IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 16 December 2022, accepted 17 January 2023, date of publication 31 January 2023, date of current version 3 February 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3241043

RESEARCH ARTICLE

User and Period Independent Transportation Mode Detection for Wheelchair Users

SUNGJIN HWANG[1], JIWOONG HEO[1], JUCHEOL MOON[2], JAEHWAN YOU[1], HANSUNG KIM[3], JAEHYUK CHA[1], (Member, IEEE), AND KWANGUK KIM[1]

[1]Department of Computer Science, Hanyang University, Seoul 04763, South Korea
[2]Department of Computer Engineering and Computer Science, California State University Long Beach, Long Beach, CA 90840, USA
[3]Department of Sociology, Hanyang University, Seoul 04763, South Korea

Corresponding author: Kwanguk Kim (kenny@hanyang.ac.kr)

This work was supported by the National Research Foundation of Korea (NRF) Grant through the Korean Government [Ministry of Science and ICT (MSIT)] under Grant 2018R1A5A7059549 and Grant 2021R1A2C2013479.

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the Hanyang University Institutional Review Board under Application No. HYI-18-187-2.

ABSTRACT Transportation mode detection (TMD) is an important research area in human activity recognition. It can improve the mobility and accessibility of people by providing a better understanding of their mobility patterns, thereby enhancing their quality of life and social inclusion. Although previous studies of TMD for people without mobility disabilities exhibited, the performance of TMD models on new users and periods was limited. This issue would be more important for people with mobility disabilities. This study investigated the negative impact of user and period differences on the performance of TMD for wheelchair users (wTMD) and suggested a method to address these challenges. Our main findings are (1) user and period differences degraded the wTMD performance from 94.28% to 59.32%; (2) the multi-DenseNet with a soft voting ensemble provided a 76.49% accuracy to data from different users and periods. We expect that our understanding of wTMD will aid in the design of more generalized wTMD models.

INDEX TERMS Transportation mode detection, deep learning, smartphone, mobility disability.

I. INTRODUCTION

Transportation mode detection (TMD) is one of the categories of human activity recognition (HAR) [1]. TMD technology has significant social benefits and real-life applications, such as urban planning, traffic control, controlling potential hazards, health monitoring, localization and positioning, and journey planning [2], [4]. To exploit such advantages, researchers have extensively studied TMD based on smartphone sensor data, which include motion and location sensors [3], [9], [10]. TMD can be more valuable for people with mobility disabilities because it aids in improving their mobility and accessibility, and such improvement is essential as they are important factors for their quality of life and social inclusion [11], [12].

The associate editor coordinating the review of this manuscript and approving it for publication was Yongming Li.

The performance of TMD models varies depending on their evaluation methods [2], [6], [13]. This issue is not limited to TMD research but also occurs within HAR research [14]. Studies in both TMD and HAR have proved that performance evaluated using the leave-user(s)-out method, where one or more users' data are tested and others are trained, was lower than that of the hold-out method, which randomly splits data from every user into training and test data [6], [14]. Moreover, some TMD researchers are concerned about data collection period differences, which might also impact performance [2], [13]. These problems can be considered as one of the common research challenges in general pattern recognition, the so-called intraclass variability, which indicates variability within the same activity conducted by different users or period states [15].

Developing a TMD model that performs accurately regardless of new users or periods is critical. Several studies, both in

TMD and HAR, emphasized the importance of this generalization issue [2], [6], [13], [14], and it becomes more critical when it is difficult to collect data [1]. As a large gap exists in the dataset size and number of participants between people with and without mobility disabilities, the TMD model for people with mobility disabilities toward the user and period differences needs to be considered.

Despite the importance of TMD models for people with mobility disabilities in the different users and periods, the impact of these differences has not yet been investigated. In this study, we focused on TMD for wheelchair users (wTMD), because wheelchairs are widely used mobility aids for people with mobility disabilities [16]. First, we evaluated the negative impact of user and period differences on wTMD performance by collecting data from different users and periods. Second, we suggested a method, the multi-DenseNet model and soft voting ensemble, to address these intraclass variabilities and evaluate its effectiveness. Our research questions are summarized as follows: **RQ1**: How do user and period differences impact wTMD performance? **RQ2**: To what extent can the multi-DenseNet and soft voting ensemble improve wTMD performance?

II. RELATED STUDIES

A. TMD FOR PEOPLE WITH AND WITHOUT MOBILITY DISABILITIES

In the early stage, researchers attempted to detect transportation modes by applying classical machine learning models [2], [17], [19]. As deep learning techniques have developed, researchers have begun applying deep learning methods to various motion or global positioning system (GPS) sensors. Studies have applied convolutional neural networks (CNNs), ensemble CNNs, convolutional long short-term memory (convolutional LSTM), or other self-designed architectures to the features extracted from GPS sensors, and they achieved up to 92.7% accuracy [9], [10], [20], [23]. In contrast, studies using motion sensors have applied neural networks (NNs), deep neural networks (DNNs), CNNs, recurrent neural networks (RNNs), or other self-designed architectures, resulting in accuracies up to 98.4% [3], [4], [6], [8], [10], [24], [26]. However, these studies only employed random sampling for the training and testing data, and they did not consider the user and period differences between training and test data.

Although TMD for people without mobility disabilities has been widely investigated, there were limited studies on TMD for people with mobility disabilities. Bantis and Haworth suggested a hierarchical dynamic Bayesian network that considers the environment and individual characteristics, such as age and disability [27]. To evaluate their model, they recruited one wheelchair user and one crutch user, and they collected GPS data every 2 min for 3 and 7 days, respectively. Four transportation modes (still, wheelchair or crutches, bus, and rail) were used for the classification. The precision values for the still, wheelchair or crutches, bus, and rail modes were 98%, 62%, 49%, and 79% for the wheelchair user, and 98%, 70%, 64%, and 60% for the crutch user, respectively.

B. INTRACLASS VARIABILITY IN TMD: USER AND PERIOD DIFFERENCES

Intraclass variability—variability within the same activity—is one of the most challenging issues in HAR. This can occur because of differences in users and environmental conditions [15]. Carpineti et al. reported that the neural network model resulted in 93% accuracy on hold-out validation, which exhibited a dramatic performance decrease to 56% on average when the leave-one-user-out cross-validation was applied [6]. Gholamiangonabadi et al. demonstrated that a human activity model with 99.85% accuracy using 10-fold cross-validation decreased to 85.1% when leave-one-user-out cross-validation was applied [14]. Wang et al. suggested that the data collection periods are also an important intraclass variability [13]. Several studies have revealed the impact of intraclass variability on detection performance. However, they did not consider the impact of it on wheelchairs.

C. PROPOSED METHOD: MULTI-DENSENET WITH SOFT VOTING ENSEMBLE

Deep convolutional neural networks are widely used in image recognition to improve their performance [28], [31]; the dense convolutional network (DenseNet) is a deep convolutional network suggested by Huang et al. that enables direct connections between layers to enhance the information flow between layers [31]. The DenseNet performs similarly to residual networks (ResNets) [30] with significantly fewer parameters and computations. It consists of dense blocks that enable information exchange between layers and transition layers which compress the information to maintain the compactness of the model. Because the TMD shares many methodological challenges, such as intraclass variabilities, similar to other image recognition problems [15], we adopted the DenseNet model to develop the multi-DenseNet for wTMD.

