceleration data was then divided into a window width of 256 and a stride width of 256. The subjects stored their smartphones in their front right pockets of their pants, with the smartphone's screen facing inward to measure the sensor data. The subjects walked for 1 min in each session, and five sessions were measured for each type of sidewalk surface per subject. After 1 min of walking, the subjects removed their smartphones from their pockets. It took a

total of 30 min for each subject to collect the acceleration data, and about 210 min of sensor data (2519 samples) was collected.

In the 1 min of walking, the subject walked only in a straight line on an asphalt sidewalk and on sand. However, on other sidewalks, the subject made U-turns or curves, because these sidewalks were not sufficiently long to enable walking straight for 1 min. Data containing U-turns and curves can have negative effects, such as reduced accuracy. However, U-turns and curves are behaviors that can occur in daily life, and we verify the accuracy of the data including their negative effects. The subjects walked with their shoes removed when walking on the mat. Originally, the measurement was meant to be performed on actual snow. However, the experiment was conducted in the summer and not during the snow season in Japan. We considered alternative materials that are closer to the feeling of walking on the snowy sidewalk through asking people; therefore, we selected Evermat[17] as the alternative to a snowy sidewalk.

The subjects' devices and clothes varied (see Table 3). Some research has shown data measurement performed using the same devices and clothes. However, we consider that this is not appropriate for evaluating the estimation accuracy of the proposed method in a real environment, because not everyone wears the same clothes in reality. In this study, we did not impose a set of conditions in order to determine the estimation accuracy of a real-life situation.

4.2 Evaluation method

In order to verify the effectiveness of the proposed method, we adopted accuracy as an evaluation index because the number of data for each class was not significantly biased. We used leave-one-subject-out cross-validation (LOSO-CV) as the evaluation method. LOSO-CV involves using one subject as the test data and the remaining subjects as training data, and replaces the test subject to test all the subjects.

4.3 Baseline methods

As baseline methods, we used the 33-dimensional HCFs proposed by Miyata *et al.*,[3] the 57-dimensional HCFs proposed by Fujii *et al.*[6] (which were used in barrier detection), and the 177-dimenstional HCFs proposed in our previous study.[15] We used a simple CNN model similar to that adopted in related work[19] as a baseline CNN model.

Table 3
Experimental conditions for each subject.

ID	Age	Gender	Device	Bottoms	Shoes
A	22	Male	iPhone 8	Jogging pants	Sneakers
B	23	Male	iPhone 8	Steteco (loose shorts)	Sandals
C	22	Male	iPhone 5s	Jeans	Sneakers
D	21	Male	iPhone 5s	Jeans	Sneakers
E	21	Male	iPhone 5s	Jeans	Sneakers
F	21	Male	iPhone 8	Jeans	Sneakers
G	21	Male	iPhone 8	Cargo pants	Sneakers

Since both studies on barrier detection[3,6] used an SVM as the machine learning method, we reproduced the methods using an SVM in this experiment. In our previous study, we used a random forest with the HCFs extracted from three-axis acceleration data, as shown in Table 4. We investigated whether the methods proposed in related studies work effectively to estimate the six types of sidewalk surface. Because the target surfaces differ from those in the original studies,[3,6] the results of this article could differ from those in the original studies as well.

We describe the HCFs in Table 4. Root mean square[27] is the square root of the mean of the square values. Intensity[28] is defined by Eq. (1), where the sample size is n and the sensor values are $X(i)$ (i: 1, 2, 3, ..., n).

$$Intensity = \frac{1}{n-1}\sum_{i=1}^{n-1}\left|X(i)-X(i+1)\right| \tag{1}$$

Skewness and kurtosis[29] are respectively given by Eqs. (2) and (3), where the sample size is n, the sensor values are $X(i)$ (i: 1, 2, 3, ..., n), the mean of the sensor values is $\bar{X}$, and the sample standard deviation is u.

$$Skewness = \frac{n}{(n-1)(n-2)}\sum_{i=1}^{n}\left(\frac{X(i)-\bar{X}}{u}\right)^3 \tag{2}$$

$$Kurtosis = \frac{n(n+1)}{(n-1)(n-2)(n-3)}\sum_{i=1}^{n}\left(\frac{X(i)-\bar{X}}{u}\right)^4 - \frac{3(n-1)^2}{(n-2)(n-3)} \tag{3}$$

