frontiers
in Computer Science

ORIGINAL RESEARCH
published: 21 September 2021
doi: 10.3389/fcomp.2021.713719

Three-Year Review of the 2018–2020 SHL Challenge on Transportation and Locomotion Mode Recognition From Mobile Sensors

Lin Wang[1†], Hristijan Gjoreski[2†], Mathias Ciliberto[3†], Paula Lago[4], Kazuya Murao[5†], Tsuyoshi Okita[6] and Daniel Roggen[3†]*

[1]Centre for Intelligent Sensing, Queen Mary University of London, London, United Kingdom, [2]Faculty of Electronic Engineering and Information Technology, Ss. Cyril and Methodius University, Skopje, North Macedonia, [3]Wearable Technologies Lab, University of Sussex, Brighton, United Kingdom, [4]Universidad Nacional Abierta y a Distancia, Bogotá, Colombia, [5]College of Information Science and Engineering, Ritsumeikan University, Shiga, Japan, [6]Kyushu Institute of Technology, Kitakyushu, Japan

The Sussex-Huawei Locomotion-Transportation (SHL) Recognition Challenges aim to advance and capture the state-of-the-art in locomotion and transportation mode recognition from smartphone motion (inertial) sensors. The goal of this series of machine learning and data science challenges was to recognize eight locomotion and transportation activities (Still, Walk, Run, Bus, Car, Train, Subway). The three challenges focused on time-independent (SHL 2018), position-independent (SHL 2019) and user-independent (SHL 2020) evaluations, respectively. Overall, we received 48 submissions (out of 93 teams who registered interest) involving 201 scientists over the three years. The survey captures the state-of-the-art through a meta-analysis of the contributions to the three challenges, including approaches, recognition performance, computational requirements, software tools and frameworks used. It was shown that state-of-the-art methods can distinguish with relative ease most modes of transportation, although the differentiating between subtly distinct activities, such as rail transport (Train and Subway) and road transport (Bus and Car) still remains challenging. We summarize insightful methods from participants that could be employed to address practical challenges of transportation mode recognition, for instance, to tackle over-fitting, to employ robust representations, to exploit data augmentation, and to exploit smart post-processing techniques to improve performance. Finally, we present baseline results to compare the three challenges with a unified recognition pipeline and decision window length.

Keywords: activity recognition, context-aware computing, deep learning, machine learning, mobile sensing, transportation mode recognition

OPEN ACCESS

Edited by:
*Jamie A Ward,
Goldsmiths University of London,
United Kingdom*

Reviewed by:
*Vito Janko,
Institut Jožef Stefan (IJS), Slovenia
Kai Kunze,
Keio University, Japan*

***Correspondence:**
*Lin Wang
lin.wang@qmul.ac.uk*

†ORCID:
*Lin Wang
https://orcid.org/0000-0001-8095-9518
Hristijan Gjoreski
https://orcid.org/0000-0002-0770-4268
Mathias Ciliberto
https://orcid.org/0000-0001-9550-7637
Kazuya Murao
https://orcid.org/0000-0002-3777-8399
Daniel Roggen
https://orcid.org/0000-0001-8033-6417*

Specialty section:
*This article was submitted to
Mobile and Ubiquitous Computing,
a section of the journal
Frontiers in Computer Science*

Received: *24 May 2021*
Accepted: *30 July 2021*
Published: *21 September 2021*

Citation:
Wang L, Gjoreski H, Ciliberto M, Lago P, Murao K, Okita T and Roggen D (2021) Three-Year Review of the 2018–2020 SHL Challenge on Transportation and Locomotion Mode Recognition From Mobile Sensors. Front. Comput. Sci. 3:713719. doi: 10.3389/fcomp.2021.713719

1 INTRODUCTION

The mode of transportation or locomotion of a user is an important contextual information and includes things such as the knowledge of the user walking, running, riding a bicycle, taking a bus, driving a car and others (Engelbrecht et al., 2015). This contextual information enables a variety of applications, for instance, to monitor the daily activity and the health condition of the user; to monitor the impact of an individual's travel behaviour on the environment; to adapt the service provided to the user intelligently based on the context information (Froehlich et al., 2009; Johnson and Trivedi, 2011; Brazil and Caulfield, 2013; Cottrill et al., 2013; Castignani et al., 2015; Mukhopadhyay, 2015; Anagnostopoulou et al., 2018).

To date, significant work has been devoted to the recognition of locomotion and transportation modes from the Global Positioning Systems (GPS) data available on smartphones. GPS data has clear advantages, such as providing exact location which can be correlated to road and rail maps in addition to providing speed and heading. However they also have drawbacks: they tend to be particularly power hungry, do not work well indoors, and they often do not provide sufficiently granular information to distinguish between some modes of transportation[1]. This survey is not concerned with GPS-based recognition and we refer the interested reader to the review (Gong et al., 2014) and to recent work such as (Dabiri and Heaslip, 2018; Guo et al., 2020) on this topic.

Today's smartphones are equipped with a variety types of sensors: in addition to GPS, they include motion sensors (i.e., inertial sensors) comprising accelerometer, gyroscope, magnetometer from which device orientation can be inferred, sound, vision and others, which can be used to identify user activities and the context in which they occur. In comparison to GPS, smartphone motion sensors are lower power sensors[2]. They can provide rich information about the phone movement and therefore the motion of the user which can be analyzed to infer the user's activities and the context in which they occur. General surveys on mobile phone sensing and their applications to activity recognition are available in (Vaizman et al., 2017) and (Lane et al., 2010).

Numerous machine learning approaches have been proposed to recognize the transportation mode of users from the smartphone motion sensors (Biancat and Brighenti, 2014; Xia et al., 2014; Yu et al., 2014). However, most research teams evaluate the performance of their algorithms with self-collected datasets and self-defined recognition tasks, which differ in terms of sensor modalities and processing latencies. This inconsistency makes it very difficult to compare the performance of different methodologies systematically, and thus hinders the progress in the field. To bridge the gap and to encourage reproducible research, we organized three successive academic challenges in the years 2018–2020 that aim to recognize from smartphone motion sensor data eight modes of locomotion and transportation (the activities include: being still, walking, running, cycling, driving a car, being in a bus, train or subway)[3].

Transportation mode recognition mainly faces three types of challenges: time-independent recognition, position-independent recognition, and user-independent recognition (Wang et al., 2019). Time-independent recognition means that a system which has been trained to recognize these activities is expected to keep working over a long period of time, despite slight variations in the way a mobile phone is carried day after day (see examples in **Figure 6**). Position-independent recognition means that an activity recognition system should work when the mobile device is placed at distinct on-body locations, such as when it is placed in a trouser's or shirt's pocket, held in the hand, or stored in a backpack or handbag (see examples in **Figure 7**). Finally, user-independent recognition means that, in order to realize a convenient product, the system should work equally well on every user, without the need for user-specific training (see examples in **Figure 8**). The three SHL events are designed based on the Sussex-Huawei Locomotion-Transportation (SHL) dataset (Gjoreski et al., 2018a; Wang et al., 2019), and focus on these three evaluation scenarios, respectively. The challenges were open during a 6–8 week period over the summer. In total, we received 48 submissions involving 201 scientists over the 3 years. These submissions lead to 45 proceeding publications in total[4] (Wang et al., 2018; Wang et al., 2019; Wang et al., 2020).

This paper introduces the challenge protocols and summarizes the contribution to the three challenges. It surveys the current state of the art through a meta-analysis of the 48 contributions, including approaches, recognition performance, computational requirements, software tools and frameworks used. We observed a growing number of submissions using deep learning over the years compared to classical machine learning. While the best results obtained with deep learning outperformed those obtained with classical machine learning, a large proportion of submissions using deep learning reached lower performance than classical approaches. This reflects the difficulty in effectively employing deep learning with multimodal data in a time-limited challenge. It was shown that state-of-the-art methods can distinguish with relative ease most modes of transportation, although the differentiating between subtly distinct activities, such as rail transport (Train and Subway) and road transport (Bus and Car) still remains challenging. Another difficulty we observed among participants was to accurately employ the training data to predict performance on the test data, despite the use of cross-validation strategies.

In this paper we summarize insightful methods from participants that could be employed to address practical challenges of transportation mode recognition, for instance, to tackle over-fitting, to employ robust representations, to exploit data augmentation, and to exploit smart post-processing techniques to improve performance. Finally, we present baseline results to compare the three challenges with a unified recognition pipeline and decision window length. It was shown that the difficulty levels of the three challenges can be ranked as: SHL 2018 being the easiest, followed by SHL 2020, and SHL 2019 being the hardest.

[1] For instance, cycling, driving a car, or being in a bus may give similar GPS traces and speed patterns in heavy urban traffic.

[2] The TDK/Invensense ICM-20948, which is a state of the art Inertial Measurement Unit (IMU) comprising in a single chip accelerometer, magnetometer and gyroscope uses 5.6 mW when continuously on and sampling data at 100 Hz. A state of art low-power GPS such as the uBlox ZOE-M8B uses 72 mW when continuously on and sampling position at 1Hz, or 15mW in a lower-power "super-efficient" mode which trades off accuracy for power. While this includes the power needed for the antenna amplification circuitry, this is still almost three times more than an IMU.

[3] http://www.shl-dataset.org/challenges/

[4] Three submissions withdrew their papers in the final stage.

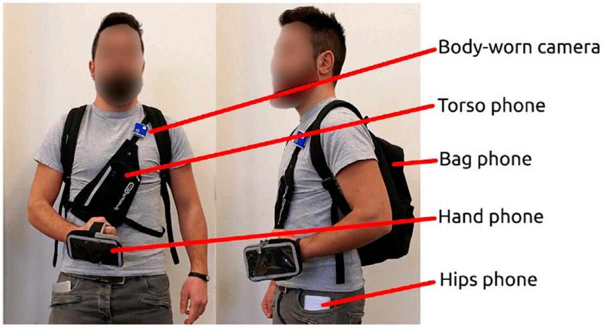


FIGURE 1 | A participant carrying four smartphones and a wearable camera when collecting the dataset. The camera was used to ensure high quality annotation of the modes of transportation and locomotion.

TABLE 1 | Sensor modalities (sampling rate) in the complete SHL dataset.

1. Accelerometer (100 Hz)
2. Gyroscope (100 Hz)
3. Magnetometer (100 Hz)
4. Linear acceleration (100 Hz)
5. Orientation (100 Hz)
6. Gravity (100 Hz)
7. Ambient light (100 Hz)
8. Ambient pressure (100 Hz)
9. Google API for activity recognition (1 Hz)
10. Battery level and temperature (1 Hz)
11. Mobilephone cell reception (1 Hz)
12. WiFi reception (1 Hz)
13. GPS satellite reception (1 Hz)
14. GPS location (1 Hz)
15. Audio (48 kHz)
16. Video (1/30 Hz)

The paper is organized as follows. In **Section 2** we give a review of the dataset and protocols used in the three challenges. In **Section 3** we analyze the performance from the participants contributing to the three challenges. In **Section 4** we discuss insightful methods contributed from the participants. In **Section 5** we present baseline results from the organization committee. Finally, we draw conclusions in **Section 7**.

2 DATASET AND PROTOCOL

2.1 SHL Dataset

The SHL dataset was collected over a period of about 7 months (161 days in total) in the year 2017 by three participants (named User1, User2 and User3) participating in eight locomotion and transportation activities (i.e. being still, walking, running, cycling, driving a car, being in a bus, train or subway)[5] in real life in the south-east of United Kingdom including London (Gjoreski et al., 2018a). Each participant carried four Huawei Mate 9 smartphones at four body positions simultaneously: in the hand, at the torso (e.g., akin to a jacket pocket), in the hip pocket, in a backpack or handbag, as shown in **Figure 1**. **Table 1** lists the 16 sensor modalities recorded by the smartphone system. The complete SHL dataset is comprised of annotated data up to 2,812 h, which correspond to a travel distance of 16,732 km. The dataset is considered the largest and public-available dataset in this research field. **Figure 2** illustrates the amount of data in each activity conducted by the three participants in the complete SHL dataset. **Table 2** lists the basic information of the three users. A detailed description of

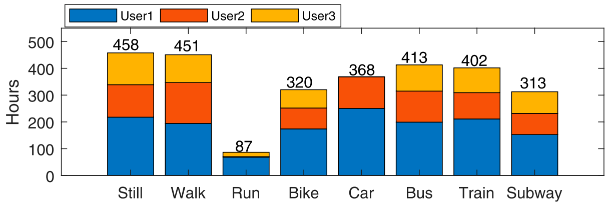


FIGURE 2 | The duration of each of the eight transportation activities by the three participants in the complete SHL dataset. The total duration of the annotated data is 2,812 h.