To improve the TMD performance, rather than by designing a more sophisticated model, previous studies also used post-processing techniques. Yu et al. suggested error correction via voting to prevent errors that occurred when a single test data contained two different pieces of transportation information owing to the transition of the transportation modes. Their method utilizes previous detection information to enhance the accuracy of the current detection, and it improved the detection accuracy from 91.53% to 94.10% for the best scenario [32]. Wang et al. also predicted a given time length of data by chopping it into small segments and applying majority voting using the prediction results from chopped segments. It increased the detection accuracy from 83.3% to 92.9% [33]. Gjoreski et al. used a stacking ensemble of models to improve performance. They used various TMD models for base models and trained meta-models to combine base models for better prediction. This increased

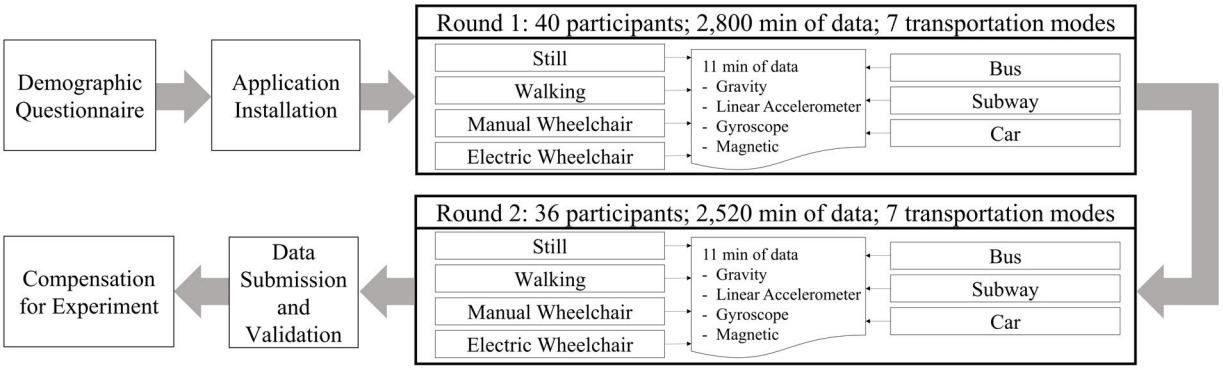

FIGURE 1. Data collection procedure.

the detection accuracy from 89.4% to 93.8%. Additionally, the performance from the ensemble of the best base models was higher than that of all the base models [8]. In this paper, we propose a multi-DenseNet with a soft voting ensemble as a method to improve TMD performance.

III. DATA COLLECTION

To investigate the impact of user and period dependencies on wTMD, we collected smartphone sensor data from different users, and each user was required to collect data corresponding to the given transportation mode from two different periods. Therefore, we collected data from 40 different users, and each participant was asked to collect sensor data in two different periods. We divided the data collection process into rounds 1 and 2, and it was collected in different weeks.

A. PARTICIPANTS AND EQUIPMENT

We recruited 40 participants (mean age = 24.1; standard deviation = 3.2; number of females = 15), and successfully collected data for round 1. Among them, only 36 participants (age mean = 24.1; standard deviation = 3.0; number of females = 12) agreed to participate in round 2 data collection and successfully completed the data collection. The participants used their smartphones to collect sensor data. They used smartphones operating with Android 9 or later. We provided a Start M1 (OttoBock, Duderstadt, Germany) as the manual wheelchair and a B400-KV (OttoBock, Duderstadt, Germany) as the electric wheelchair.

B. DATA COLLECTION PROCEDURE

When participants visited our laboratory, we briefly introduced the data collection process as presented in Figure 1. Participants completed the demographic questionnaires while we installed the smartphone application. After the installation, we demonstrated the use of the data collection application. The application first asks participants to choose a transportation mode that they are going to take. After that, it collects data for 11 min and labels the collected data with the selected transportation mode. Participants collected sensor data for manual and electric wheelchairs from an outdoor area on the campus. For safety, the participants were always accompanied by an experimenter, and the maximum speed of the electric wheelchair was limited to 4 km/h. Participants were asked to collect data from five other transportation modes (still, walking, taking a bus, riding the subway, and driving a car) during their daily life over the next five days. They submitted the collected data after completing the data collection for all seven transportation modes. To address user- and period-independent issues as presented in Figure 2, participants repeated the same data collection procedure after the first round of data collection when they agreed to participate in the second round. After all data collection was complete, we validated the collected data to ensure that there was no error in the collected data. Participants received $20/h as compensation, and the experiment was approved by the research site/university's institutional review board, all participants were provided with a detailed description of the procedure and written consent.

C. DATA PREPROCESSING

Because we collected data using the participants' own smartphones, the sampling rates of sensors varied depending on the smartphone specifications. The raw data were preprocessed to obtain the same sampling rate and eliminate potential errors. To obtain the same sampling rate, we transformed the raw data using linear interpolation to 60 Hz. To remove potential errors, the first and last 30 s of the raw data were removed. Therefore, the total length of each data point was 10 min with a 60 Hz sampling rate.

Each type of sensor output has its own range of values. Because the initial weights of the wTMD model were set with random numbers within the same range, different ranges of data samples might have affected the training process. To avoid these problems, the data samples were standardized using their means and standard deviations from the training dataset. For the value x in the data sample of each sensor, x was transformed to $z = (x - \mu)/\sigma$, where μ and σ are the mean and standard deviation of all the data samples of the sensor in the training dataset.

D. COLLECTED DATA

We collected four motion sensors from the smartphone for seven transportation modes: manual and electric wheelchairs,

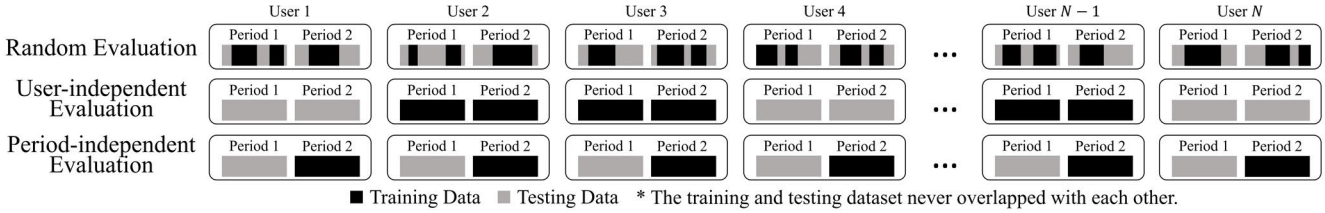

FIGURE 2. Three different evaluations for wTMD: random evaluation (top), user-independent evaluation (middle), and period-independent evaluation (bottom). All evaluation methods were conducted twice by swapping training and testing data. The final values were the average of two evaluations.

TABLE 1. Summary of the collected dataset.

	Round 1	Round 2
Users	40 participants	36 participants
Dataset Size (min)	2,800	2,520
Transportation modes	Manual and electric wheelchairs, still, walking, bus, subway, car	
Sensors	Gravity, linear accelerometer, gyroscope, magnetic	

still, walking, bus, subway, and car. Manual and electric wheelchairs were widely used by wheelchair users. The manual wheelchair requires a certain level of physical abilities or assistance from others, whereas the electric wheelchair is much easier to use. Bus, subway, and car are common transportations which are accessible to wheelchair users and non-wheelchair users. Walking and still were included because they are basic transportation modes in previous TMD studies [6], [13], [18], [20].