The zero-crossing rate[30] is the rate of the number of times the average and observed values intersect. For the frequency domain, we subjected the frame to a fast Fourier transform (FFT) and calculated the same feature values in the all-frequency, low-frequency, mid-frequency, and high-frequency regions. The low-frequency region is 0–4.2 Hz, the mid-frequency region is 4.2–8.4 Hz, and the high-frequency region is 8.4–12.6 Hz. Henceforth, we will refer to the

Table 4
List of HCFs proposed in our preliminary experiment.

Time domain		Frequency domain
Mean	Interquartile range	Maximum
Mean of absolute values	Correlation coefficient between axes	Frequency of maximum
Standard deviation	Correlation coefficient of absolute values between axes	2nd maximum
Standard deviation of absolute values	Initial value in the frame	Frequency of 2nd maximum
Minimum	Final value in the frame	Standard deviation
Maximum	Intensity	1st quartile
Root mean square	Skewness	Median
1st quartile	Kurtosis	3rd quartile
Median	Zero-crossing rate	Interquartile range
3rd quartile	—	Correlation coefficient between axes

method using Miyata *et al.*'s features and the SVM as SVM-A, the method using Fujii *et al.*'s features and the SVM as SVM-B, and our previous study as RF. The baseline CNN model consists of four alternating layers of convolutional and pooling layers. Finally, there are five baselines: the aforementioned SVM-A, SVM-B, and RF, the four-layer CNN without transfer learning (CNN), and the four-layer CNN with transfer learning (CNN+FT). Our proposed methods are called VGG16, VGG16 with transfer learning (VGG16+FT), and VGG16+FT combined with HCFs (VGG16+FT+HC).

All the CNN models were optimized using Adam.[31] We used the rectified linear unit (ReLU) function[32] as the activation function of convolutional and fully connected layers. As a result of the search, the learning rate was set to 1.0×10^{-3} to pretrain the activity recognition model. Additionally, the learning rate was set to 1.0×10^{-5} to train the sidewalk surface recognition with or without transfer learning. When training VGG16+FT+HC, the learning rate was set to 1.0×10^{-4}. The loss function was categorical cross entropy, the minibatch size was 20, and the number of epochs was 100.

4.4 Experimental results

4.4.1 Comparison of estimation accuracy

We compare the estimation accuracies of each method for each subject. Table 5 shows accuracies for each subject. The accuracies in bold indicate the highest accuracy achieved for each subject. "Avg." at the bottom of the table is the average estimated accuracy of all the subjects. The accuracy of VGG16+FT+HC is the highest (when combining the 11 HCFs; see the detailed discussion in the next section). Focusing on "Avg." in Table 5, our proposed method, VGG16+FT+HC, achieved the highest estimation accuracy. Looking at the individual subjects, VGG+FT+HC had the highest accuracy for five of the seven subjects.

We compare the three main points of our proposed method (representation learning with VGG16, FT, and combined HCFs) with the baseline methods. First, we compare conventional methods using only HCFs (SVM-A, B, and RF), the representation learning of the simple CNN, and the representation learning of our proposed method (VGG16). Although the accuracy of the simple four-layer CNN is the same as that of RF with HCFs, VGG16 achieved a higher accuracy of about 5%. The relatively shallow CNN model could not acquire effective feature

Table 5
Accuracy of each method for each subject.

Subject	SVM-A	SVM-B	RF	CNN	CNN+FT	VGG16	VGG16+FT	VGG16+FT+HC
A	**37.5**	27.2	31.1	22.5	24.7	27.8	30.6	36.9
B	22.2	15.6	40.8	40.7	36.2	41.8	35.7	**42.1**
C	45.8	36.7	56.4	50.8	54.4	**57.2**	44.2	50.8
D	47.8	43.9	53.6	54.7	48.3	33.7	**68.3**	66.9
E	40.0	22.2	50.6	53.6	53.1	58.3	66.4	**71.4**
F	26.1	37.5	46.1	45.6	41.1	52.8	56.9	**65.3**
G	24.2	33.1	35.6	38.9	43.3	54.2	56.7	**68.3**
Avg.	34.8	30.9	44.9	43.9	43.0	48.1	51.2	**57.4**