TABLE 2 | Three users in the complete SHL datasets.

	Gender	Age	Days of data collection
User1	Male	20–40	85
User2	Male	20–40	43
User3	Female	20–40	33

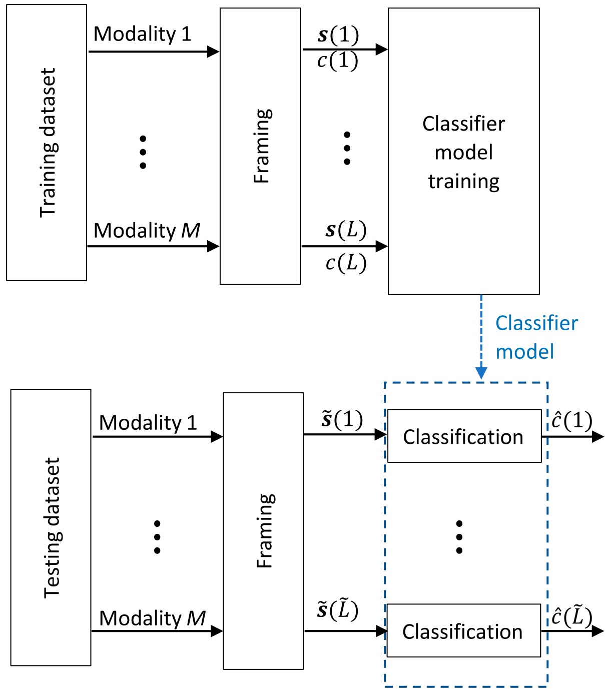


FIGURE 3 | Generic formulation of the transportation mode recognition problem: a recognition pipeline is developed and optimized on a training set, and then evaluated on a distinct testing set. The selection of sensor modalities and the partitioning of the data in the training and testing sets were provided by the challenge organizers.

[5] We refer to them as Still, Walk, Run, Bike, Car, Bus, Train, and Subway for short in the following figures and text.

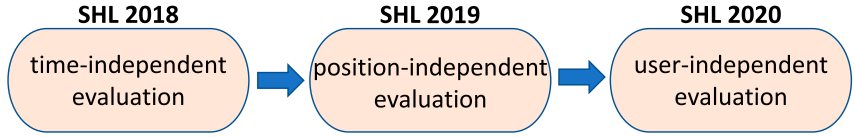

SHL 2018
time-independent evaluation
SHL 2019
position-independent evaluation
SHL 2020
user-independent evaluation

FIGURE 4 | Evolution of the three SHL challenges, which focused on time-independent, position-independent and user-independent evaluation, respectively.

the dataset collection protocol and procedure can be found in (Gjoreski et al., 2018a).

2.2 Recognition Task Overview

While focusing on various evaluation aspects, the recognition tasks addressed in the three challenges can be generalized to develop a recognition system based on the training dataset and then employ this system to recognize the mode of transportation from the multimodal sensor data in the testing dataset. **Figure 3** depicts this formulation of the problem.

The sensor data in the training set is segmented into L frames: $\{\boldsymbol{s}(1), \ldots, \boldsymbol{s}(l), \ldots, \boldsymbol{s}(L)\}$, where $\boldsymbol{s}(l)$ denotes the data in the l-th frame. Each frame contains the data from M modalities, i.e.

$$\boldsymbol{s}(l) = [\boldsymbol{s}_1(l), \ldots, \boldsymbol{s}_m(l), \ldots, \boldsymbol{s}_M(l)] \tag{1}$$

and each modality contains T_m data samples, i.e. $\boldsymbol{s}_m(l) = [s_m^l(1), \ldots, s_m^l(T_m)]$. The labels of these frames, $\{c(1), \ldots, c(l), \ldots, c(L)\}$, are provided to the participants and indicate which activity takes place in each frame.[6] This allows to train a classifier model.

In the testing dataset, the sensor data is also segmented into $\tilde{L}$ frames: $\{\tilde{s}(1), \ldots, \tilde{s}(L)\}$, however labels are hidden from participants. The challenge consists in identifying the transportation or locomotion classes with the trained classifier model, i.e. to estimate the labels $\{\hat{c}(1), \ldots, \hat{c}(\tilde{L})\}$.

To evaluate the recognition performance, we use the F1 score averaged over all the activity classes. Let $\boldsymbol{M}$ be the confusion matrix, where the (i, j)-th element M_{ij} gives the number of frames originally from class i but was classified as class j. Suppose we have $C = 8$ classes, the F1 score can be computed as:

$$\text{recall}_i = \frac{M_{ii}}{\sum_{j=1}^{C} M_{ij}}, \quad \text{precision}_j = \frac{M_{jj}}{\sum_{i=1}^{C} M_{ij}}, \tag{2}$$

$$F1 = \frac{1}{C}\sum_{i=1}^{C} \frac{2 \cdot \text{recall}_i \cdot \text{precision}_i}{\text{recall}_i + \text{precision}_i}. \tag{3}$$

The three challenges use various subsets of the complete SHL dataset, and differ in terms of the definition of the training and testing dataset. As shown in **Figure 4**, the three challenges focused on time-independent, position-independent and user-independent evaluation, respectively, although the latter two types of evaluation naturally entail the first one. They all used the raw data from the following 7 sensors (**Table 3**): accelerometer (3 channels), gyroscope (3 channels), magnetometer (3 channels), linear acceleration (3 channels), gravity (3 channel), orientation (4 channels), and ambient pressure (1 channel). The sampling rate of all these sensors is 100 Hz.

2.3 Time-independent Recognition (SHL 2018 Challenge)

The SHL challenge 2018 focused on time-independent evaluation, and used the data recorded by one user (User1) with the phone at the hip pocket position (Hips). The challenge used data recorded in 82 days (5–8 h per day), which is split into training (62 days, 272 h) and testing (20 days, 95 h)[7]. **Figure 5**A depicts the amount of data in each class activity for training and testing.

The data in both training and testing sets are segmented into frames with a sliding window of 1 minute long and a jump size of 60 s. The order of the frames in both training and testing sets was shuffled randomly to ensure that there was no temporal dependency between neighbouring frames. The objective to impose an upper limit on the recognition latency, with the maximum frame size used by the participants to be 1 minute. The temporal order of the frames in the training set was disclosed during the challenge. However, the original order of the testing frames was hidden from the participants during the challenge, but released after that.

Table 3 specifies the data files provided in the challenge. The training dataset provides 21 files corresponding to 20 data channels (from 7 sensor modalities) plus the label and frame order. The testing set provides 20 files corresponding to 20 data challenges from 7 sensor modalities, but without the label files. which is similar to the training set but without the label and the frame order. All the data are provided as plain text files in ASCII format. The total sizes of the files are 5.5 and 1.9 GB for the training and testing set, respectively. In the training set, the data in each sensor channel corresponds to a matrix of size 16,310 × 6,000, which contains 16,310 frames each with 6,000 data samples (i.e. 1 minute long at sampling rate 100 Hz). The label file contains a matrix of the same size (16,310 × 6,000), which indicates sample-wise activities. In the testing set, the data file in each sensor channel corresponds to a matrix of size 5,698 × 6,000, which contains 5,698 frames each with 6,000 data samples. The label file will be used by the challenge organizer for

[6] In the three challenges, in each frame the labels are given on a sample-by-sample basis.

[7] The exact dates for the training and testing data is released at the website of the SHL 2018 challenge http://www.shl-dataset.org/activity-recognition-challenge/.

TABLE 3 | Data files provided by the SHL recognition challenges 2018–2020. Position: B—Bag; T—Torso; Hi—Hips; Ha—Hand; Un—Unknown position; U1—User1; U2—User2; U3—User3.

Modality	File	SHL 2018		SHL 2019			SHL 2020		
		Train (Hi)	Test (Hi)	Train (BIT/Hi)	Validation (B/T/Hi/Ha)	Test (Ha)	Train (BIT/Hi/Ha)	Validation (BIT/Hi/Ha)	Test (Un)
	User	U1	U1	U1	U1	U1	U1	U2+U3	U2+U3
Accelerometer	Acc x.txt	✓	✓	✓	✓	✓	✓	✓	✓
	Acc y.txt	✓	✓	✓	✓	✓	✓	✓	✓
	Acc z.txt	✓	✓	✓	✓	✓	✓	✓	✓
Gyroscope	Gyr_x.txt	✓	✓	✓	✓	✓	✓	✓	✓
	Gyr y.txt	✓	✓	✓	✓	✓	✓	✓	✓
	Gyr_z.txt	✓	✓	✓	✓	✓	✓	✓	✓
Magnetometer	Mag_x.txt	✓	✓	✓	✓	✓	✓	✓	✓
	Mag_y.txt	✓	✓	✓	✓	✓	✓	✓	✓
	Mag_z.txt	✓	✓	✓	✓	✓	✓	✓	✓
Linear acc.	LAcc x.txt	✓	✓	✓	✓	✓	✓	✓	✓
	LAcc_y.txt	✓	✓	✓	✓	✓	✓	✓	✓
	LAcc_z.txt	✓	✓	✓	✓	✓	✓	✓	✓
Gravity	Gra_x.txt	✓	✓	✓	✓	✓	✓	✓	✓
	Gra y.txt	✓	✓	✓	✓	✓	✓	✓	✓
	Gra_z.txt	✓	✓	✓	✓	✓	✓	✓	✓
Orientation	Ori_w.txt	✓	✓	✓	✓	✓	✓	✓	✓
	Ori_x.txt	✓	✓	✓	✓	✓	✓	✓	✓
	Ori y.txt	✓	✓	✓	✓	✓	✓	✓	✓
	Ori_z.txt	✓	✓	✓	✓	✓	✓	✓	✓
Pressure	Pressure.txt	✓	✓	✓	✓	✓	✓	✓	✓
Label	Label.txt	✓	×	✓	✓	×	✓	✓	×
Dimension of each file (row x column)		16,310 × 6,000	5,698 × 6,000	196,072 × 500	12,177 × 500	55,811 × 500	196,072 × 500	28,789 × 500	57,573 × 500
Total number of files		21	20	63	84	20	84	252	20
Total size of all the files (GB)		5.5	1.9	57.6	4.8	5.4	76.9	11.3	5.6

performance evaluation, and thus was hidden from the participants until the end of the challenge.

In the training set, each sensor data file contains a matrix of size 16,310 lines × 6,000 columns, which corresponds to 16,310 frames each containing 6,000 data samples (i.e. 1-min data at sampling rate 100 Hz). The label file contains a matrix of the same size (16,310 × 6,000), which indicates sample-wise activities. In the testing set, each sensor data file contains a matrix of size 5,698 × 6,000, corresponding to 5,698 frames each containing 6,000 samples. The label file will be used by the challenge organizer for performance evaluation, and thus was hidden from the participants until the end of the challenge.

Thirty-five teams registered their interests in the initial advertising stage. Each team would have maximally one and half months (1 June–July 15, 2018) to work on the challenge task and develop its method. Eventually, we received 19 submissions from 17 teams (two teams each contributed two submissions) by the challenge deadline (July 20, 2018).

2.4 Position-independent Recognition (SHL 2019 Challenge)

The SHL challenge 2019 focused on position-independent activity recognition. It used the data recorded by one user (User1) with smartphones at four body positions simultaneously (see **Figure 1**). The challenge data was recorded in 82 days (5–8 h per day), which is split into training, testing and validation. Specifically, the training set contains 59 days of data recorded at three positions (Hips, Torso and Bag); the testing set contains 20 days of data recorded at the new position (Hand); the validation set contains 3 days of data recorded at all the four positions[8]. The rationale of having validation data is to help participants better train the classification model. In total, we have 271 × 3 h of training data, 77 h of testing data and 17 × 4 h of validation data, respectively. Here × 3 and × 4 refer to data at three and four locations, respectively. **Figure 5B** depicts the amount of data in each class activity in the training, validation and testing sets.

The data in the training, testing and validation sets were segmented into frames with a sliding window of 5 s long and a jump size of 5 s. The objective is to impose an upper limit on the recognition latency, with the maximum frame size used by the challenge participants to be 5 s, which can benefit real-time interactions. The frames in the training set are temporally consecutive. The frames in the testing and validation sets are shuffled randomly. The original order of the frames in these two sets remains confidential until the end of the challenge.