Four motion sensors are measured based on the three-dimensional coordinate system [34]. Collected four motion sensors are as follows: (1) gravity sensor which measures the magnitude of gravity for each axis, (2) linear accelerometer sensor which measures acceleration excluding gravity along each axis, (3) gyroscope sensor which measures the rate of rotation around each axis, and (4) magnetic field sensor which measures changes in the earth's magnetic field for each axis [35].

From 40 participants, 2,800 min of data were collected for the first round. Among participants from the first round, only 36 participants agreed to collect data in the second round: 2,520 min of additional data were collected for the second round (Table 1).

IV. METHODOLOGY

A. CLASSIFICATION MODELS

1) MULTIMODALITY

Our proposed architecture includes N models, where each model uses one of the sensors as input. These models operate independently from each other, and their outputs are concatenated before the classification process. Recent studies on TMD based on deep learning methods have widely used multimodal structures [4], [7], [36]. Similarly, we also designed a multi-DenseNet and three baseline models in a multimodal paradigm.

TABLE 2. Architectures of the DenseNet model in the multi-DenseNet for different depth values (L).

Layers	Output Size	$L = 22$	$L = 40$	$L = 76$	$L = 101$
Convolution	150	Convolution layer (kernel size 7, stride 2)-BN[a]-ReLU[b]			
Pooling	75	Max pooling (kernel size 7, stride 2)			
Dense Block (1)	75	$[C_1(1)^c$-$C_3(1)]\times3$	$[C_1(1)$-$C_3(1)]\times6$	$[C_1(1)$-$C_3(1)]\times12$	
Transition Layer (1)	37	$C_1(1)$-AP[d]			
Dense Block (2)	37	$[C_1(1)$-$C_3(1)]\times3$	$[C_1(1)$-$C_3(1)]\times6$	$[C_1(1)$-$C_3(1)]\times12$	
Transition Layer (2)	18	$C_1(1)$-AP			
Dense Block (3)	18	$[C_1(1)$-$C_3(1)]\times3$	$[C_1(1)$-$C_3(1)]\times6$	$[C_1(1)$-$C_3(1)]\times12$	
Transition Layer (3)	9	-			$C_1(1)$-AP
Dense Block (4)	9				$[C_1(1)$-$C_3(1)]\times12$
Pooling	1	Global average pooling			

[a]BN – batch normalization layer. [b]ReLU – ReLU activation. [c]$C_n(a)$ – sequence of BN-ReLU-1D convolutional layer with kernel size n and a strides. [d]AP – average pooling with kernel size 2 and stride 2.

2) MULTI-DENSENET

The current method adopts the DenseNet proposed by Huang et al., which was suggested for image classification problems [31]. The major components of the DenseNet model are the dense and transition layers. In a dense block, the l^{th} layer inputs all preceding layers to provide dense connectivity. The output of the l^{th} layer ($\mathbf{x}_l$) is computed using the following formula: $\mathbf{x}_l = H_l([\mathbf{x}_0, \mathbf{x}_1, \cdots, \mathbf{x}_{l-1}])$, where, $[\mathbf{x}_0, \mathbf{x}_1, \cdots, \mathbf{x}_{l-1}]$ indicates the concatenation of the feature maps of the outputs from layer 0 to $l-1$, and H_l is a composite function. The composite function H_l produces k feature maps, and k is the growth rate. It consists of batch normalization (BN), the rectified linear unit (ReLU) activation function, a one-dimensional (1D) convolutional layer (filters $= 4k$, kernel size $= 1$, and stride $= 1$), BN, ReLU activation function, and a 1D convolutional layer (filters $= k$, kernel sizes $= 3$, and stride $= 1$). As the input of H_l has $k_0 + k \times (l-1)$ feature maps (where k_0 is the number of feature maps in the first input layer of the dense block), the first convolution layer a filter size of $4k$ is used to reduce the input size, thereby improving computational efficiency.

TABLE 3. Architectures of three baseline models.

Model	Architecture
Multi-DNN	[FC(32)[a]-FC(64)-FC(128)][b]-FC(256)-D(0.2)[c]-Softmax
Multi-CNN	[C_3(32)[d]-MP-C_3(64)-MP-C_3(128)-MP]-FC(256)-D(0.2)-Softmax
Multi-LSTM	[LSTM(128)[e]-LSTM(128)]-FC(256)- D(0.2)-Softmax

[a]FC(a) - fully connected layer with a units. [b]Layers in brackets are applied to each sensor independently and layers not in brackets are applied after concatenation, as all baseline models are multimodal. [c]D(a) - drop out layer with a drop out ratio. [d]$C_n(a)$ - 1D convolutional layer with kernel size n, stride 1, and a filters. [e]LSTM(a) - LSTM layer with a units.

Transition layers were placed between the dense blocks. These consist of a BN layer, the ReLU activation layer, a convolutional layer (filters $=$ $\lfloor m/2 \rfloor$, kernel size $=$ 1, and stride $=$ 1), and an average pooling layer (pooling size $=$ 2 and stride $=$ 2), where m is the number of feature maps from the previous dense block. Here, the convolutional layer in the transition block compresses information from the previous dense block by half, and the average pooling layer enables down-sampling, which is an essential component of convolutional networks.

Table 2 presents the architecture of the DenseNet model used in the multi-DenseNet for depths $L = 22$, 40, 76, and 101. The filter sizes used for the convolutional layers in multi-DenseNet are dependent on the growth rate k. After the last dense block, the global average pooling is applied. To construct the multi-DenseNet that satisfies the multimodal paradigm, the outputs of the global average pooling for all sensors are concatenated, and the concatenated vector passes to the softmax layer for the classification process.

3) BASELINE METHOD

We adapted multi-DNN, multi-CNN, and multi-LSTM as baseline methods to compare the performance of the proposed multi-DenseNet. All three models have multimodal architectures. Multi-DNN, multi-CNN, and multi-LSTM consist of three fully connected, three convolutional, and two LSTM layers, respectively, for each sensor. The detailed architectures are shown in Table 3.

4) SOFT VOTING ENSEMBLE

A soft voting ensemble gathers decisions from various classifiers to improve the classification performance. The soft voting ensemble sums the prediction probability from each classifier and predicts the transportation mode with the highest summation value. To design a soft voting ensemble, we combined multi-DenseNet models trained using different sensor combinations. To determine the best models for the soft voting ensemble, we compared the performance of 15 different sensor combinations that are possible from four different sensors (gravity, linear accelerometer, gyroscope, and magnetic). Starting with the model from the highest performing sensor combination, we consecutively added the next best models for the ensemble until the performance of the ensemble model started to decrease. To reduce the dependency of the ensemble model on the dataset, we conducted 4-fold cross-validation.

B. INTRACLASS VARIABILITY CONDITIONS

We trained and evaluated our wTMD models with data samples generated by considering the following three different intraclass variability conditions based on user and period differences: 1) train and test data share both user and period (control condition), 2) train and test data are selected from different users but share period (user-independent condition), and 3) train and test data are selected from different period but share users (period-independent condition).

In the control condition, we selected both training and test data from the consecutive data collected from the same participant because they can be assumed to have the same users and periods. 50% of the dataset was randomly selected as test data to ensure that any test data had training data in its neighbor. In the user-independent condition, we selected training and test data from different users but in the same periods (same rounds). This data. Finally, in the period-independent condition, we selected training and test data from different data collection periods (different rounds) but the same user to enable the training and test data to have only period differences. Figure 2 shows the detailed training and testing data used to evaluate each condition. We kept training size similarly between conditions to ensure the size of the training data would not impact on the performance.