representations for sidewalk surface estimation; however, the relatively deep CNN model (VGG16) led to the acquisition of effective feature representations that improve estimation accuracy. Next, we focused on the introduction of FT, which was not effective in the CNN, because the CNN and CNN+FT are almost equally accurate. In contrast, FT was effective in VGG16, because the accuracy of VGG16+FT is about 3% higher than that of VGG16. Because of the relatively deep model, it can be assumed that FT facilitated the training of all layers. In particular, it is useful knowledge that FT from a large activity recognition dataset helps acquire representations to estimate actual sidewalk surfaces. Finally, we focus on the combination of HCFs. The combination model of VGG16 with 11 HCFs had an even higher estimation accuracy than VGG16+FT. In particular, for Subject A, the accuracy of VGG16+FT was lower than that of RF. However, the accuracy of VGG16+FT+HC improved by about 5% compared with that of RF. Therefore, HCFs are effective for some users. Since five of the seven subjects achieved the highest accuracy with VGG16+FT+HC, the application of transfer learning in combination with HC features is likely to be effective in many cases. However, it should be noted that the number of HCFs was tuned with correct labels of validation data.

In summary, among the conventional methods using HCFs, the method in the preliminary experiments (177 HCFs+RF) achieved an accuracy of 44.9%, which was about 10% higher than those of the methods proposed in related studies (SVM-A and B). The simple CNN had a similar estimation accuracy of 43.9%. However, by adopting the relatively deep VGG16 structure, we achieved an accuracy of 48.1% (RF+3%). Moreover, by adopting FT, we achieved an accuracy of 51.2% (RF+6%). In addition, by combining HCFs, we observed a significant improvement, achieving an accuracy of 57.4% (RF+14%).

Figure 4 shows the training and test accuracies of all the CNN models of Subject E, who achieved the highest accuracy using VGG16+FT+HC. The solid line indicates training accuracy, and the dashed line indicates test accuracy. Training accuracy was generally close to 1 in all the models except VGG16+FT+HC, suggesting that the training converged. On the other hand, in the case of VGG16+FT+HC, a training accuracy of only up to about 80% in 100 epochs was achieved. Therefore, it seems that the training did not converge. Training accuracy may be

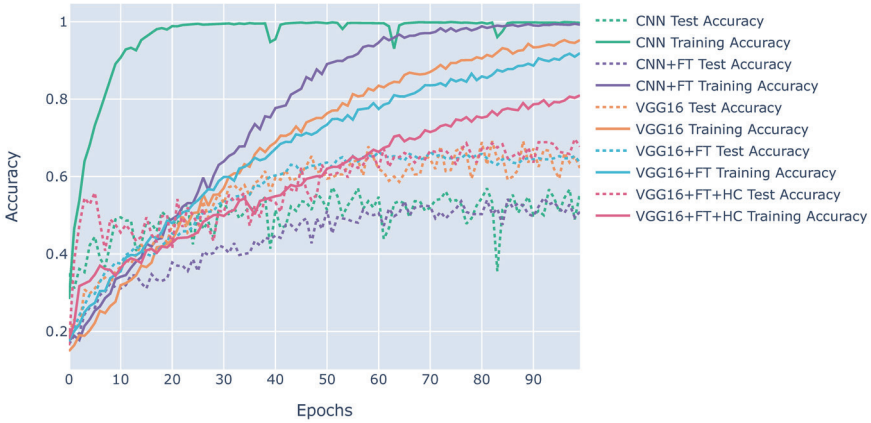

Fig. 4. (Color online) Training and test accuracies of all CNN models of Subject E.

improved by increasing the number of epochs. When we conducted the experiment with the number of epochs set to 300, the training accuracy improved. However, test accuracy did not change and it was confirmed that test loss was increased and overfitting occurred. Therefore, the number of epochs was set to 100.