[8] The exact dates for the training and testing data is released at the website of the SHL 2019 challenge http://www.shl-dataset.org/activity-recognition-challenge-2019/.

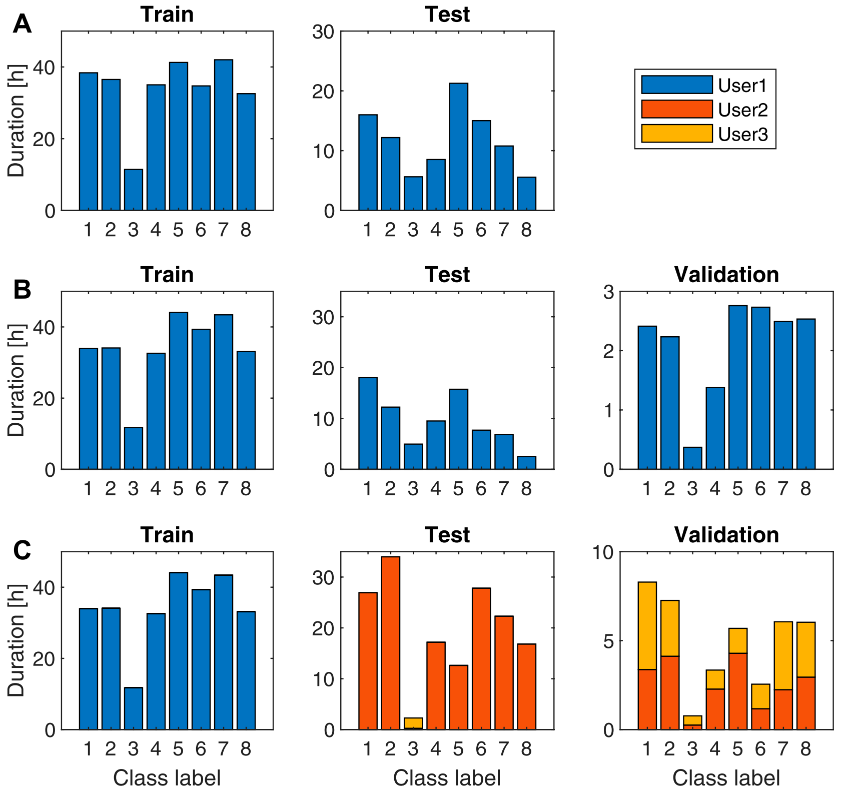


FIGURE 5 | Amount of data for each of the eight transportation activities for the three challenges. **(A)** SHL 2018. **(B)** SHL 2019. **(C)** SHL 2020. The eight class activities are: 1—Still; 2—Walk; 3—Run; 4—Bike; 5—Car; 6—Bus; 7—Train; 8—Subway.

Table 3 specifies the data files provided in the challenge. The training, testing and validation sets contain data collected at various body positions. In the training and validation set, each position contains 21 files corresponding to 20 data files (from 7 sensor modalities) and 1 label file. In the testing set, each position contains 20 files of sensor data, but excluding the label file.

In the training set, the data file of each sensor channel corresponds to a matrix of size 196,072 × 500, which contains 196,072 frames each with 500 data samples (i.e. 5-s data at sampling rate 100 Hz). The label file contains a matrix of the same size (196,072 × 500), which indicates sample-wise activities. In the validation set, the data file of each sensor channel corresponds to a matrix of size 12,177 × 500 and the label file is of the same size. In the testing set, the data file in each sensor channel contains a matrix of size 55,811 × 500. The label file will be used by the challenge organizer for performance evaluation, and thus was hidden from the participants until the end of the challenge. All the data are provided as plain text files in ASCII format. The total sizes of the training, testing and validation set are 57.6, 5.4, and 4.8 GB, respectively.

Twenty-five teams registered their interests in the initial advertising stage. Each team would have maximally one and half months (15 May–June 30, 2019) to work on the challenge task and develop its method. Eventually, we received 14 submissions from 14 teams by the submission deadline (June 30, 2019).

2.5 User-independent Recognition (SHL 2020 Challenge)

The SHL challenge 2020 focused on user-independent evaluation. We have a "Train" user and a "Test" user. The "Train" user uses the data from User1. The "Test" user combines the data from User2 and User3 in order to get balanced testing data, as none of the two users are able to participate in all the eight transportation activities (User2 can not drive a car and User3 is deficient in running). The challenge data is divided into three parts: training, testing and validation. The training set contains 59 days of data collected by the "Train" user at the four body positions (Bag, Hips, Torso, Hand). The testing set contains the data collected by the "Test" user at the Hips position (this position was unknown to the participants during the challenge). The validation set contains 6 days of data collected by the "Test" user at the four body position[9]. Similar to SHL 2019, the validation data is to help participants better train the classification model. In total, we have training data of 272 × 4 h, testing data of 160 h, and validation data of 40 × 4 h, respectively. **Figure 5**C depicts the amount of data in each transportation activity in the training, testing, and validation sets.

[9] The exact dates for the training and testing data are released at the website of the SHL 2020 challenge http://www.shl-dataset.org/activity-recognition-challenge-2020/.

Similar to SHL 2019, The data in the training and validation sets were segmented into frames with a sliding window of 5 s long and a jump size of 5 s. The frames in the training and validation sets are consecutive in time. The data in the testing set was also segmented into frames 5 s long, and the frames are permuted randomly. The original order of the frames in the testing dataset remained confidential until the end of the challenge. In the previous challenge 2019, some teams reconstructed the complete time series by looking at the signal continuity. To prevent this in the challenge 2020, we used a larger jump size (10 s) when segmenting the testing data.

Table 3 specifies the data files provided in the challenge. The training, testing and validation sets contain data collected by various users and at various body positions. In the training and validation set, each position contains 21 files corresponding to 20 data files (from 7 sensor modalities) and 1 label file. In the testing set, each position contains 20 files of sensor data, but excluding the label file.

In the training set, the data file in each sensor channel contains a matrix of size 196,072 × 500, which corresponds to 196,072 frames each with 500 data samples (i.e. 5-s data at sampling rate 100 Hz). The label file contains a matrix of the same size (196,072 × 500), which indicates sample-wise activities. In the validation set, the data file in each sensor channel contains a matrix of size 28,789 × 500. The label file is of the same size as the data file. In the testing set, each sensor file contains a matrix of size 57,573 × 500. The label file will be used by the challenge organizer for performance evaluation, and thus was hidden from the participants until the end of the challenge. All the data are provided as plain text files in ASCII format. The total sizes of the training, testing and validation set are 76.9, 5.6, and 11.3 GB, respectively.

Thirty-three teams registered their interests in the initial advertising stage. Each team would have maximally two and half months (05 April–June 25, 2020) to work on the challenge task and develop its method. Eventually, we received 15 submissions from 15 teams by the submission deadline (June 25, 2020).

2.6 Exemplary Samples

Figure 6 depicts the exemplary samples of accelerometer data (X-, Y- and *Z*-axis) in SHL 2018. The data is collected by User1 at Hips position during the eight transportation and locomotion activities. The three activities Walk, Run and Bike, which involve intense human actions, show much stronger vibration than the other five activities. These three activities have also shown certain cyclic properties due to the rhythmic action of human. The activities Car and Bus show stronger vibration than the remaining three activities Still, Train and Subway. Based on the observation, while some activities clearly exhibit distinct signatures, it is still difficult to identify each activity precisely solely based on their visual appearance. In SHL 2018, even though we consider a fixed phone position in a trouser's pocket, there is variability in the way the phone may be placed and oriented, and due to change in clothing. This can also be observed by comparing the data value at three-axes across the eight activities that occur at different time instances. For instance, in **Figure 6** the Y-axis value reads at −10 during the Bus activity and at 10 during the Subway activity: this indicates an upside-down placement of the phone.

Figure 7 depicts the exemplary accelerometer data (X-, Y-, and *Z*-axis) for the eight transportation and locomotion activities in SHL 2020. The data is collected by User1 at the four positions (Bag, Torso, Hips, and Hand) in the same time interval. While being collected at the same time, the data at the four positions appear different, due to different phone placement and orientation. During Bike activity, the Hips phone moving together with the human thigh presents more evident cyclic pedalling behaviour than the phones at other positions. Due to the engagement between hand and phone, the data at the Hand positions appears noisier than the other three positions. This highlights the challenge of position-independent recognition, where the testing data is at the Hand position while the training data is provided at the other positions.

Figure 8 depicts the exemplary accelerometer data (X-, Y-, and *Z*-axis) for the eight transportation and locomotion activities in SHL 2020. The data is collected by three users (User1, User2, and User3) at the Hips position. The data presents obvious differences across the three users, due to different phone orientations and diverse human behaviours during transportation. This observation also highlights the challenge of user-independent recognition, where the model is trained for User1 and tested on User2 and User3.

3 RESULTS AND ANALYSIS

3.1 Overview of the Results

Figure 9A depicts the results of the submissions to the three challenges. The submissions to each challenge are ranked according to their actual performance (F1 score) on the testing data. As each participant team employs a distinct cross-validation strategy, we requested each team to predict its testing performance using the data that are made available to them (i.e., the training data and the validation data). For ease of comparison, in **Figure 9A** the predicted performance is plotted together with the actual performance on the testing data. **Figure 9B** gives the confusion matrices obtained by the top team in each challenge, i.e., (Gjoreski et al., 2018b) (SHL 2018), (Janko et al., 2019), (SHL 2019), and (Zhu et al., 2020) (SHL 2020). **Figure 9A** also indicates the baseline performance achieved by the organization committee (**Section 5**).

For SHL 2018 (time-independent recognition), the actual performance of the 19 submissions varies from 53.2 to 93.9% on the testing set. Among these participants, two submissions achieve an F1 score above 90%, eight submissions achieve an F1 score between 80 and 90%, five submissions achieve an F1 score between 70 and 80%, and four submissions between 50 and 70%. The best performance (F1 score of 93.9%) is reported by (Gjoreski et al., 2018b), which employed an ensemble of classifiers, including classical machine-learning and deep-learning models, to do the prediction, and then smoothed the results with a post-filter (hidden Markov model).

For SHL 2019 (position-independent recognition), the actual performance of the 14 submissions varies from 31.5 to 78.4% on

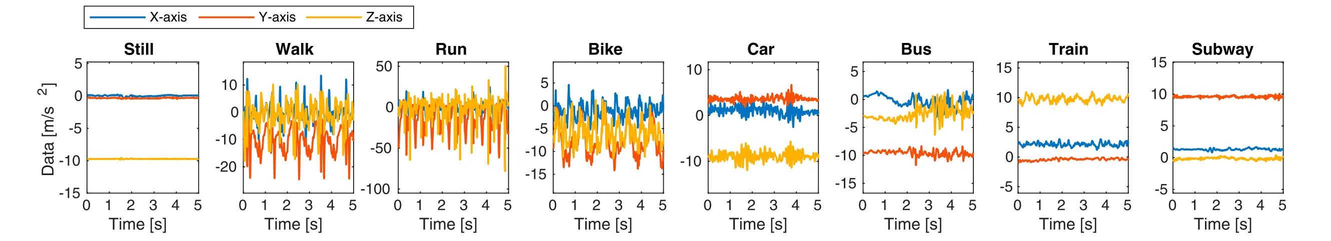


FIGURE 6 | Exemplary samples of accelerometer data in SHL 2018. The data is collected by User1 at Hips position.