C. PERFORMANCE MEASUREMENTS

1) ACCURACY, PRECISION, AND RECALL

In this study, classification accuracy was used to evaluate the wTMD performance of models, as data were balanced between their classes. Precision and recall values for each transportation mode were also used to illustrate the performance of the models in a specific transportation mode. The three performance measures were computed as follows, where TP, TN, FP, and FN indicates true positive, true negative, false positive, and false negative, respectively:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN},$$
$$Precision = \frac{TP}{TP + FP}, \quad Recall = \frac{TP}{TP + FN}$$

2) IMPLEMENTATION DETAILS

All models used in the study were implemented using Python 3.7 and the Keras 2.4.3, TensorFlow 2.5.0, pandas 1.3.1, and NumPy 1.19.5 libraries. A personal computer with the following configuration was used to train and test the wTMD model: Intel Core i7-10700K processor, 32 GB of RAM, and an NVIDIA GeForce RTX 3080 graphic card.

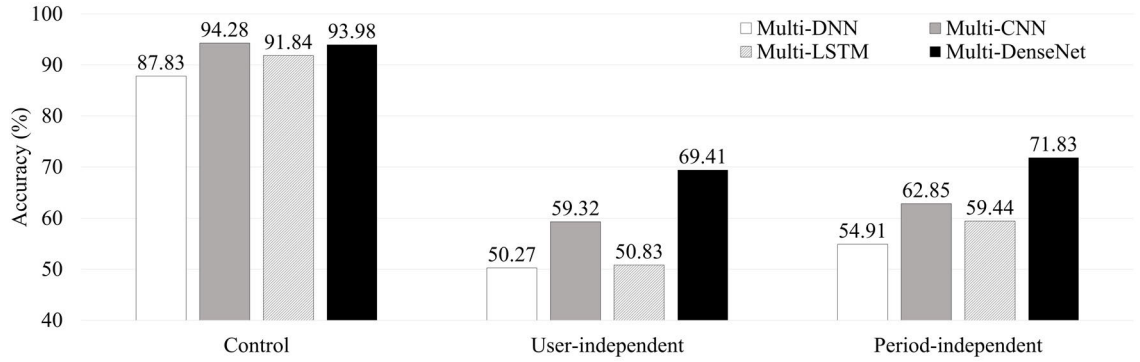

FIGURE 3. Effect of intraclass variability on the wTMD performance.

V. EXPERIMENTAL EVALUATIONS AND RESULTS

A. HOW DOES INTRACLASS VARIABILITY (USER AND PERIOD DIFFERENCES) IMPACT WTMD PERFORMANCE?

Both user and period differences between the training and test data decreased the wTMD performance regardless of the model, transportation mode, and window size. The multi-DenseNet was the most robust to intraclass variability when compared with the baseline models. The detailed results are included below.

1) USER AND PERIOD DIFFERENCES ON WTMD

To determine the impact of user and period differences on wTMD performance, we compared the performance of three intraclass variability conditions (control condition, user-independent condition, period-independent condition) using multi-DenseNet ($L = 101$, $k = 24$) and three baseline models (multi-DNN, multi-CNN, and multi-LSTM). Multi-DenseNet was trained for 50 epochs with 128 batch sizes. The learning rate began from 1×10^{-3} and reduced to 1×10^{-4}, 1×10^{-5}, and 1×10^{-6} at the 21st, 31st, and 41st epochs, respectively, to stabilize the training. Baseline models (multi- DNN, multi-CNN, and multi-LSTM) were trained for 30 epochs with 128 batch sizes. The learning rate began at 1×10^{-3} and decreased to 1×10^{-4}, and 1×10^{-5} at the 11th and 21st epochs, respectively. Every model was trained using the Adam optimizer which computes individual adaptive learning rates using the first and second moments of the gradients with hyper-parameters set as $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\varepsilon = 1 \times 10^{-7}$ [37] and the cross-entropy loss function.

Figure 3 shows that regardless of the model, both user and period differences degraded the wTMD performance. The two factors strongly deteriorated the performance. The impact of the user difference (control condition vs. user-independent condition) was more than 34.96 percentage points (pp) for the baseline models and 24.57pp for the multi-DenseNet. The impact of the period difference (control condition vs. period-independent condition) was more than 31.43pp for the baseline models and 22.15pp for the multi-DenseNet. In contrast, at most 8.61pp for baseline models and 2.42pp for multi-DenseNet gap existed between the user and period differences (user-independent condition vs. period-independent condition). Table 4 shows the detailed impacts of user and period differences on wTMD performance.

TABLE 4. Effect of user and period differences on the wTMD performance.

	Multi-DNN	Multi-CNN	Multi-LSTM	Multi-DenseNet
User Difference[a] (pp)	37.56	34.96	41.01	24.57
Period Difference[b] (pp)	32.92	31.43	32.40	22.15

[a]User difference was computed by subtracting the performance of the user-independent condition from that of the control condition.
[b]Period difference was computed by subtracting the performance of the period-independent condition from that of the control condition.

2) EFFECTS ON DEEP LEARNING MODELS

Figure 3 also shows that the difference in wTMD performance between models became more evident under user- and period-independent conditions. For the control condition, multi-CNN slightly outperformed multi-DenseNet. However, multi-DenseNet performed higher on both user- and period-independent conditions showing that it is the most robust model against intraclass variability when compared with the baseline models (Table 4). The wTMD performance of multi-DenseNet only decreased by 24.57pp (user difference) and 22.15pp (period difference), whereas it decreased by at least 34.96pp (user difference) and 31.43pp (period difference) for the baseline models.

3) EFFECTS ON TRANSPORTATION MODES

To find out the impact of intraclass variability on transportation modes, we compared three intraclass variability conditions (control, user-, and period-independent conditions) on seven recall values. Moreover, to determine the performance variation between data, we computed the standard deviation of the performance from each consecutive data in the dataset. The performance is evaluated using multi-DenseNet ($L = 101$, $k = 24$).

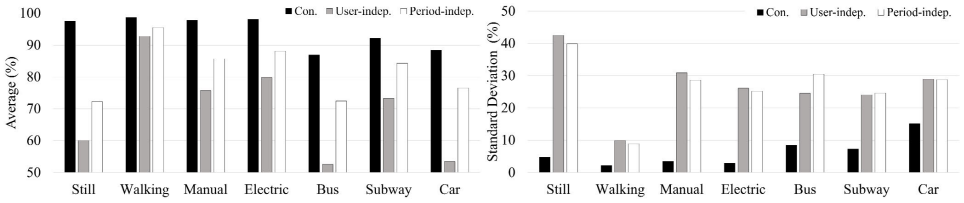

FIGURE 4. Comparison between three conditions of the average (left) and standard deviation (right) of recall values from consecutive 10 min data on seven transportation modes. Con. – control condition. User-indep. – user-independent condition. Period-indep. – period-independent condition.