Figure 5 shows a boxplot of the F-measure for each method with each estimation target. The solid line inside the box indicates the median value, and the dashed line indicates the mean value. In using VGG16+FT+HC, the F-measure is improved for asphalt, lawn, grass, and sand compared with that for RF. The use of the CNN particularly improved the estimation accuracy for asphalt, which was further improved by using VGG16+FT+HC. On the other hand, the F-measure of the mat using RF was the highest. This shows that the most effective method varies depending on the target. When using SVM-A, it is possible, to some extent, to estimate "difficult" sidewalk surfaces, such as the mat and sand; however, it is difficult to estimate other types of surface. This result suggests that the method used for barrier detection is reasonably effective on difficult sidewalk surfaces but not for those that are relatively easy to walk on. Even when the proposed method is used, the effect of individual differences remained significant. Therefore, the F-measure variation is large. Also, the minimum value is about 10% in the gravel. One possible reason for the increased variability is the large effect of the individual differences. In addition, for targets with a low F-measure overall, the users' gait is not likely to change. Therefore, it was difficult to estimate with a high degree of accuracy. Owing to the large effect of individual differences, accuracy may be improved by individual adaptation methods.

4.4.2 Tuning of the number of HCFs combined with VGG16+FT

We examined the changes in estimation accuracy for each subject when the number of HCFs was varied. Figure 6 shows a boxplot of the accuracy for the original VGG16+FT, with different numbers of HCFs ranked in order of importance from 1 to 15. The ranking of importance was determined from the average value of importance calculated for each subject from the random forest. The boxplot where the number of HCFs is 0 indicates VGG16+FT not combined with

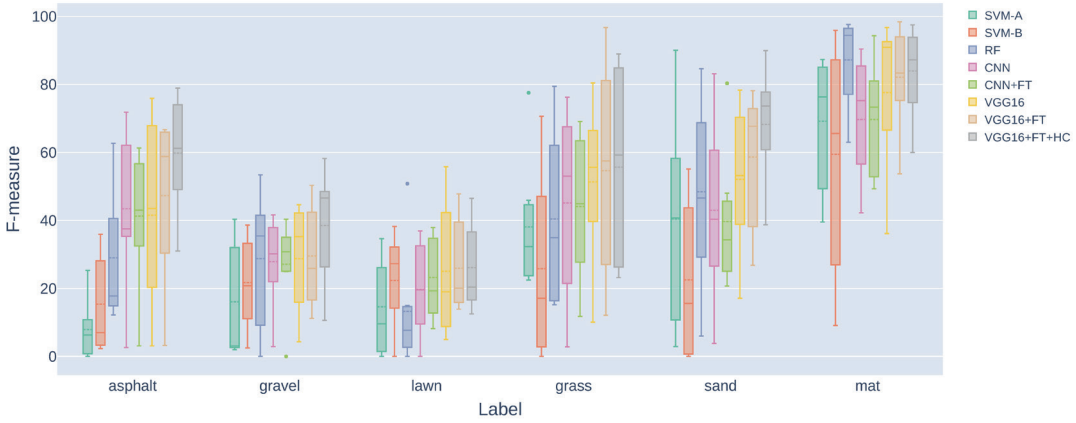

Fig. 5. (Color online) Boxplot of F-measure of each target for each method.

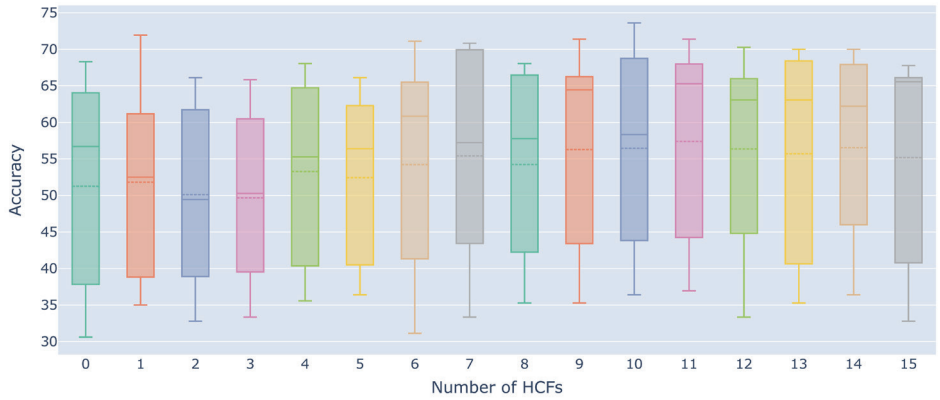

Fig. 6. (Color online) Boxplot of original VGG16+FT with different numbers of HCFs ranked in order of importance from 1 to 15.