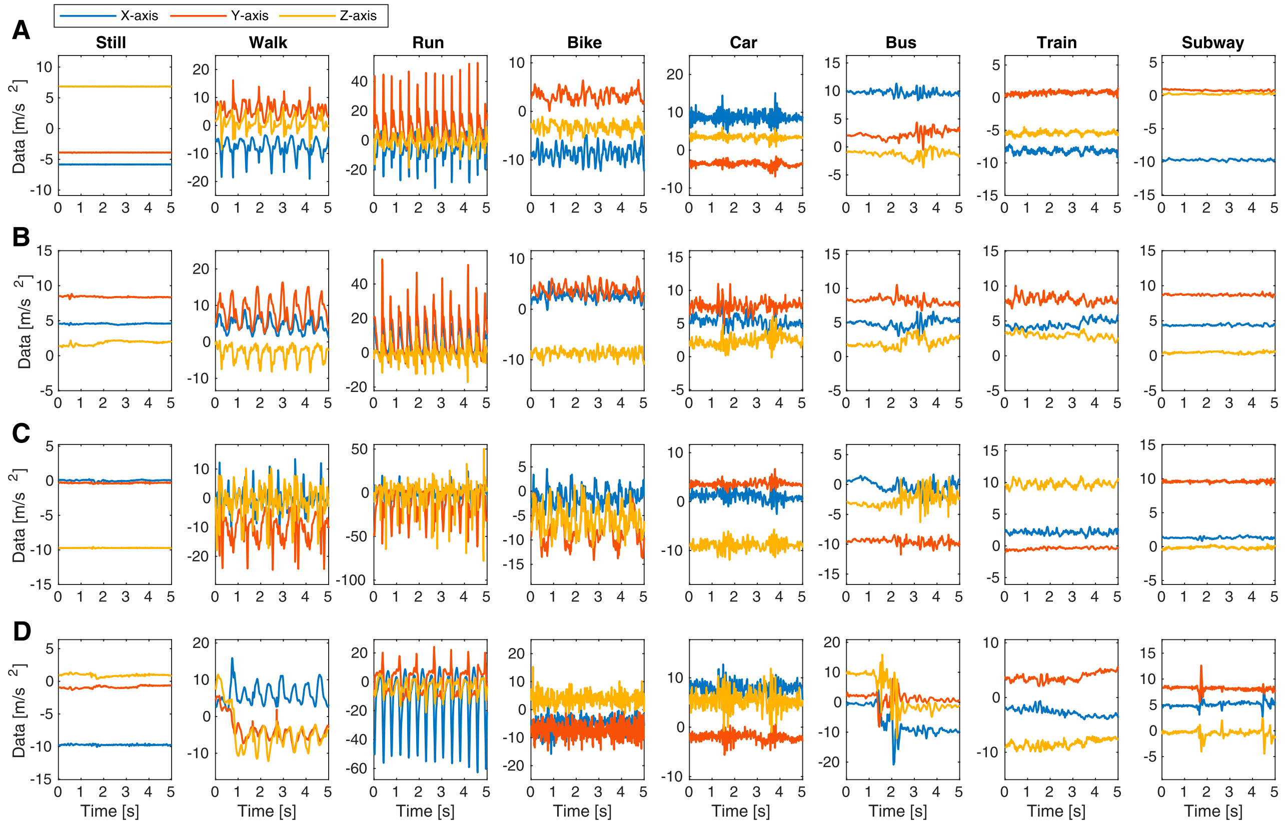


FIGURE 7 | Exemplary samples of accelerometer data in SHL 2019. The data is collected by User1 at four body position in the same time interval: **(A)** Bag; **(B)** Torso; **(C)** Hips; **(D)** Hand.

the testing set. Among these participants, 0 submission achieves an F1 score above 80%, three submissions achieve an F1 score above 70%, four submissions achieve an F1 score between 60 and 70%, five submissions achieve an F1 score between 50 and 60%, and one submission below 50%. The best performance (F1 score of 78.3%.) is reported by (Janko et al., 2019), which was based on cross-position transfer learning. In this method, two classification models are trained: one model is used to classify all the data samples and another model is used to re-classify the data samples that are classified as vehicle or still activities previously. The first model is trained based on data at the three positions (Bag, Hips, Torso) in the training and validation set; while the second model is trained based on the data at the Hand position, which is provided in the validation set.

For SHL 2020 (user-independent recognition), the actual performance of the 15 submissions varies from 17.8 to 88.5% on the testing set. Among these participants, one submission achieves an F1 score above 80%, four submissions achieve an F1 score between 60 and 70%, three submissions achieve an F1 score between 50 and 60%, and four submissions below 50%. The best performance (F1 score of 88.5%.) is reported by (Zhu et al., 2020), which, used a 1D DenseNet model for the classification task. The method employed a pre-processing strategy that converts the multimodal sensor data that are measured in a phone-centered

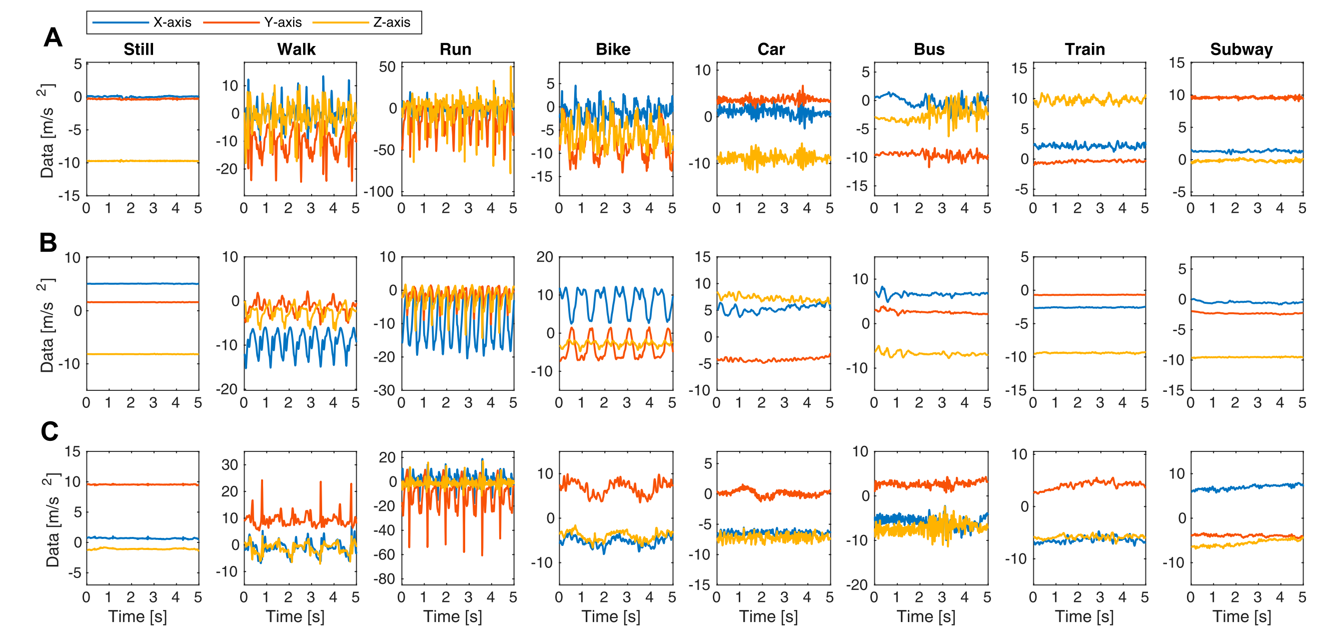


FIGURE 8 | Exemplary samples of accelerometer data in SHL 2020. The data is collected by three users at Hips position: **(A)** User1; **(B)** User2; **(C)** User3.

coordinate system to a human-centered coordinate system before feeding them to the deep neural network.

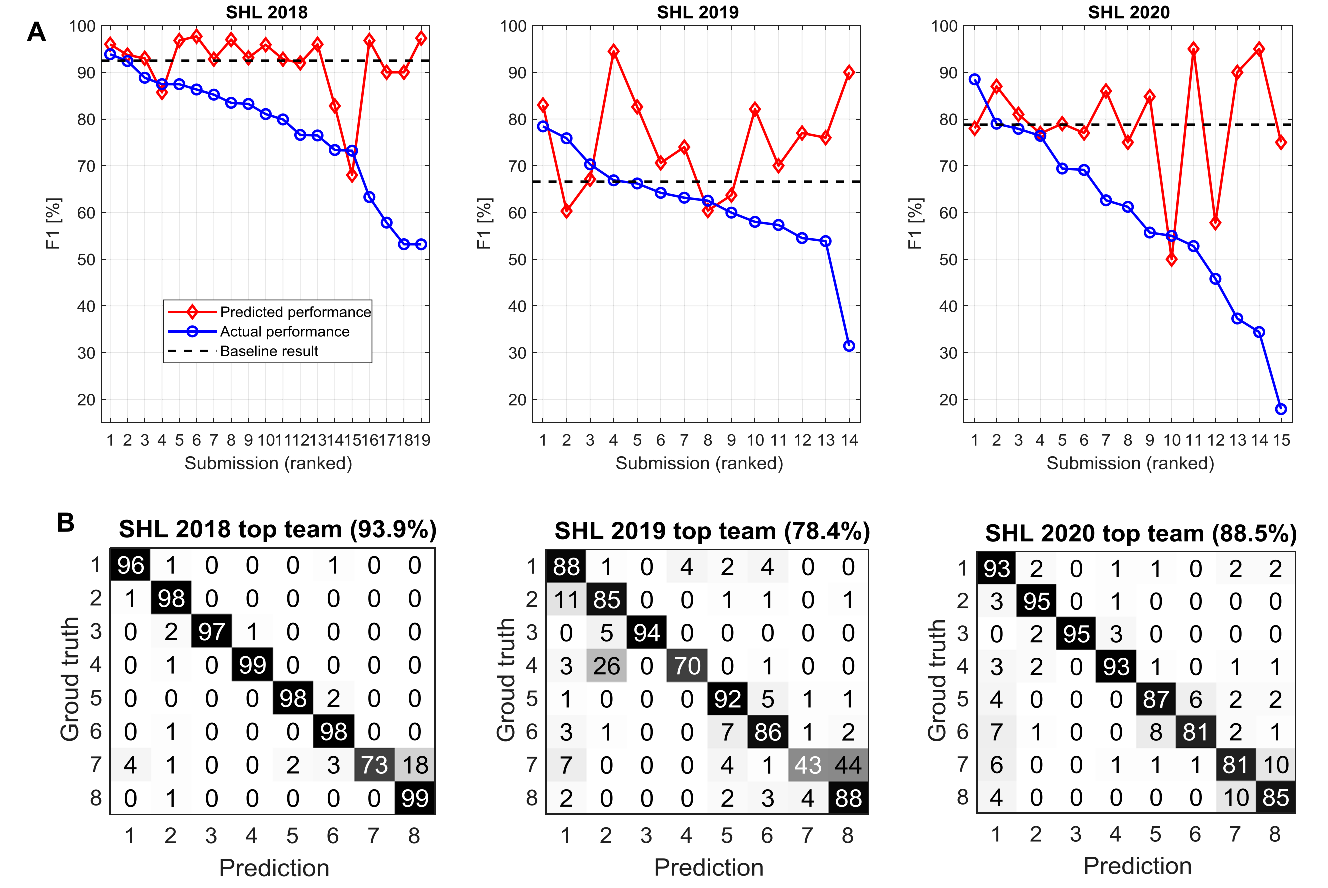
3.2 Performance Analysis

To avoid the influence of outliers, we take the top 10 results from each challenge for comparison. For each submission, we compute a confusion matrix and interpret the diagonal elements of the matrix as the recognition accuracy of each class activity. In **Figure 10A** we show a box-plot of the recognition accuracy of each activity achieved by the top 10 submissions in each challenge, and in **Figure 10B** we depict the averaged confusion matrix. The technical details of the top 10 submissions in each challenge are summarized in **Table 4**.

Some consistent observations can be made throughout the three challenges. From the box-plot we observe that the first four transportation activities (Still, Walk, Run, and Bike) are better identified in comparison to another four activities (Car, Bus, Train, and Subway). The movement of the smartphones during walking, running and cycling is more intense than when the user is sitting or standing in the bus, car, train or subway. This may contribute to making the former four more distinctive than the latter four. In SHL 2018, Train is the most challenging activity to identify, followed by Subway and Car. In SHL 2019, Car is the most challenging activity to identify, followed by Train and Subway. In SHL 2020, Subway is the most challenging class to identify, followed by Bus and Subway. From the confusion matrices of all the three challenges, mutual confusion can be clearly observed between motor vehicles (i.e. Bus versus Car), and also between rail vehicles (i.e. Train versus Subway). This is possibly because the smartphones carried by the users show similar motion patterns during vehicle transportation, with two activities being road transportation and two others being rail transportation. From the confusion matrices, confusion is also observed between the Still activity and four vehicle activities (Car, Bus, Train and Subway). In particular, some vehicle activities are misclassified as Still. It may occur that the smartphones become motionless when a vehicle stops during travel.