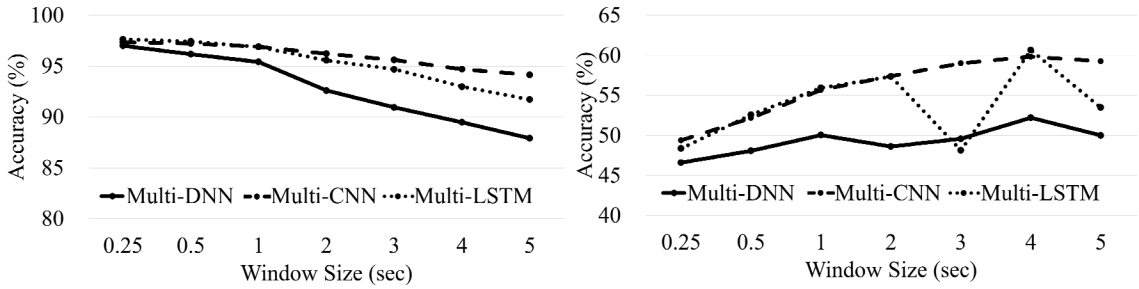

FIGURE 5. wTMD accuracies of multi-CNN and multi-LSTM on control condition (left) and user-independent condition (right) depending on window size.

TABLE 5. Precisions and recall values (%) of the multi-DenseNet and baseline models on the user-independent condition. Manual and electric indicate manual and electric wheelchairs.

Model	Multi-DNN		Multi-CNN		Multi-LSTM		Multi-DenseNet	
	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall
Still	74.41	39.78	73.06	50.55	76.97	43.40	83.13	60.07
Walking	83.00	87.17	90.58	91.27	95.20	94.04	93.38	92.76
Manual[a]	52.08	55.07	66.61	79.43	56.10	58.07	81.94	75.72
Electric[b]	46.10	43.16	67.81	56.18	57.52	57.02	80.08	79.80
Bus	28.33	31.60	40.23	38.62	35.79	34.45	45.50	52.59
Subway	43.06	64.74	43.50	63.03	36.36	64.21	63.41	73.27
Car	35.45	28.46	41.11	35.79	33.46	23.20	50.81	53.46

[a]Manual – manual wheelchair. [b]Electric – electric wheelchair.

Figure 4 (left) shows that the intraclass variability diminished the detection performance in every transportation mode. The impact of intraclass variability (control condition vs. user-independent condition) highly debased the performance of the still (37.47pp), manual (22.09pp) and electric (18.29pp) wheelchair, bus (34.26pp), subway (18.89pp), and car (34.94pp) modes compared with walking (5.92%) mode. However, note that when two similar classes, such as manual and electric wheelchairs, or bus and car, were merged into wheelchairs or bus/car, respectively, a large performance decrement due to intraclass variability significantly shrunk to 7.31pp for wheelchairs and 16.54pp for bus/car.

Figure 4 (right) shows that the intraclass variability also largely deteriorated the standard deviation for the still (37.47pp), manual wheelchair (27.38pp), and electric wheelchair (23.19pp) modes. However, the walking, bus, subway, and car scenarios were not highly impacted by intraclass variability compared to still, manual, and electric wheelchairs. Moreover, the standard deviation values for walking were mostly less than 10% and mostly more than 10% for the bus, subway, and car modes, regardless of the condition. However, when similar classes were merged, as mentioned above, the standard deviations of wheelchairs and bus/car decreased to 11.24pp and 12.19pp. Furthermore, the still and subway modes had standard deviations of more than 20% in user- and period-independent conditions, whereas the standard deviations for the other modes were mostly lower than 20% under these conditions.

4) EFFECTS ON WINDOW SIZES

We compared the user-independent condition and control condition for seven different window sizes (0.25, 0.5, 1, 2, 3, 4, and 5 s). The comparison was conducted using three baseline models: multi-DNN, multi-CNN, and multi-LSTM. Figure 5 shows that the wTMD performance decreased because of the intraclass variability, regardless of window size. However, for the user-independent condition, the performance tended to increase as the window size increased; in contrast, it decreased for the control condition as the window size increased.

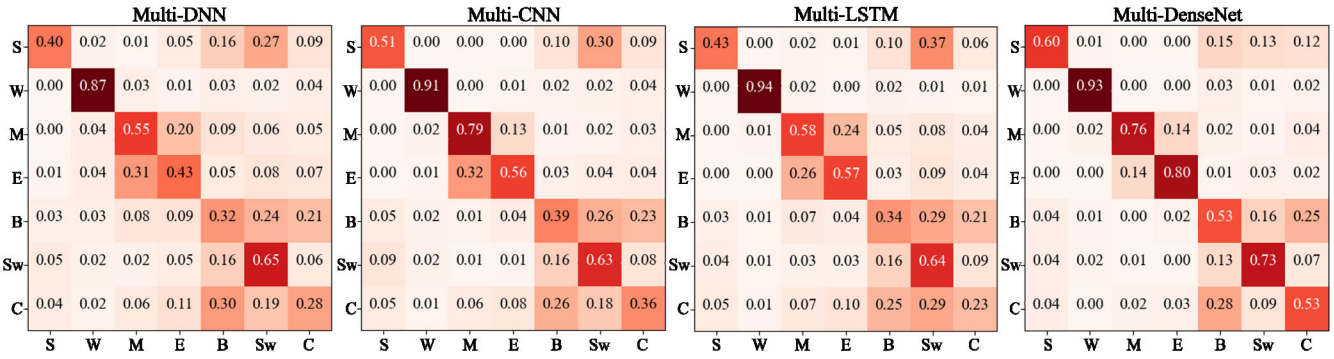

FIGURE 6. Confusion matrices of the multi-DNN, multi-CNN, multi-LSTM, and multi-DenseNet (L=101, k=24) on the user-independent condition. The vertical axis of each confusion matrix is the ground truth and the horizontal axis is the prediction. S – still, W – walking, M – manual wheelchair, E – electric wheelchair, B – bus, Sw – subway, C – car.

B. TO WHAT EXTENT CAN MULTI-DENSENT AND THE SOFT VOTING ENSEMBLE IMPROVE WTMD PERFORMANCE?

The multi-DenseNet performed 10.09pp higher than the highest baseline model (multi-CNN). A higher depth and growth rate in the multi-DenseNet improved the wTMD performance. Moreover, the soft voting ensemble improved the performance by 7.08pp. The detailed results are included below.

1) IMPROVEMENT OF THE WTMD PERFORMANCE USING THE MULTI-DENSENET

The multi-DenseNet provided a 69.41% accuracy, which was 10.09pp higher than the highest baseline model (multi-CNN). The precision and recall values for each transportation mode shown in Table 5 indicate that the multi-DenseNet had higher precision (15.33pp) on a manual wheelchair with a slightly lower recall rate (3.71pp) than the best values from the baseline models. For the electric wheelchair, it enhanced both precision (12.27pp) and recall (22.78pp) rates. In addition to wheelchairs, multi-DenseNet was more robust in detecting the still mode. This significantly improved the recall rate (9.52pp). The confusion matrices in Figure 6 show that this was achieved by reducing the number of misclassifications of the still to subway mode. A noticeable increase was also observed in the recall rates for the bus (13.97pp) and car (17.67pp) modes by reducing the number of misclassifications of the bus or car to subway mode. A significant increase in the precision value (19.91pp) on the subway mode was also observed. We compared multi-DenseNet with the highest precision or recall value among the three baselines to compute the values mentioned above.