Table 6
Top 15 HCFs in order of importance.

Rank	Axis	Domain	Features
1	*y*	Time	Skewness
2	*y*	Frequency	Frequency of 2nd maximum
3	*y*	Frequency (low)	Maximum
4	*y*	Frequency (high)	Standard deviation
5	*y*	Time	Zero-crossing rate
6	*z*	Time	Standard deviation
7	*z*	Time	Root mean square
8	*y*	Frequency (high)	Maximum
9	*z*	Frequency (low)	2nd maximum
10	*z*	Time	Skewness
11	*y*	Frequency (high)	2nd maximum
12	*z*	Frequency (low)	Frequency of maximum
13	*z*	Time	Mean of absolute value
14	*z*	Time	Standard deviation of absolute value
15	*z*	Time	1st quartile

HCFs. Table 6 shows the top 15 HCFs in order of importance. As mentioned above, an average estimation accuracy of 57.4% was achieved when 11 HCFs were combined, and that remained the highest estimation accuracy of all patterns. The median and maximum values were highest when 10 features were combined. From this, it can be seen that the number of effective HCFs differs depending on the subject. The combination of HCFs did not result in a significant decrease in accuracy, and the minimum values generally improved, suggesting that the combination of HCFs is effective. We also verified whether there was a significant difference in accuracy between VGG16+FT and VGG16+FT+HC (11 HCFs) using a t-test. As a result, the p value was 0.00617, which was less than 0.05; therefore, it can be considered that there is a statistically significant difference when comparing VGG16+FT and VGG16+FT+HC (11 HCFs).

One possible reason for the improved accuracy is that features that could not be acquired by VGG16+FT exist in the HCFs. When we calculated the correlation coefficient between

the 3072-dimensional feature representation, which is the output of the Flatten layer of the encoder of VGG16+FT, and the 177-dimensional HCFs, there was almost no correlation. This result indicates that HCFs have feature representations that were not automatically acquired by VGG16+FT.

4.4.3 Possibility of detailed classification of sidewalk surface type

We consider the extent to which the proposed method can estimate the sidewalk surface condition in detail from the sensor data measured by the smartphone in each user's pocket. Tables 7–9 respectively show the confusion matrices for RF, VGG16+FT, and VGG16+FT+HC. Recall is the rate of correctly predicted data in each class, precision is the rate of correctly predicted data out of all the predicted data, and the F-measure is the harmonic average of recall and precision. The F-measure shows that the mat can be estimated with a high degree of accuracy by any method. In addition, when VGG16+FT and VGG16+FT+HC are used, the

Table 7
(Color online) Confusion matrix of RF.

Pre. \ Cor.	Asphalt	Gravel	Lawn	Grass	Sand	Mat	Precision
Asphalt	104	34	42	16	7	1	51.0
Gravel	152	164	145	77	39	0	28.4
Lawn	105	60	70	19	62	4	21.9
Grass	24	95	81	184	55	9	41.1
Sand	32	63	68	88	230	27	45.3
Mat	3	4	14	36	27	379	81.9
Recall	24.8	32.9	16.7	43.8	54.8	90.2	44.9
F-measure	33.3	32.9	18.9	42.4	49.6	85.8	

Table 8
(Color online) Confusion matrix of VGG16-FT.

Pre. \ Cor.	Asphalt	Gravel	Lawn	Grass	Sand	Mat	Precision
Asphalt	228	75	92	9	49	3	50.0
Gravel	82	130	113	45	26	4	32.5
Lawn	75	105	106	43	35	8	28.5
Grass	3	62	68	228	36	27	53.8
Sand	29	45	20	62	242	21	57.8
Mat	3	3	21	33	31	357	79.7
Recall	54.3	31.0	26.8	54.0	57.8	85.0	51.3
F-measure	52.1	31.7	26.8	54.0	57.8	82.3	

Table 9
(Color online) Confusion matrix of VGG16-FT+HC.