The testing performance reported in SHL 2018 (the highest F1 93.4% and the average F1 86.9%) is much higher than the performance reported in SHL 2019 (the highest F1 78.4% and the average F1 66.5%) and SHL 2020 (the highest 88.5% and the average 69.5%). There are mainly two reasons for the decreased performance in the latter 2 years. First, SHL 2018 only considered the challenge of temporal variation, while SHL 2019 and 2020 had additional challenges from the variation of the phone positioning and the variation of the user. During the discussions at the challenge workshops (HASCA), it has been widely reported from the participant teams of SHL 2019 and SHL 2020 that the mismatch between the training data (which is recorded at a specific body position/user) and the testing data (which is recorded at a new body position or by a new user) degrades the recognition performance significantly. This is also a challenge in real-life applications. Second, in SHL 2018 the data was segmented into 1-min frames, while in SHL 2019 and 2020 the data was segmented into 5-s frames. It is difficult to apply a post-processing scheme (e.g. sequence modeling or temporal smoothing) within a 5-s decision window in SHL 2019 and SHL 2020. In contrast, it was reported in SHL 2018 that applying post-processing within the 1-min frame can improve the recognition performance remarkably over individual 5-s frames (Wang et al., 2018). For instance, the reference (Wang et al., 2018) achieved a 10 percentage points higher F1 score with temporal smoothing.

The performance reported in SHL 2020 is comparable to the performance reported in SHL 2019. While the top performance in



SHL 2018 top team (93.9%)	1	2	3	4	5	6	7	8
1	96	1	0	0	0	1	0	0
2	1	98	0	0	0	0	0	0
3	0	2	97	1	0	0	0	0
4	0	1	0	99	0	0	0	0
5	0	0	0	0	98	2	0	0
6	0	1	0	0	0	98	0	0
7	4	1	0	0	2	3	73	18
8	0	1	0	0	0	0	0	99

SHL 2019 top team (78.4%)	1	2	3	4	5	6	7	8
1	88	1	0	4	2	4	0	0
2	11	85	0	0	1	1	0	1
3	0	5	94	0	0	0	0	0
4	3	26	0	70	0	1	0	0
5	1	0	0	0	92	5	1	1
6	3	1	0	0	7	86	1	2
7	7	0	0	0	4	1	43	44
8	2	0	0	0	2	3	4	88

SHL 2020 top team (88.5%)	1	2	3	4	5	6	7	8
1	93	2	0	1	1	0	2	2
2	3	95	0	1	0	0	0	0
3	0	2	95	3	0	0	0	0
4	3	2	0	93	1	0	1	1
5	4	0	0	0	87	6	2	2
6	7	1	0	0	8	81	2	1
7	6	0	0	1	1	1	81	10
8	4	0	0	0	0	0	10	85

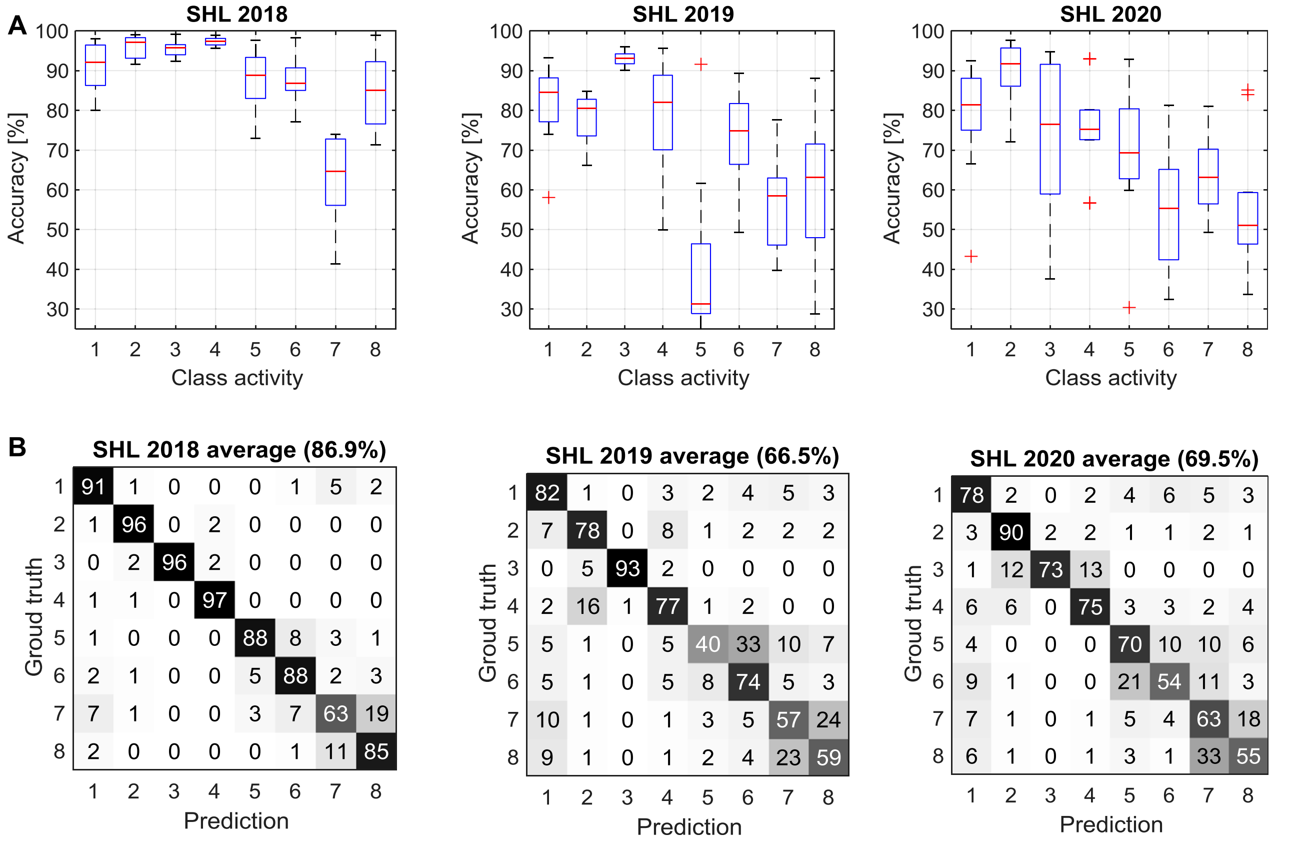
FIGURE 9 | Submission results of SHL 2018–2020. **(A)** Submission ranked based the F1 score for the testing data. The baseline results from the organization committee (**Section 5**) are also indicated. **(B)** Confusion matrix from the top team. The eight class activities are: 1—Still; 2—Walk; 3—Run; 4—Bike; 5—Car; 6—Bus; 7—Train; 8—Subway.

SHL 2020 (highest F1 88.4%) is much higher than the top one in SHL 2019 (highest F1 78.4%), their mean performance (among the top 10 submissions) are close, where SHL 2020 (average F1 69.5%) is only slightly higher than SHL 2019 (average F1 65.5%). On one side, the user variation in SHL 2020 imposes more challenges than SHL 2019, which used the data from the same user. On the other side, performing recognition using the data at the "Hand" phone in SHL 2019 is more difficult than the "Hips" phone in SHL 2020. Combining both factors, it seems reasonable that the two challenges achieve similar performance.

3.3 Machine Learning Algorithms

With the rise of deep learning in the activity recognition community, it is interesting to evaluate the approaches employed by participants to the challenge. The submissions to the SHL challenges can be broadly divided into two categories: classical machine learning (ML) and deep learning (DL). Specifically, there are 11 ML and 8 DL submissions in SHL 2018, 6 ML, and 8 DL submissions in SHL 2019, and 6 ML, and 9 DL submissions in SHL 2020, respectively. While the amount of data is small, this reflects the growing interest in deep learning methods. For ML approaches, most classifiers compute hand-crafted features as input. These hand-crafted features can be roughly divided into time-domain features (statistical information of signal amplitude within a short time window, e.g. mean, variance, median, quantile and auto correlation) and frequency-domain features (e.g. Fourier transform coefficients, energy and subband energy). Due to the diversity of the features employed by the participants, a systematic analysis or grouping is difficult. We refer the interested reader to a recent work (Wang et al., 2019), which reviewed the features for transportation mode recognition. For DL approaches, several types of input are considered, including time-domain raw data, time-frequency spectrogram, hand-crafted features, or a hybrid mixture of them. There is a trend that more teams are choosing to use DL over ML.

We show in **Figure 11A** the box-plot of the F1 scores obtained from these two families in the three challenges. Overall, DL achieves a comparable higher upper bound at SHL 2018 and 2019, and a much larger higher upper bound at SHL 2020. However, the median performance achieved by DL is slightly lower than ML in the three challenges. One likely reason is that DL approaches require more effort on architecture optimization. The degrees of freedom make it more easily possible to deploy a



SHL 2018 average (86.9%)

	1	2	3	4	5	6	7	8
1	91	1	0	0	0	1	5	2
2	1	96	0	2	0	0	0	0
3	0	2	96	2	0	0	0	0
4	1	1	0	97	0	0	0	0
5	1	0	0	0	88	8	3	1
6	2	1	0	0	5	88	2	3
7	7	1	0	0	3	7	63	19
8	2	0	0	0	0	1	11	85

SHL 2019 average (66.5%)

	1	2	3	4	5	6	7	8
1	82	1	0	3	2	4	5	3
2	7	78	0	8	1	2	2	2
3	0	5	93	2	0	0	0	0
4	2	16	1	77	1	2	0	0
5	5	1	0	5	40	33	10	7
6	5	1	0	5	8	74	5	3
7	10	1	0	1	3	5	57	24
8	9	1	0	1	2	4	23	59

SHL 2020 average (69.5%)

	1	2	3	4	5	6	7	8
1	78	2	0	2	4	6	5	3
2	3	90	2	2	1	1	2	1
3	1	12	73	13	0	0	0	0
4	6	6	0	75	3	3	2	4
5	4	0	0	0	70	10	10	6
6	9	1	0	0	21	54	11	3
7	7	1	0	1	5	4	63	18
8	6	1	0	1	3	1	33	55

FIGURE 10 | Average performance of the top 10 teams in SHL 2018–2020. **(A)** Recognition accuracy for each class activity. **(B)** Average confusion matrix. The eight class activities are: 1—Still; 2—Walk; 3—Run; 4—Bike; 5—Car; 6—Bus; 7—Train; 8—Subway. The boxplot illustrates the minimum, first quartile, median, third quartile, and maximum values in each group of data, and the also the outlier values.

DL approach inadequately, for instance with a sub-optimal architecture, or encountering overfitting. In comparison to DL, ML approaches have much less hyper-parameters to optimize, and they have been well studied since longer time ago, which means that there is more expertise available in using these approaches effectively. This reflects that DL is a relatively recent development for activity recognition, and the community is still in the process of gathering the necessary expertise to employ it effectively. This is further compounded by the long training time, which limits the amount of exploration possible in a time-limited challenge.

In SHL 2018, the best performance achieved by DL approaches (F1 93.9%) (Gjoreski et al., 2018b) is only 1.5 percentage points higher than the best one by ML approaches (92.4%) (Janko et al., 2018). In SHL 2019, the best performance by DL (75.9%) (Choi and Lee, 2019) is 2.5 percentage points lower than the best performance by ML (78.4%) (Janko et al., 2019). This is slightly contradictory to the observations in the other 2 years, and is possibly due to the big difference between the traning and the testing data: one collected at Torso, Bag, and Hips positions while another one collected at Hand position. The latent features learned by DL approaches in the source domain do not generalize well to the target domain. ML approaches use hand-crafted features, which can include engineering knowledge, and thus are more robust to handle the difference between the source and the target domain. In SHL 2020, the best DL approach (F1 88.5%) (Zhu et al., 2020) outperforms the best ML approach (F1 77.9%) (Kalabakov et al., 2020) by 10.6 percentage points. However, the boxplot of ML performance has a smaller dynamic range than the boxplot of DL performance. This implies that the hand-crafted features utilized in ML approaches are more robust to user variation while the DL features do not generalize well to this variation.

In **Figures 11B,C** we show the box-plot the computation time reported by the ML and DL submissions. Since different research groups use various computational facilities, it is difficult to make a fair comparison (See **Table 4**). Overall, DL is much more computationally complex than ML, consuming a larger amount of time for training and testing.