The confusion matrix of the multi-DenseNet in Figure 6 shows that there was confusion within wheelchairs (14% of manual to electric and 14% of electric to manual) and within road vehicles (25% of buses to cars and 28% of cars to buses). 4.5% of wheelchairs were misclassified as road vehicles (buses and cars) on average, and 20% of the subway to road vehicles. Confusion also occurred between still and vehicle (bus, subway, and car) modes. 4% of vehicles were misclassified as still on average, and 27% of the still mode were classified as road vehicles and 13% as the subway. Walking had no meaningful confusion with any other transportation modes.

TABLE 6. wTMD performances of multi-DenseNet on the user-independent condition with eight different configurations.

Depth	Growth Rate	Size of Parameters	Training Time (hour)	Accuracy (%)
22	12	0.2M[1]	0.4	66.81
22	24	0.7M	0.5	65.47
40	12	0.6M	0.7	65.96
40	24	1.9M	0.7	68.47
76	12	1.7M	1.4	68.52
76	24	5.9M	2.0	68.05
101	12	2.5M	2.1	69.35
101	24	8.7M	2.6	69.41

[a]M – millions.

2) DEPTH AND GROWTH RATE OF THE MULTI-DENSENET

To determine the impact of the depth and growth rate of the multi-DenseNet on the user-independent condition, four different depths (22, 40, 76, and 101) and two different growth rates (12 and 24) were evaluated. All multi-DenseNets were trained for 50 epochs with 128 batch sizes. The learning rate began from 1×10^{-3} and reduced to 1×10^{-4}, 1×10^{-5}, and 1×10^{-6} at the 21st, 31st, and 41st epochs, respectively, to stabilize the training. The Adam optimizer and cross-entropy loss function were used for training.

Table 6 shows that the increase in both the depth and growth rate in multi-DenseNet resulted in a higher wTMD performance. Additionally, stacking additional convolutional layers was more efficient compared with applying a higher growth rate, as a model using additional dense block better with fewer parameters compared with a model having a higher growth rate. For example, multi- DenseNet ($L = 40$, $k = 12$) exhibited better performance with lower values of the parameters compared with multi-DenseNet ($L = 22, k = 24$). Owing to deeper layers and a higher growth rate, the accuracy increased by 2.60pp at most.

TABLE 7. Accuracies for each sensor combination and the soft voting ensemble in the user-independent condition using the multi-DenseNet.

Rank[a]	Sensor Combination				4-fold CV[b] (%)	4-fold CV Soft Voting Ensemble[c] (%)
	GR	LA	G	M		
1		V	V	V	76.86	76.86
2		V		V	76.18	77.92
3		V	V		75.32	78.97
4		V			75.23	79.71
5	V	V	V	V	74.21	79.70
6	V	V		V	74.21	79.52
7	V	V	V		73.71	79.75
8	V	V			72.10	79.71
9			V	V	68.41	80.05
10	V		V	V	66.78	79.76
11			V		65.91	80.34
12	V		V		61.88	80.15
13	V			V	51.41	79.79
14				V	48.54	79.88
15	V				43.90	79.54

[a]Sensor combination is ranked in descending order of the average accuracy of 4-fold cross-validation. [b]4-fold CV indicates the average accuracy of the results from leave-users-out 4-fold cross-validation. [c]4-fold CV Soft Voting Ensemble indicates the average accuracy of the results from the soft voting ensemble.

3) OPTIMAL WINDOW SIZE FOR THE MULTI-DENSENET

Every multi-DenseNet was initially tested with a window size of 5 s; however, because it may not be the optimal window size for the multi-DenseNet ($L = 101, k = 24$), we estimated additional window sizes (3, 4, and 6 s): 67.29% for 3 s, 67.32% for 4 s, and 66.58% for 6 s. As a result, a size of 5 s (69.41%) was also optimal for the multi-DenseNet.

4) SOFT VOTING ENSEMBLE

We tested our soft voting ensemble of multi-DenseNets under the user-independent condition. To find out the best sensor combinations for the soft voting ensemble, we conducted leave-users-out 4-fold cross-validation on the dataset, which corresponds to the user-independent condition. Table 7 shows that the soft voting ensemble had the best accuracy (80.34%) when the 11th sensor combination models were used in the 4-fold cross-validation. This indicated that not every sensor combination aids in increasing the performance as the performance of the ensemble began to decrease after the 11^{th} model. The final result of the soft voting ensemble of these 11^{th} sensor combination models evaluated on the user-independent condition was 76.49%, which was 7.08pp higher than that of the multi-DenseNet (69.41% accuracy) from RQ1. Recall values in Figure 7 show that the soft voting ensemble improved the performance for every transportation mode when compared with that from Figure 6.

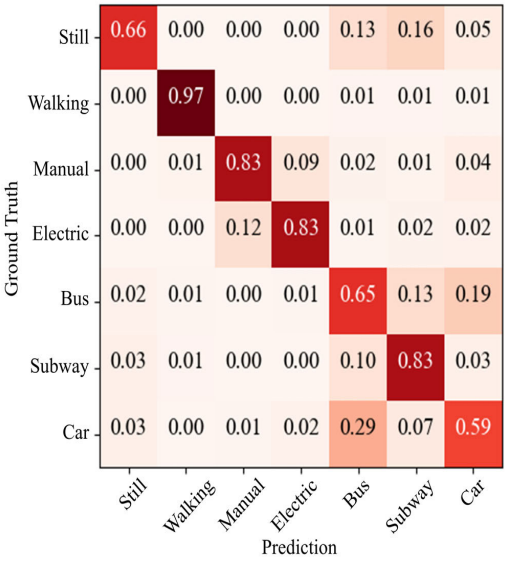

FIGURE 7. Confusion matrices of the soft voting ensemble of 11 models trained from different sensor combinations in the user-independent condition.

VI. DISCUSSION

The aim of this study was to demonstrate the impact of user and period differences on wTMD performance and overcome the decrease in the wTMD performance using a method, the multi-DenseNet, and the soft voting ensemble. By comparing the wTMD performance on three different intraclass variabilities, we demonstrated the importance of intraclass variability. Moreover, the multi-DenseNet outperformed other baselines when evaluated in high intraclass variability conditions, and the soft voting ensemble of models from various sensor combinations improved the performance.

Our results on wTMD were in line with the results from previous TMD studies. Carpineti et al. evaluated the leave-one-user-out method, which corresponds with the user-independent condition in our study, and demonstrated that the TMD performance decreased from 93% to 56% when compared with the hold-out validation [6]. Wang et al., the average TMD performance for different users (63.3%), which corresponds with the user-independent condition, was similar to the performance at different positions (63.0%) [13]. Moreover, the gap between the performance at different positions (63.0%) and different periods (71.1%) by Wang et al. suggests that environmental differences can also be subdivided into several factors, and the impact of each sub-factor might differ. Thus, our results, as well as previous studies, suggest that intraclass variability is significant for both wTMD and TMD research.

Regardless of the models, modes, and window sizes, the wTMD performance decreased as the intraclass variability increased. Interestingly, the impact of intraclass variability on these factors differed according to the machine learning models. The impact of intraclass variability was larger in the three

baseline models, as compared with that in multi-DenseNet. As the baseline models might more easily learn data-oriented characteristics, instead of general characteristics, than the proposed method, we may need a method for better wTMD intraclass variability. With regard to the modes, the impact of intraclass variability on manual and electric wheelchair modes was relatively critical in the current study. Furthermore, short window sizes afforded better performance under the control condition, whereas sufficiently long window sizes enabled better performance for wTMD intraclass variability. Short window sizes made it easier to share data-specific characteristics with the neighboring training data. Therefore, a long window size may be required. These results suggest that the effects of the transportation modes and longer window sizes, as well as those of the proposed model, should be considered for future wTMD.