Pre. \ Cor.	Asphalt	Gravel	Lawn	Grass	Sand	Mat	Precision
Asphalt	261	56	65	10	27	1	62.1
Gravel	54	183	167	58	21	0	37.9
Lawn	74	77	102	31	28	2	32.5
Grass	0	64	36	223	18	4	64.6
Sand	29	28	26	27	282	18	68.8
Mat	2	12	24	71	43	395	72.2
Recall	62.1	43.6	24.3	53.1	67.3	94.0	57.4
F-measure	62.1	40.5	27.8	58.3	68.0	81.7	

number of cases in which asphalt is misclassified as gravel or lawn is reduced when compared with that of RF. This is confirmed by the fact that the F-measure of asphalt was greatly improved because the activity recognition dataset contained a large amount of data measured on asphalt. The "walk" data on asphalt in the HASC dataset in particular was expected to be similar to that of "asphalt" in this study. Therefore, it is possible that the characteristics of asphalt were acquired by pretraining the activity recognition dataset. However, even when VGG16+FT and VGG16+FT+HC are used, there are many cases where gravel and lawn are misclassified. Therefore, it is difficult to classify sidewalk surface types using only the sensor data measured from the user's smartphone in the pocket.

In summary, it was found to be possible to estimate sidewalk surface types, such as grass, sand, and mat, with a relatively high degree of accuracy from the acceleration data obtained in the user's pocket due to their extreme change in gait. The estimation accuracy of asphalt was improved with the proposed method of VGG16+FT+HC; however, it was difficult to estimate sidewalks (such as gravel and lawn) on which the users had no change in gait. However, the classification of these sidewalks is essential for the purpose of this study. In the future, we will consider using sensors not used in this study or applying more advanced deep learning methods.

4.4.4 Discussion about window size

As a model input, our proposed method used 256 sample acceleration data, which corresponded to about 5 s data because of the sampling frequency of 50 Hz. This means that, when actually implemented as an application, our method needs to measure acceleration data for 5 s. However, in taking 5 s for estimation, it may be difficult to recognize small sidewalk surface changes. Therefore, we examine the effect of window size on accuracy. Figure 7 shows a boxplot of the accuracy of VGG16+FT+HC with various window sizes. In this case, we verified the accuracy with window sizes of 32, 64, 128, and 256. Figure 7 shows that the estimation accuracy significantly decreases as the window size decreases. The mean and median values when the window size is 128 are lower than those when the window size is 256. However, the maximum and minimum values are slightly improved, and, in some cases, a window size of 128 may be effective.

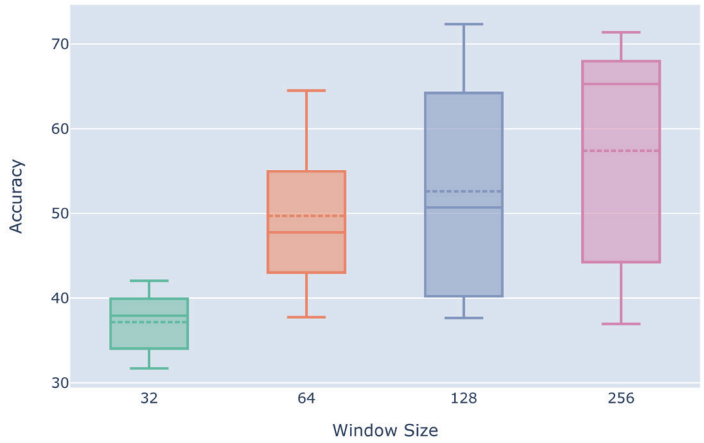

Fig. 7. (Color online) Boxplot of VGG16+FT+HC with window sizes of 32, 64, 128, and 256.