Figure 12 summarizes the classifiers used by ML and DL approaches in SHL 2018–2020. The ML approaches mainly employ five types of classifiers: random forest (RF), support vector machine (SVM), extreme gradient boost (XGBoost), multi-layer perceptron neural network with up to two hidden layers (MLP), and ensembles of classifiers (Ensembles). Among these classifiers, RF is the most popular one (8 submissions), followed by XGBoost (6 submissions) and MLP (4 submissions). For the recognition tasks, XGBoost (Janko et al., 2018; Kalabakov

TABLE 4 | Summary of the top 10 submissions to SHL 2018–2020.

Challenge	Approach	Rank	Team	Classifier	Input	Sensor modality	Performance		Computational resource		Time		Implementation		Model size (MB)	References
							Train (%)	Test (%)	CPU	GPU	Train (h)	Test (s)	Language	Library		
SHL 2018	ML	2	JSI-Classic	XGBoost	Features	LAGMOPR	93.7	92.4	4-core@ 3.6 GHz RAM-16G	/	8.5	20	Python	ScikitLearn	43	Janko et al. (2018)
		4	S304	MLP	Features	AGMP	85.7	87.5	24-core@ 2.5 GHz: RAM-64G	/	0.25	50	Java	/	0.035	Widhalm et al. (2018)
		5	Confusion Matrix	RF	Features	LAGMOPR	96.8	87.5	4-core@ 2.5 GHz RAM-8G	/	0.15	32.4	Python	ScikitLearn	1,122	Akbari et al. (2018)
		7	UCLab-Vrano	RF	Features	AGMRP	94.2	85.2	16-core@ 3.4 GHz RAM-64G	Titan V	3	10	Python	ScikitLearn	130	Matsuyama et al. (2018)
		8	Ubi-NUTS	RF	Features	LAGMOPR	97.0	83.5	6-core@ 3.6 GHz RAM-128G	2 × GP100	0.41	7	Python	ScikitLearn	198	Nakamura et al. (2018)
		10	Drifters1	SVM	Features	LGOPR	95.9	81.1	8-core@ 2.4 GHz RAM-30G	/	0.5	900	Java	WEKA	170	Wu et al. (2018)
	DL	1	JSI-Deep	Ensemble (DNN + ML)	Spectrogram + features	LAGMOPR	96.0	93.9	4-core@ 3.3 GHz RAM-16G	GTX 1070	6.5	20	Python	Keras ScikitLearn	500	Gjoreski et al. (2018)
		5	Tesaguri	CNN	Spectrogram	AG	93.0	88.8	6 × (4-core@ 2.5 GHz RAM-8G)	/	90	300	Python	Keras	3	Ito et al. (2018)
		6	Drifters2	DNN	Features	LAGMOPR	97.7	86.3	8-core@ 2.4 GHz RAM-30G	GTX 950M	1	180	Python	Keras	84	Akbari et al. (2018)
		9	UCLab-Nozaki	CNN + LSTM	Raw data	AGMP	93.1	83.2	24-core@ 3.4 GHz RAM-192G	5 GPU	12	600	Python	Pytorch	12	Yuki et al. (2018)
SHL 2019	ML	1	JSI-First	Random Forest	Features	LAGMOPR	83.0	78.4	4-core@ 3.6GHs RAM-16G	/	8.5	20	Python	ScikitLearn	43	Janko et al. (2019)
		4	Jellyfish	XGBoost + MLP	Features	AGMP	94.5	66.9	28-core@ 2.5 GHz RAM-64G	5 × GTX 2080	1.25	50	Python	ScikitLearn Tensorflow	40.54	Lu et al. (2019)
		6	Gradient Descent	Classifier ensembles	Features	LAGMOPR	70.6	64.2	4-core@ 2.5 GHz RAM-8G	?	6.7	556	Python	ScikitLearn Tensorflow	383.1	Ahmed et al. (2019)
		7	S304	MLP ensembles	Features	AGM	74.0	63.2	4-core@ 2.8 GHz RAM-8G	/	1	30	Java	AIT	0.2	Widhalm et al. (2019)
	DL	2	Yonsei-MCML	CNN	Time + Frequency	LAGMOPR	60.3	75.9	4-core@ 4.2 GHz RAM-32G	GTX 1080	17	1,500	Python	Tensorflow	210.9	Choi and Lee, (2019)
		3	We-can-fly	CNN	Time	LAGMPR	67.1	70.3	14-core@ 2.6 GHz RAM-64G	TESLA V100	6	120	Python	Pytorch	11	Zhu et al. (2019)

(Continued on following page)

TABLE 4 | (*Continued*) Summary of the top 10 submissions to SHL 2018–2020.

Challenge	Approach	Rank	Team	Classifier	Input	Sensor modality	Performance		Computational resource		Time		Implementation		Model size (MB)	References
							Train (%)	Test (%)	CPU	GPU	Train (h)	Test (s)	Language	Library		
		5	UESTC_IndRNN	RNN	Frequency	LAGMR	82.6	56.2	10-core@ 2.4 GHz RAM-256G	Titan XP	3	600	Python	Pytorch	34.6	Zheng et al. (2019)
		8	Orange Lab	LSTM	Time	LA	60.4	62.5	8-core@ 2.5 GHz RAM-16G	GTX 1080	5	5	Python	Pytorch	2.1	Alwan et al. (2019)
		9	GanbareAMT	LSTM	Time	AGMP	63.7	50.0	12-core@ 2.2 GHz, RAM-120G	TESLA P100	100.3	21	Python	Keras (Tensorflow)	0.7	Friedich et al. (2019)
		10	TDU-DSML	CNN	Frequency	AG	82.1	58.0	24-core@ 2.2 GHz RAM-128G	2 × GTX 1080	6.7	600	Python	Keras (Tensorflow)	10.2	Ito et al. (2019)
SHL 2020	ML	3	ThirdTime's ACharm	XGBoost	Features	LAGMOPR	81.0	77.9	8-core@ 3.6 GHz RAM-16G	RTX 2060	0.08	15	Python	ScikitLearn	60	Kalabakov et al. (2020)
		6	RED_CIRCLE	RF	Features	LAGMOPR	77.0	59.1	2-core@ 2.3 GHz RAM-13G	/	1	41	Python	ScikitLearn	1825	Siraj and et al. (2020)
		7	ASIA	RF	Features	LAGMP	86.0	62.6	8-core@ 2.3 GHz RAM-16G	/	0.08	1	Python	ScikitLearn	278	Brajesh and Ray, (2020)
		8	MDCA	MLP	Features	AGMOPR	75.0	61.2	8-core@ 1.9 GHz RAM-16G	/	0.03	3	Java	AIT	0.2	Widhalm et al. (2020)
		10	SensingGO	XGBoost	Features	LAGMP	50.0	55.0	12-core@ 2 GHz RAM-128G	/	0.3	180	Python	ScikitLearn	0.7	Tseng et al. (2020)
	DL	1	We-can-fly	CNN	Time	LAGMPR	73.0	88.5	14-core@ 2.6 GHz RAM-128G	Tesla V100	6	120	Python	Pytorch	30	Zhu et al. (2020)
		2	IndRNN	RNN	Features	AGMOP	87.0	79.0	10-core@ 2.4 GHz RAM-256G	Titan XP	18	2,540	Python	Pytorch	43	Zhao et al. (2020)
		4	DSML_TDU	CNN	Features	LGM	67.9	76.4	8-core@ 3.5 GHz RAM-128G	GTX 1080Ti	2	300	Python	Keras (Tensorflow)	103	Yaguchi et al. (2020)
		5	DL_Lock	CNN	Features	LAGM	79.0	59.4	6-core@ 3.5 GHz RAM-32G	RTX 2080 Ti	0,75	16	Python	Keras (Tensorflow)	19.3	Naseeb and Saeedi, (2020)
		9	TDU_BSA	CNN	Frequency	LAGMOP	84.8	55.7	6-core@ 3.2 GHz RAM-32G	RTX 2060	12	60	Python	Keras (Tensorflow)	114	Sekiguchi et al. (2020)

Sensor modalitiy: L—Linear accelerometer; A—Accelerometer; G—Gyroscope; M—Magnetometer; O—Orientation; P—Pressure; R—Gravity.
Input: Time—raw data in the time domain; Frequency—raw data in the freqeucy domain; Spectrogram—raw data in the time-freqeuncy domain; Feature: hand-crafted feature.

et al., 2020) performs the best in SHL 2018 and 2020, RF (Janko et al., 2019) performs the best in SHL 2019 (see **Table 4**).

The DL approaches mainly employ four types of classifiers: deep multi-layer perceptron neural network with more than two hidden layers (DNN), convolutional neural network (CNN), recursive neural network or long-short term memory neural network (RNN), and CNN plus RNN (CNN-RNN). Among these classifiers, CNN is the most popular one (10 submissions), followed by RNN (6 submissions). For the recognition tasks, DNN (Gjoreski et al., 2018b) performs the best in SHL 2018, CNN (Choi and Lee, 2019) performs the best in SHL 2019, and CNN (Zhu et al., 2020) performs the best in SHL 2020 (see **Table 6**). In addition, the employment of adversarial auto-encoder (AAE) (Balabka, 2019) and generative adversarial networks (GAN) (Gunthermann et al., 2020) were reported in SHL 2019 and 2020, respectively, although these two classifiers did not achieve good performance (out of top 10) in the challenges.

3.4 Software Implementation

Figure 13 summarizes the programming languages and libraries used in the three challenges. For ML approaches, Python (15 submissions) is the most popular language, followed by Matlab (4 submissions) and Java (4 submissions). The languages C and R are only sporadically used. Python Scikit-Learn (15 submissions) is the most used library, followed by Matlab Machine Learning Toolbox (4 submissions) and Java AIT (3 submissions). For DL approaches, Python (25 submissions) was the sole programming language. Keras (13 submissions) was the most widely used library, followed by Tensorflow (7 submissions) and Pytorch (7 submissions). Keras is a high-level deep learning library that is built on top of Tensorflow, Microsoft Cognitive Toolkit (CNTK), or Theano. Interestingly, all the Kera submissions used the Tensorflow backend.

4 DISCUSSION

The numerous competition submissions and the discussions during the presentations of the best performing approaches, revealed numerous ideas and techniques on how to tackle the temporal dynamics in the data, the position of the devices, and the user variations in the sensor data. In this section we discuss the most relevant.

4.1 Tackling Over-fitting

Over-fitting is a very general, but serious, problem in transportation mode recognition. Referring back to **Figure 9**, most teams across the three challenges suffered from the over-fitting problem, with the predicted performance much higher than the actual performance for the testing data. Three strategies have shown to be the most successful in tackling the over-fitting problem.

Cross-validation is an effective way to detect the over-fitting problem. This strategy proposes splitting the data into mutually exclusive K folds, and then training and evaluating the models on each fold, i.e., K times. The strategy has been investigated specifically in (Widhalm et al., 2018) and it was discovered that, for the training dataset with random-order segments, the standard K-folds partitioning scheme tends to introduce upward (optimistic) bias of the performance. Therefore, the authors proposed an improved version of the strategy, i.e., first to unshuffle the data and recover the temporal order of the segments (using the order file provided by the challenge organizer), and then applying K-fold cross-validation. This strategy yielded more accurate and realistic performance estimation.

Ensemble method is a effective approach to tackle the over-fitting problem, e.g. by using RF (Antar et al., 2018; Matsuyama et al., 2018; Nakamura et al., 2018; Siraj and et al., 2020; Brajesh and Ray, 2020), XGBoost (Janko et al., 2018; Lu et al., 2019; Kalabakov et al., 2020; Tseng et al., 2020) or ensembles of classifiers (Gjoreski et al., 2018b; Janko et al., 2019; Ahmed et al., 2019; Widhalm et al., 2019). In ML in general, it has been shown numerous times that ensembles with a large number of base predictors (above 50) are less prone to over-fitting. This was also proven with our challenge, in which on average the ensemble-based approaches achieved better results than the single classifier methods.

Dropout is another strategy to avoid over-fitting, especially for the approaches that use large Deep Learning architecture. This approach randomly deactivates (ignores) a predefined number of units in the neural network during training. This procedure is applied to each mini-batch of data, i.e., predefined units are randomly deactivated. This allows the model to train and update with a different "view" of the configured network. Dropout makes the training process noisy, forcing units within a layer to take on more or less responsibility for the inputs and making the model more robust. This has been also shown in the competition's submissions, where each Deep Learning architecture employed dropout in order to improve the performance (Srivastava et al., 2014).

4.2 Robust Representation

The unknown rotation and orientation of phone placement and human-phone engagement impose challenges to position-independent recognition. To tackle this challenge, several strategies were proposed to use orientation/position-independent representation of the sensor data.