The multi-DenseNet was the most robust model for intraclass variability compared with other baseline models (multi-DNN, multi-CNN, and multi-LSTM) in the wTMD. The improvement was achieved by reducing confusion among vehicles (bus, subway, and car), wheelchairs (manual and electric), and between the still mode and vehicles. Although it significantly improved the wTMD performance, some confusion between wheelchairs or vehicles remained. Such confusion can be interpreted in two ways. First, confusion among vehicles or wheelchairs can be reasoned because of the similarity between activities. As activities such as using two wheelchairs or two road vehicles (bus and car) are highly similar, it can result in similarity in sensor information. Second, confusion between still and vehicles occurs because of differences in local and global behaviors. Vehicles periodically stop because of traffic signals or arrival at the next station. Only local sensor information from a short window size may be insufficient, resulting in confusion.

The performance of multi-DenseNet under high intraclass variability conditions might suggest the possibility of deep convolutional neural networks to improve the performance of wTMD. In image classification studies, deep convolutional neural networks including DenseNet have been extensively investigated to improve the detection performance [28], [31]. However, it has not been extended to wTMD problems. Considering that wTMD and image classification share some characteristics in the sense of pattern recognition, exploiting some advantages in image classification models or other pattern recognition areas might improve wTMD performance. Soft voting enables the use of models trained using different sensor combinations to reduce the errors made by each model. Our experiment demonstrated that 11th models are required for the best performance of the ensemble model. One reason for this is that an ensemble may require a sufficient number of models to ignore the erroneous judgment from a single model. When using more than 10 models, each model can only provide a maximum of 10% error to the ensemble model, which can be rectified by decisions from other models. The performance of the ensemble decreased when the 12^{th} model was added. This was because the performance of each model after the 12^{th} model decreased, which may hinder making better decisions when used for the ensemble. Similarly, a previous TMD study suggested an ensemble model and observed that using only the best models for the ensemble performed slightly higher than using every model [8].

Notably, this study had some limitations. First, we only investigated the TMD in wheelchairs, among other mobility disabilities. A crutch or mobility scooter may also be used by people with mobility disabilities; therefore, transportation modes other than wheelchairs should be evaluated in future studies. Second, although, we showed the user and period dependency of the wTMD model, it still remains limited. Other environmental differences such as smartphone position or travel routes (e.g. indoor subway or outdoor subway) that may impact the wTMD can also exist.

VII. CONCLUSION

Understanding the response of wTMD to a new user and period is important for its application in real-world systems. In this study, we investigated the impact of intraclass variability, a variation between training and test data due to the user or period differences, on wTMD by designing three intraclass variability conditions. Our results indicated that intraclass variability degrades wTMD performance, and user and period differences exist. The multi-DenseNet, a deep convolutional network, and a soft voting ensemble that we suggested improved the wTMD performance. Our experimental findings suggest that a model for intraclass variability is important for designing a real-world applicable wTMD framework. Moreover, using deep convolutional networks and a soft voting ensemble can be a better solution to achieve robustness. We expect that understanding intraclass variability on wTMD will foster the development of generalized wTMD models that will eventually enhance the quality of life and social inclusion of people with mobility disabilities.

ACKNOWLEDGMENT

This article is based on the master's thesis of Sungjin Hwang from Hanyang University. *(Sungjin Hwang and Jiwoong Heo contributed equally to this work.)*

REFERENCES

[1] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 3, pp. 1192–1209, 3rd Quart., 2013, doi: 10.1109/SURV.2012.110112.00192.

[2] S. Hemminki, P. Nurmi, and S. Tarkoma, "Accelerometer-based transportation mode detection on smartphones," in *Proc. 11th ACM Conf. Embedded Networked Sensor Syst.*, Nov. 2013, pp. 1–14.

[3] S.-H. Fang, Y.-X. Fei, Z. Xu, and Y. Tsao, "Learning transportation modes from smartphone sensors based on deep neural network," *IEEE Sensors J.*, vol. 17, no. 18, pp. 6111–6118, Sep. 2017, doi: 10.1109/JSEN.2017.2737825.

[4] Y. Qin, H. Luo, F. Zhao, C. Wang, J. Wang, and Y. Zhang, "Toward transportation mode recognition using deep convolutional and long short-term memory recurrent neural networks," *IEEE Access*, vol. 7, pp. 142353–142367, 2019, doi: 10.1109/ACCESS.2019.2944686.

[5] X. Liang and G. Wang, "A convolutional neural network for transportation mode detection based on smartphone platform," in *Proc. IEEE 14th Int. Conf. Mobile Ad Hoc Sensor Syst. (MASS)*, Oct. 2017, pp. 338–342.

[6] C. Carpineti, V. Lomonaco, L. Bedogni, M. D. Felice, and L. Bononi, "Custom dual transportation mode detection by smartphone devices exploiting sensor diversity," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. Workshops (PerCom Workshops)*, Mar. 2018, pp. 367–372.

[7] C. Wang, H. Luo, F. Zhao, and Y. Qin, "Combining residual and LSTM recurrent networks for transportation mode detection using multimodal sensors integrated in smartphones," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 9, pp. 5473–5485, Sep. 2021, doi: 10.1109/TITS.2020.2987598.

[8] M. Gjoreski, V. Janko, G. Slapničar, M. Mlakar, N. Reščič, J. Bizjak, V. Drobnič, M. Marinko, N. Mlakar, M. Luštrek, and M. Gams, "Classical and deep learning methods for recognizing human activities and modes of transportation with smartphone sensors," *Inf. Fusion*, vol. 62, pp. 47–62, Oct. 2020, doi: 10.1016/j.inffus.2020.04.004.

[9] G. Jiang, S.-K. Lam, P. He, C. Ou, and D. Ai, "A multi-scale attributes attention model for transport mode identification," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 1, pp. 152–164, Jan. 2022, doi: 10.1109/TITS.2020.3008469.

[10] J. J. Q. Yu, "Travel mode identification with GPS trajectories using wavelet transform and deep learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 2, pp. 1093–1103, Feb. 2021, doi: 10.1109/TITS.2019.2962741.

[11] H. Matthews, L. Beale, P. Picton, and D. Briggs, "Modelling access with GIS in urban systems (MAGUS): Capturing the experiences of wheelchair users," *Area*, vol. 35, no. 1, pp. 34–45, Mar. 2003, doi: 10.1111/1475-4762.00108.

[12] R. Velho, "Transport accessibility for wheelchair users: A qualitative analysis of inclusion and health," *Int. J. Transp. Sci. Technol.*, vol. 8, no. 2, pp. 103–115, Jun. 2019, doi: 10.1016/j.ijtst.2018.04.005.

[13] L. Wang, H. Gjoreski, M. Ciliberto, S. Mekki, S. Valentin, and D. Roggen, "Enabling reproducible research in sensor-based transportation mode recognition with the Sussex-Huawei dataset," *IEEE Access*, vol. 7, pp. 10870–10891, 2019, doi: 10.1109/ACCESS.2019.2890793.