5. Conclusions

In this study, we developed and evaluated a method of estimating sidewalk surface types from acceleration data measured by smartphones while a user is walking. We proposed a new model architecture combining VGG16 with HCFs and a training method with transfer learning using a human activity recognition dataset as a source domain. We conducted experiments to collect sensor data and confirmed that the estimation accuracy increased when the HCFs were combined. By transfer learning, the estimation accuracy increased by 6.4% from that of RF with 177 HCFs. In addition, combining HCFs improved the estimation accuracy by 12.5%. We attempted to determine to what extent sidewalk surfaces can be realized from the data measured by a smartphone. It was found that it is difficult to classify gravel roads and lawns—even using our proposed method. In order to improve the estimation accuracy, we are considering using other sensors not used in this study (e.g., a gyroscope) or applying advanced techniques, such as metric learning, to verify that it is possible to classify sidewalk surfaces in detail. In addition, since the effective model differs depending on the subject, we would like to consider broadening the scope of the investigation to determine what features are effective for estimating a sidewalk surface with high accuracy. In addition to the six types of sidewalk surface targeted in this study, there are other sidewalk surfaces that can be encountered in daily life (e.g., brick and tile roads and slippery concreate roads). It is also therefore our future task to verify whether the proposed method is effective for such surfaces.

Acknowledgments

This research was partially supported by a grant from the Suzuki Foundation.

References

1 J. R. Kwapisz, G. M. Weiss, and S. A. Moore: SIGKDD Explorations **12** (2011) 78. https://doi.org/10.1145/1964897.1964918
2 I. Uyanik, A. Khatri, D. Majesty, M. Ugur, D. Shastri, and I. Pavlidis: Proc. 33rd Annu. ACM Conf. Extended Abstracts on Human Factors In Computing Systems (ACM, 2015) 1397–1402. https://doi.org/10.1145/2702613.2732764
3 A. Miyata, I. Araki, T. Wang: Universal Access In Human-Computer Interaction. Virtual, Augmented, And Intelligent Environments (Springer, Berlin/Heidelberg, 2018) pp. 1397–1402. https://doi.org/10.1007/978-3-319-92052-8_24
4 G. Lee, B. Choi, H. Jebelli, and C. R. Ahn: J. Comput. Civ. Eng. **34** (2020) 04020002. https://ascelibrary.org/doi/abs/10.1061/(ASCE)CP.1943-5487.0000879
5 WheeLog: https://www.wheelog.com/hp/ (accessed September 2020).
6 K. Fujii, M. Hadano, K. Nishida, H. Toda, H. Sawada, and H. Kashima: DEIM 2016 (in Japanese). https://db-event.jpn.org/deim2016/papers/401.pdf
7 B. E. Boser, I. M. Guyou, and V. N. Vapnik: Proc. 5th Annu. Workshop Computational Learning Theory (ACM, 1992) 144–152. https://doi.org/10.1145/130385.130401
8 J. Takahashi, Y. Kobana, Y. Tobe, and G. Lopez: Electron. Commun. Jpn. **101** (2018) 3. https://doi.org/10.1002/ecj.12027
9 M. Hoffmann, M. Mock, and M. May: Proc. 3rd Int. Conf. Ubiquitous Data Mining (CeUR-WS, 2013) 39–43. http://ceur-ws.org/Vol-1088/paper7.pdf
10 S. Jain, C. Borgiattino, Y. Ren, M. Gruteser, Y. Chen, and F. Chiasserini: Proc. 13th Annual Int. Conf. Mobile Systems, Applications, and Services (ACM, 2015) 257–271. https://dx.doi.org/10.1145/2742647.2742669