Magnitude Representation

The magnitude of sensor data (accelerometer, gyroscope and magnetometer), which is the magnitude of the vector represented in 3D coordinate system, has been widely used (by most teams) for feature computation or classifier training. Some of the teams were going a step further, and remove the vector projections (x, y, and z) and keeping only the magnitudes. This way, on one side they are losing valuable information about the phone orientation, but on the other hand the representations are orientation independent and thus are having a more robust model.

Coordinate Transformation

Several submissions converted the sensor data from phone-centred coordinate system to human-centred coordination system, which can potentially increase the robustness to phone placement (Zhu et al., 2020; Zhao et al., 2020; Siraj

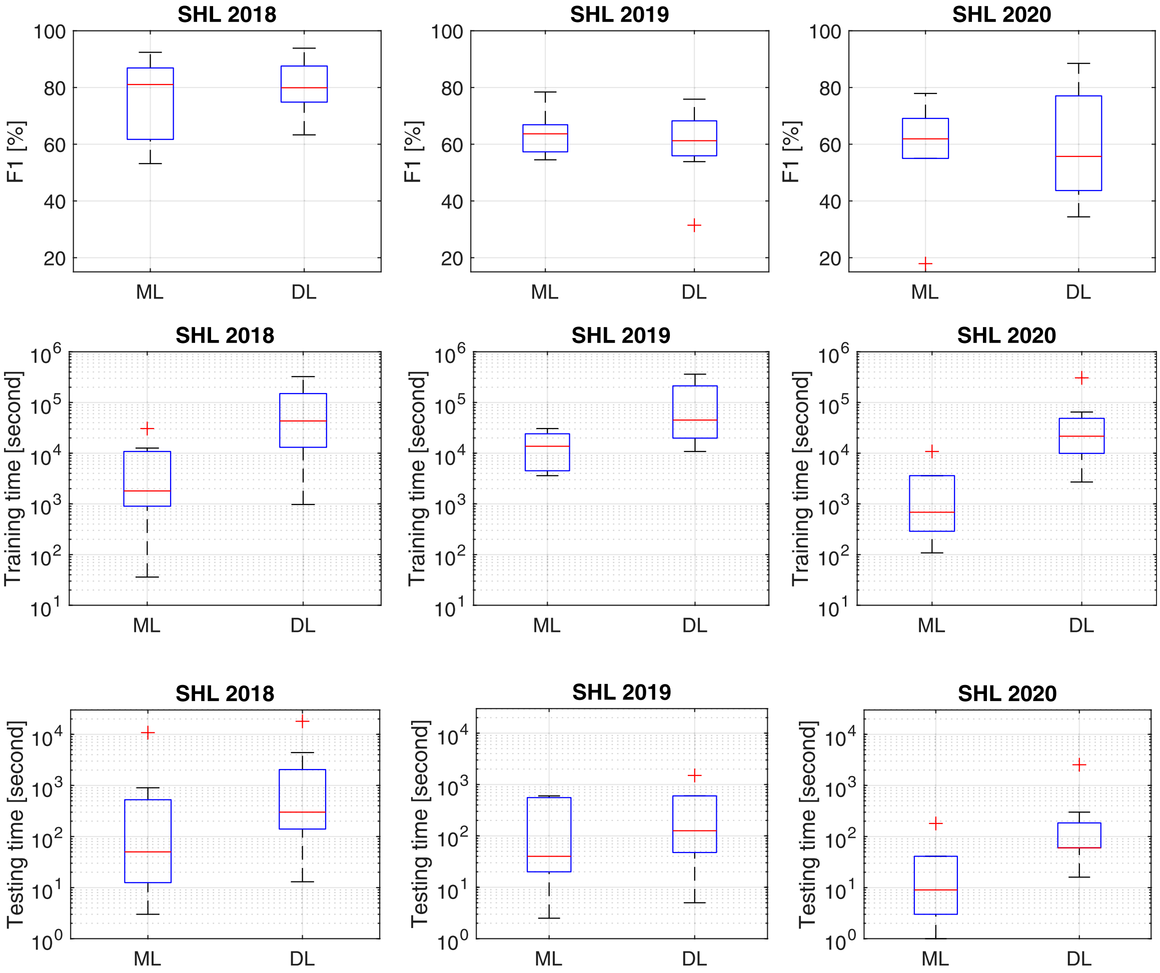


FIGURE 11 | Comparison between classical machine learning and deep learning approaches. **(A)** F1 score for the testing data. **(B)** Training time. **(C)** Testing time. The boxplot illustrates the minimum, first quartile, median, third quartile, and maximum values in each group of data, and the also the outlier values.

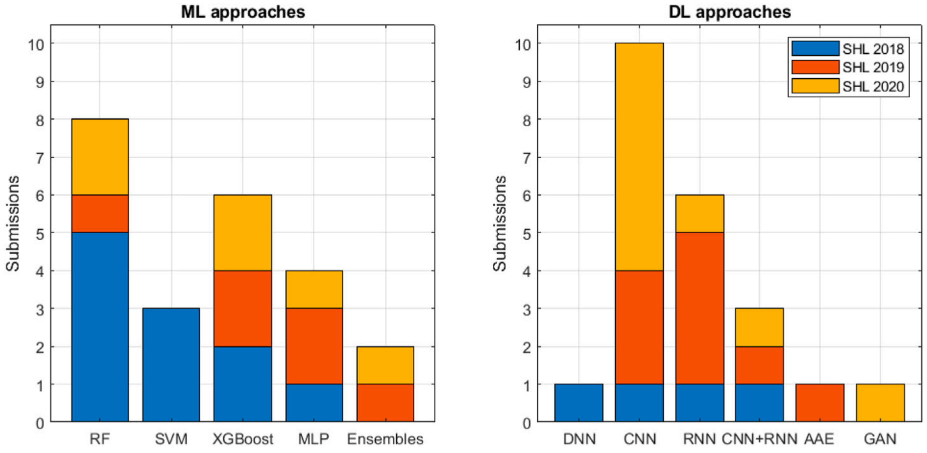


FIGURE 12 | Specific classifier employed by classical machine learning and deep learning approaches in SHL 2018–2020.

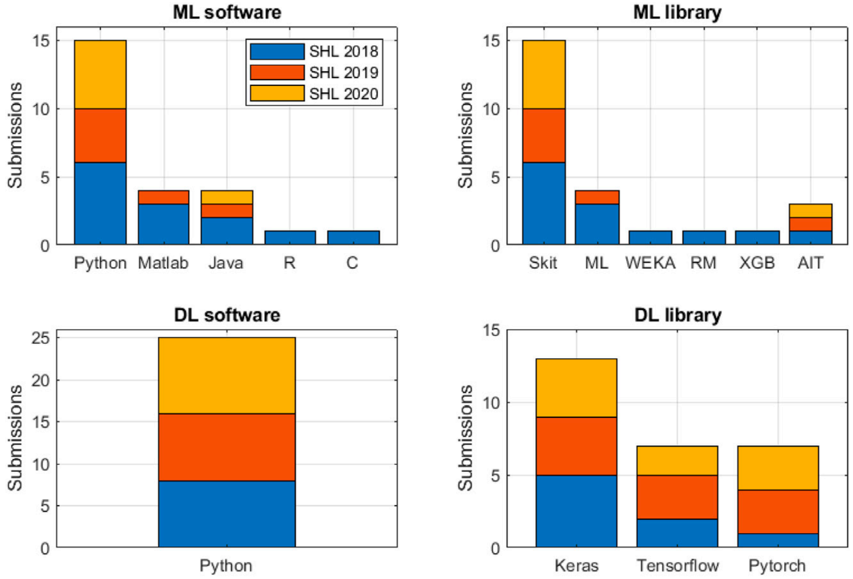


FIGURE 13 | Programming languages and libraries used by the submissions for classical machine learning and deep learning.

and et al., 2020; Tseng et al., 2020; Sekiguchi et al., 2020; Ahmed et al., 2019). The idea with this technique is to reduce the dependence on the orientation of the phone relative to the body. Once this transformation is performed, one can additionally filter the data and remove the orientation-specific features.

Random Rotation

This scheme proposed in (Choi and Lee, 2019) is to randomly change the orientation of the sensor data in the training set, aiming to increase the robustness of the trained model to new phone positioning. This could also be a strategy against over-fitting. The rationale behind this strategy is that the phone orientation is not the same for the same activities for different users, e.g., the phone in the pocket can have various orientations—some users put it with the screen towards the leg, others vice versa. This technique is especially useful if you are using not only the magnitude of the acceleration vector, but also the vector projection along the three axes.

4.3 Data Augmentation

The challenges 2019 and 2020 provided a small amount of validation data, e.g. the data collected at "Hand" phones (SHL 2019), and the data collected by User2 and User3 (SHL 2020). The validation data was a possibility to exploit data augmentation techniques.

Exploiting Target Domain Data

Several submissions proposed to enhance the training performance by exploiting the validation dataset (Janko et al., 2019; Widhalm et al., 2019; Choi and Lee, 2019; Ito et al., 2019). In SHL 2018 and 2019, the frames in the validation set are permuted randomly. It has been reported that the traditional cross-validation scheme that randomly splits the dataset, neglecting the temporal correlation between the neighbouring frames, may lead to an upward scoring bias (Widhalm et al., 2018). To solve this problem, one submission (Janko et al., 2019) proposed an order-recovering approach which that can roughly recover the temporal orders of the frames by looking at temporal dependencies across frames.

Transfer Learning

The idea with transfer learning is that a previously learned model is used and adapted to another, related task. This approach has proven successful in numerous applications of Deep Learning, where a Neural network trained on some dataset is reused and adapted to make classifications on another dataset. In our case, some researchers also tried to implement it so that they tackle the problem with cross-location of the smartphones, i.e., pocket, hand, backpack, etc. In particular, this approach trains the model at source locations and generalizes it to new target locations, exploiting the small amount of validation data (Janko et al., 2019; Ito et al., 2019).

Position-specific Modeling

In SHL 2020, the phone placement location in the testing set is unknown, but is disclosed to be one of the four locations in the training set. To exploit this fact, several submissions employed machine learning techniques to recognize the phone location first, and then developed a position-dependent model using the training and validation data. Interestingly, most of these submissions can estimate the phone location ("Hips") correctly (Kalabakov et al., 2020; Widhalm et al., 2020; Zhao et al., 2020; Yaguchi et al., 2020; Sekiguchi et al., 2020), whereas one submission obtained a wrong estimation of the phone location ("Hand") (Siraj and et al., 2020).

User-specific Modeling

In SHL 2020, the validation and the testing dataset are collected by the same users (User2 and User3). One submission (Kalabakov et al., 2020) developed a user-specific modeling approach that exploited this fact to improve the recognition performance. The method trains two user-dependent models (for User2 and User3, respectively) by applying transfer learning techniques to the validation dataset. During testing, the method classifies the data into two users and applies the user-specific models for recognition.

4.4 Post-processing

In SHL 2018, the length of the testing segment is 1 minute. Since for performance evaluation, the label is one label per sensor sample, not per window, participants explored a variety of windows (subframe) size, ranging from 0.1 s up to the whole 1-min segment, and then improved the recognition accuracy with post-processing approaches that exploit the temporal correlation between neighbouring subframes. Majority voting and the hidden Markov model (HMM) are two most popular post-processing techniques.

Most submissions that employed a 1-min decision window assume a single class activity within this frame. However, it was observed in the challenge data that a 1-min frame may contain more than one activity, i. e, the transition between two activities may occur. For instance, in the testing data of SHL 2018, 226 out of the 5,698 frames contain such a transition. In this case, either doing classification or performing post-processing within the 1-min frame may produce erroneous results, because the stationary assumption does not hold any more. One submission (Widhalm et al., 2018) managed to deal with this issue by employing a 3-s decision window followed by a hidden Markov model (HMM) for post-processing. This approach can better capture the transition within the long segment of 1 min. Indeed, HMMs are well suited to model the temporal dynamics of activities as most activities will transition from the "Walk" activity to another one and back to "Walk".