[14] D. Gholamiangonabadi, N. Kiselov, and K. Grolinger, "Deep neural networks for human activity recognition with wearable sensors: Leave-one-subject-out cross-validation for model selection," *IEEE Access*, vol. 8, pp. 133982–133994, 2020, doi: 10.1109/ACCESS.2020.3010715.

[15] A. Bulling, U. Blanke, and B. Schiele, "A tutorial on human activity recognition using body-worn inertial sensors," *ACM Comput. Surv.*, vol. 46, no. 3, pp. 1–33, 2014, doi: 10.1145/2499621.

[16] C. Torkia, D. Reid, N. Korner-Bitensky, D. Kairy, P. W. Rushton, L. Demers, and P. S. Archambault, "Power wheelchair driving challenges in the community: A users' perspective," *Disab. Rehabil., Assistive Technol.*, vol. 10, no. 3, pp. 211–215, 2015, doi: 10.3109/17483107.2014.898159.

[17] S. Wang, C. Chen, and J. Ma, "Accelerometer based transportation mode recognition on mobile phones," in *Proc. Asia–Pacific Conf. Wearable Comput. Syst.*, 2010, pp. 44–46.

[18] S. Reddy, M. Mun, J. Burke, D. Estrin, M. Hansen, and M. Srivastava, "Using mobile phones to determine transportation modes," *ACM Trans. Sensor Netw.*, vol. 6, no. 2, pp. 1–27, 2010, doi: 10.1145/1689239.1689243.

[19] P. Widhalm, P. Nitsche, and N. Brändie, "Transport mode detection with realistic smartphone sensor data," in *Proc. 21st Int. Conf. Pattern Recognit. (ICPR)*, Nov. 2012, pp. 573–576.

[20] S. Dabiri and K. Heaslip, "Inferring transportation modes from GPS trajectories using a convolutional neural network," *Transp. Res. C, Emerg. Technol.*, vol. 86, pp. 360–371, Jan. 2018, doi: 10.1016/j.trc.2017.11.021.

[21] L. Li, J. Zhu, H. Zhang, H. Tan, B. Du, and B. Ran, "Coupled application of generative adversarial networks and conventional neural networks for travel mode detection using GPS data," *Transp. Res. A, Policy Pract.*, vol. 136, pp. 282–292, Jun. 2020, doi: 10.1016/j.tra.2020.04.005.

[22] A. Nawaz, H. Zhiqiu, W. Senzhang, Y. Hussain, I. Khan, and Z. Khan, "Convolutional LSTM based transportation mode learning from raw GPS trajectories," *IET Intell. Transp. Syst.*, vol. 14, no. 6, pp. 570–577, Jun. 2020, doi: 10.1049/iet-its.2019.0017.

[23] A. Yazdizadeh, Z. Patterson, and B. Farooq, "Ensemble convolutional neural networks for mode inference in smartphone travel survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 6, pp. 2232–2239, Jun. 2020, doi: 10.1109/TITS.2019.2918923.

[24] T. H. Vu, L. Dung, and J.-C. Wang, "Transportation mode detection on mobile devices using recurrent nets," in *Proc. 24th ACM Int. Conf. Multimedia*, Oct. 2016, pp. 392–396.

[25] A. Sharma, S. K. Singh, S. S. Udmale, A. K. Singh, and R. Singh, "Early transportation mode detection using smartphone sensing data," *IEEE Sensors J.*, vol. 21, no. 14, pp. 15651–15659, Jul. 2021, doi: 10.1109/JSEN.2020.3009312.

[26] G. Asci and M. A. Guvensan, "A novel input set for LSTM-based transport mode detection," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. Workshops (PerCom Workshops)*, Mar. 2019, pp. 107–112.

[27] T. Bantis and J. Haworth, "Who you are is how you travel: A framework for transportation mode detection using individual and environmental characteristics," *Transp. Res. C, Emerg. Technol.*, vol. 80, pp. 286–309, Jul. 2017, doi: 10.1016/j.trc.2017.05.003.

[28] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1097–1105, doi: 10.1145/3065386.

[29] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[31] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4700–4708.

[32] M.-C. Yu, T. Yu, S.-C. Wang, C.-J. Lin, and E. Y. Chang, "Big data small footprint: The design of a low-power classifier for detecting transportation modes," *Proc. VLDB Endowment*, vol. 7, no. 13, pp. 1429–1440, 2014, doi: 10.14778/2733004.2733015.

[33] L. Wang and D. Roggen, "Sound-based transportation mode recognition with smartphones," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 930–934.

[34] *Sensors Overview*. Accessed: Oct. 7, 2022. [Online]. Available: https://developer.android.com/guide/topics/sensors/sensors_overview

[35] *Motion Sensors*. Accessed: Oct. 7, 2022. [Online]. Available: https://developer.android.com/guide/topics/sensors/sensors_motion

[36] B. Friedrich, C. Lübbe, and A. Hein, "Analyzing the importance of sensors for mode of transportation classification," *Sensors*, vol. 21, no. 1, p. 176, Dec. 2020, doi: 10.3390/s21010176.

[37] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

SUNGJIN HWANG received the B.S. degree in mathematics and the M.S. degree in computer science from Hanyang University, Seoul, South Korea, in 2020 and 2022, respectively. He is a Researcher with the Human–Computer Interaction Laboratory, Hanyang University. His research interests include human–computer interaction, wearable computing, digital healthcare, and accessibility.

JIWOONG HEO received the B.E. degree from the Global School of Media, Soongsil University, Seoul, South Korea, in 2016. He is currently pursuing the Ph.D. degree in computer science with the Laboratory for Human-Computer Interaction, Hanyang University, Seoul. His research interests include transportation mode detection, machine learning, and deep learning.

JUCHEOL MOON received the B.S. degree in physics from Korea University, Seoul, South Korea, in 2004, the M.S. degree in computer science from South Dakota State University, Brookings, USA, in 2012, and the Ph.D. degree in computer science from Iowa State University, Ames, USA, in 2017. He is an Assistant Professor with the Department of Computer Engineering and Computer Science, California State University, Long Beach. His research interests include interpretable deep learning, representation learning, and data analysis.

JAEHWAN YOU received the B.S. degree in digital media from Ajou University, South Korea, in 2017. He is currently pursuing the Ph.D. degree in computer science with Hanyang University. His research interests include human factor and mixed reality.

HANSUNG KIM received the M.S.W. degree from Michigan State University and the Ph.D. degree in social work from the University of Southern California. He was an Assistant Professor at the Department of Social Work, California State University, Fullerton. He is currently with Hanyang University, Seoul, South Korea, where he teaches courses in social policy, sociology of welfare, and social research methods. His research interests include social policies and social inequality.

JAEHYUK CHA (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in computer science from Seoul National University, South Korea, in 1987, 1991, and 1997, respectively. He was with the Korea Research Information Center, from 1997 to 1998. Since 1998, he has been a Professor with the Department of Computer and Software, Hanyang University, Seoul, South Korea. His research interests include DBMS, flash storage systems, and multimedia content adaptation.

KWANGUK (KENNY) KIM received the Ph.D. degree in biomedical engineering from Hanyang University, Seoul, South Korea, in 2009. He was a Postdoctoral Researcher at Duke University, Durham, NC, USA, in 2009 and 2010, and the University of California, Davis, CA, USA, in 2010 and 2013. Since 2013, he has been with the Department of Computer Science, Hanyang University, where he is currently an Associate Professor. His research interests include human–computer interaction, human–AI interaction, and human–vehicle interaction.