11 Y. Ohashi, Y. Enokibori, and K. Mase: IPSJ SIG Technical Report 2013-HCI-155 (2013) 1–6 (in Japanese).
12 Y. Ohashi, Y. Enokibori, and K. Mase: IPSJ SIG Technical Report 2014-HCI-160 (2013) 1–8 (in Japanese).
13 D. E. Nunes and V. F. S. Mota: J. Internet Services Appl. **10** (2019) 13. https://doi.org/10.1186/s13174-019-0111-1
14 T. Nomura and Y. Shiraishi: Int. J. Inf. Soc. **7** (2015) 29. http://www.infsoc.org/journal/vol07/IJIS_07_1_029-036.pdf
15 S. Kobayashi, R. Katsurada, and T. Hasegawa: Proc. 7th Int. Conf. Information Technology: IoT and Smart City (ACM, 2019) 497–502. https://doi.org/10.1145/3377170.3377263
16 L. Breiman: Mach. Learning **45** (2001) 5. https://dx.doi.org/10.1023/A:1010933404324
17 NAKAJOU. INC. : http://www.nakajou.com/evermat (accessed September 2020).
18 J. B. Yang, M. N. Nguyen, P. P. San, X. L. Li, and S. Krishnaswamy: Proc. 24th Int. Joint Conf. Artificial Intelligence (ACM, 2015) 3995–4001.
19 H. Gjoreski, J. Bizjak, M. Gjoreski, and M. Gams: Proc. 25th Int. Joint Conf. Artificial Intelligence (ACM, 2016) 1–7.
20 F. Li, K. Shirahama, M. A. Nisar, and L. Köping: Sensors **18** (2019) 1. https://doi.org/10.3390/s18020679
21 T. Hasegawa and M. Koshino: Proc. 2nd Int. Conf. Computational Intelligence and Intelligent Systems (ACM, 2019) 99–104. https://doi.org/10.1145/3372422.3372439
22 N. Kawaguchi, N. Ogawa, Y. Iwasaki, K. Kaji, T. Terada, K. Nurao, S. Inoue, Y. Kawahara, Y. Sumi, and N. Nishino: Proc. 2nd Augmented Human Int. Conf. (ACM, 2011) 1–5. https://doi.org/10.1145/1959826.1959853
23 K. Shimonyan and A. Zisserman: Proc. Int. Conf. Learning Representations (2015). https://arxiv.org/abs/1409.1556
24 J. Deng, E. Dong, R. Socher, L. Li, K. Li, and K. Fei-Fei: Proc. 2009 IEEE Conf. Computer Vision and Pattern Recognition. (IEEE, 2009) 248–255. https://doi.org/10.1109/CVPR.2009.5206848
25 S. Matsui, N. Inoue, Y. Akagi, G. Nagino, and K. Shinoda: Proc. 25th European Signal Processing Conf. (2017) https://doi.org/10.23919/EUSIPCO.2017.8081308
26 T. Hayashi, M. Nishida, N. Kitaoka, T. Noda, and K. Takeda: IEICE Trans. Fundamentals of Electronics, Communications and Computer Sciences E101 (2018) 199–210. https://doi.org/10.1587/transfun.E101.A.199
27 L. Atallah, B. Lo, R. King, and G.-Z. Yang: Proc. 2010 Int. Conf. Body Sensor Networks (IEEE, 2010) 24–29. https://dx.doi.org/10.1109/BSN.2010.23
28 N. Györbíró, Á. Fábián, and G. Hományi: Mobile Netw. Appl. **14** (2009) 82. https://doi.org/10.1007/s11036-008-0112-y
29 K. Altun and B. Barshan: Proc. 1st Int. Conf. Human Behavior Understanding (Springer, 2010) 38–51.
30 J. Yang: Proc. 1st Int. Workshop Interactive Multimedia for Consumer Electronics (ACM, 2009) 1–10. https://dx.doi.org/10.1145/1631040.1631042
31 D. P. Kingma and J. L. Ba: Proc. 3rd Int. Conf. Learning Representations (2015).
32 X. Glorot, A. Bordes, and Y. Bengio: Proc. 14th Int. Conf. Artificial Intelligence and Statistics (2011). http://proceedings.mlr.press/v15/glorot11a.html

About the Authors

Satoshi Kobayashi is a student at Graduate School of Engineering, University of Fukui. His research interests include human activity recognition and deep learning application.

Tatsuhito Hasegawa received his Ph.D. degree in engineering from Kanazawa University, Ishikawa, in 2015. From 2011 to 2013, he was a system engineer with Fujitsu Hokuriku Systems Limited. From 2014 to 2017, he was an assistant with the Tokyo Healthcare University. Since 2017, he has been a senior lecturer with the Graduate School of Engineering, University of Fukui. His research interests include human activity recognition, deep learning application, and intelligent learning support systems. He is a member of IEEE, IPSJ, and JASAG. (t-hase@u-fukui.ac.jp)