In SHL 2019 and 2020, the length of the testing segment is reduced to 5 s to encourage real-life applications with a shorter decision delay. The transition between activities rarely occurs in such a short segment. To prevent temporal smoothing, the testing frames are randomly shuffled, with the temporal order remain unknown to the participants. Interestingly, one submission (Janko et al., 2019) at SHL 2019 "cleverly" reconstructed the temporal sequence of the shuffled frames by exploiting the correlation between neighbouring frames, and that helped reach high performance. To prevent this trick (Janko et al., 2019), we used a larger jump size (10 s) when segmenting the testing data in SHL 2020. Despite this, one submission (Widhalm et al., 2020) managed to smooth the decision temporally by proposing a nearest-neighbour smoothing approach.

5 BASELINE IMPLEMENTATION

To better understand the challenges of the three tasks, we (the challenge organization committee) implemented two baseline systems that we applied to SHL 2018–2020. The baseline systems are based on the classical machine learning (ML) and deep learning (DL) pipelines, respectively, and were originally proposed in (Wang et al., 2018).

5.1 Classical Machine Learning Pipeline

Figure 14 depicts the classical machine learning pipeline for transportation mode recognition. We compute hand-crafted features from the sensor data and feed them as input to the classifier.

We use the data from three inertial sensors: accelerometer, gyroscope, magnetometer. Each sensor contains three channels of measurement along the X, Y, and Z axis of the device, respectively. To increase the robustness to unknown orientation and placement of the smartphone device, we compute the magnitude combining the three channels of each sensor, i.e.

$$s_{acc}(n) = \sqrt{s_{acc\_x}^2(n) + s_{acc\_y}^2(n) + s_{acc\_z}^2(n)} \tag{4}$$

$$s_{gyr}(n) = \sqrt{s_{gyr\_x}^2(n) + s_{gyr\_y}^2(n) + s_{gyr\_z}^2(n)} \tag{5}$$

$$s_{mag}(n) = \sqrt{s_{mag\_x}^2(n) + s_{mag\_y}^2(n) + s_{mag\_z}^2(n)} \tag{6}$$

where n denotes the sample index. We denote the magnitude sequence in the l frame of size T as

$$\boldsymbol{s}_{acc}(l) = [\boldsymbol{s}_{acc}^l(1), , \ldots, \boldsymbol{s}_{acc}^l(T)] \tag{7}$$

$$\boldsymbol{s}_{gyr}(l) = [\boldsymbol{s}_{gyr}^l(1), , \ldots, \boldsymbol{s}_{gyr}^l(T)] \tag{8}$$

$$\boldsymbol{s}_{mag}(l) = [\boldsymbol{s}_{mag}^l(1), , \ldots, \boldsymbol{s}_{mag}^l(T)] \tag{9}$$

In each data frame, we compute hand-crafted features on the three sensors, separately, and then cascade the features into a vector as

$$\boldsymbol{S}_H(l) = \begin{bmatrix} \boldsymbol{f}_{acc}(l) \\ \boldsymbol{f}_{gyr}(l) \\ \boldsymbol{f}_{mag}(l) \end{bmatrix}_{445\times 1}, \tag{10}$$

where $\boldsymbol{f}_{acc}(l)$, $\boldsymbol{f}_{gyr}(l)$, $\boldsymbol{f}_{mag}(l)$ denote a vector of hand-crafted features computed on the l-th frame of the magnitude sequence on the accelerometer, gyroscope, and magnetometer, respectively. As suggested in (Wang et al., 2019), we compute 147 accelerometer features, 150 gyroscope features and 148 magnetometer features, which were previously identified using a mutual information feature selection approach. The dimension of the feature vector $\boldsymbol{s}_H$ is 445. The detailed definition of these features is given in (Wang et al., 2019), which include various quantile ranges of the data value in the time domain, various subband energies in the frequency domain, and various statistical variables.

To improve the robustness of the classifiers, we employ a normalization pre-processing that reduce the dynamic range of the features by mapping them to a range of $(0, 1)$. Let's take the m-th feature f_m as an example. After computing f_m across all the frames in the training dataset, we obtain the percentile 95 (Q_m^{95}) and percentile 5 (Q_m^5) of f_m for the training data. The normalization is conducted in each frame l as

$$\bar{f}_m(l) \leftarrow \min\left(\max\left(\frac{f_m(l) - Q_m^5}{Q_m^{95} - Q_m^5}, 0\right), 1\right). \tag{11}$$

After normalization, the new feature vector is expressed as

$$\bar{\boldsymbol{S}}_H(l) = \begin{bmatrix} \bar{\boldsymbol{f}}_{acc}(l) \\ \bar{\boldsymbol{f}}_{gyr}(l) \\ \bar{\boldsymbol{f}}_{mag}(l) \end{bmatrix}_{445\times 1}. \tag{12}$$

Note that we can apply the same normalization procedure to the data frames in the testing set, given Q_m^{95} and Q_m^5 are already computed from the training data in advance.

For the recognition task, we use the well-known random forest classifier. The classifier is implemented with the Machine Learning Toolbox of Matlab. We set the number of trees as 20 and in each tree the parameter "minleafsize" as 1,000, and set other parameters as default in the toolbox.

5.2 Deep Learning Pipeline

Figure 15 illustrates the deep learning pipeline that infers the mode of transportation from sensor data with a convolutional neural network (CNN).

In each data frame l, we convert the time-domain sequence $\boldsymbol{s}_{acc}(l)$, $\boldsymbol{s}_{gyr}(l)$, and $\boldsymbol{S}_{mag}(l)$ to the frequency domain *via* Fourier transform. We retain only the data in first half frequency range, i.e [0, $fs/2$]), and then compute the magnitude of the sequence as $\boldsymbol{S}_{acc}(l)$, $\boldsymbol{S}_{gyr}(l)$, and $\boldsymbol{S}_{mag}(l)$. We cascade the new data from the three sensors into a vector as.

$$\boldsymbol{S}_F(l) = \begin{bmatrix} \boldsymbol{S}_{acc}(l) \\ \boldsymbol{S}_{gyr}(l) \\ \boldsymbol{S}_{mag}(l) \end{bmatrix}_{753\times 1}. \tag{13}$$

Suppose each frame contains 500 data samples, the size of the vector $\boldsymbol{S}_{acc}(l)$, $\boldsymbol{S}_{gyr}(l)$, $\boldsymbol{S}_{mag}(l)$ is 251 × 1 each, and the size of the new vector $\boldsymbol{S}_F$ (l) is thus 753 × 1.

Similarly to the ML pipeline, we employ a normalization preprocessing that reduce the dynamic range of the features by mapping them to a range of (0, 1). Let's take $\boldsymbol{S}_F^k(l)$, the k-the frequency bin in the l-the frame as an example. After computing $\boldsymbol{S}_F(l)$ across all the frames in the training dataset, we obtain the percentile 95 (Q_k^{95}) and percentile 5 (Q_k^5) of $\boldsymbol{S}_F^k$ for the training data. The normalization is conducted at each frequency bin l as

$$\bar{S}_F^k(l) \leftarrow \frac{S_F^k(l) - Q_k^5}{Q_k^{95} - Q_k^5}. \tag{14}$$

After normalization, we represent the new feature vector as

$$\bar{\boldsymbol{S}}_F(l) = \begin{bmatrix} \bar{\boldsymbol{S}}_{acc}(l) \\ \bar{\boldsymbol{S}}_{gyr}(l) \\ \bar{\boldsymbol{S}}_{mag}(l) \end{bmatrix}_{753\times 1}. \tag{15}$$

Note that we can apply the same normalization procedure to the data frames in the testing set, given Q_k^{95} and Q_k^5 are already computed from the training data in advance.

Figure 15 illustrates the architecture of the deep neural network, which sequentially consists of one input layer, multiple convolutional neural network (CNN) blocks, multiple fully-connected neural network (FCNN) blocks, and one decision layer. The input layer receives and stores the normalized feature vector from the sensor data. Each CNN block consists of one convolution layer (Conv), one batch-normalization layer (Norm), one nonlinear layer (ReLU). Each FCNN block consists of one fully-connected layer (FC), one batch-normalization layer (Norm) and one nonlinear layer (ReLU) and one dropout layer (Drop). The decision layer consists of a fully-connected layer (FC) and a nonlinear layer (SoftMax), which outputs the prediction result on the transportation mode. The batch normalization processes the sensor data in a mini-batch style, which normalizes the updates of the weights of the neural network per small subsets of training samples. It can accelerate the training speed and increase the robustness to random initializations. The dropout layer randomly forces the parameters of the neural network to zero with a predefined probability, which can prevent over-fitting effectively (Srivastava et al., 2014). The detailed configuration of the proposed architecture is given in **Table 5**.

The CNN classifier is implemented with the Deep Learning Toolbox of Matlab. We use the stochastic gradient descent with momentum (SGDM) as the optimizer, and set other parameters as default in the toolbox.

5.3 Baseline Results

We applied the two baseline systems to the recognition tasks of the three challenges. For each task, we retrain the recognition classifier using the training/validation data provided in the corresponding challenge. For ease of comparison, we consistently use a decision window of 5 s.

We test two schemes to use the validation data: AO (trAining data Only) and AV (trAining and Validation). In the first scheme AO, we only use the training data to train the classifier. In the second scheme, we use both training and validation data to train the classifier.

For SHL 2018, where the data is provided in the format of 1-min segments, we chopped data into 5-s frames, each containing 500 samples. For the frames in the training set, we used a sliding window of 5 s long and skip size 2.5 s. For the frames in the testing set, we used a sliding window of 5 s long and skip size 5 s. In this way, we generate training data of 375,130 frames and testing data of 68,376 frames. For each pipeline (ML and DL), the classifier is trained using the training data and then applied to the testing data.

For SHL 2019, where the data is provided in the format of 5-s segments, we don't need any pre-processing. In total, we have training data of 558,216 frames, validation data of 42,177 frames, and testing data of 55,811 frames. As mentioned earlier, for each pipeline (ML and DL), we trained two types of classifiers: AO and AV.

Similarly, for SHL 2020, where the data is also provided in the format of 5-s segments, we have training data of 784,288 frames, validation data of 115,156 frames and testing data of 57,573 frames. For each pipeline (ML and DL), we trained two types of classifiers: one using the training data only (AO) and one using both the training and validation data (AV).

For model training and testing, we used a desktop computer equipped with an Intel i7-4,770 4-core CPU @ 3.40 GHz with 32 GB memory, and a GeForce GTX 1080 Ti GPU with 3584

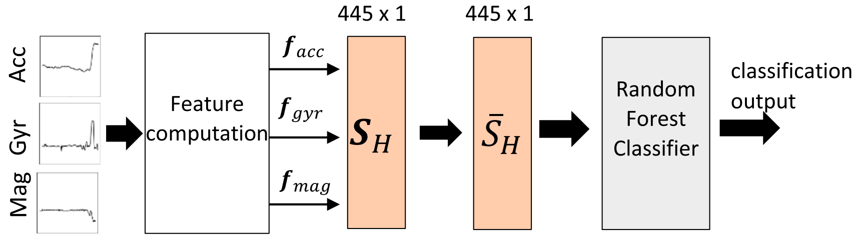


FIGURE 14 | Classical machine learning pipeline based on random forest.

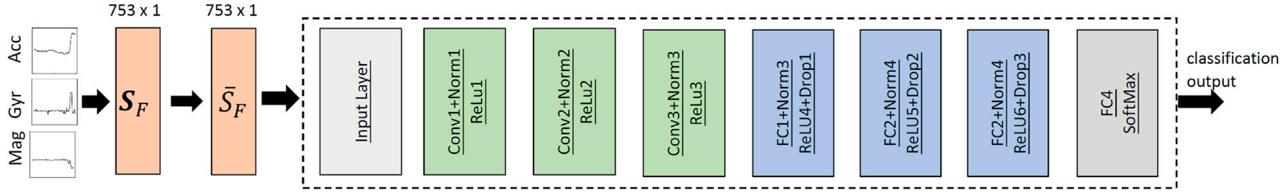


FIGURE 15 | Deep learning pipeline based on the convolutional neural network.

CUDA cores @ 1.58 GHz and 11 GB memory. The programs are coded in Matlab and are based on the Deep Learning and Machine Learning Toolbox.

Table 6 shows the detailed baseline results for SHL 2018–2020.[10] From the baseline results, the challenging level of the three tasks can be ranked as SHL 2018 < SHL 2020 < SHL 2019. Using the data collected at the same body position (Hips) by the same user, SHL 2018 is the easiest task considering only the temporal variation. SHL 2019 used the data collected from the same user but at different body positions. The testing position (Hand) is very different from the training position (Bag, Hips, Torso) as it involves many interactions between humans and smartphones, which are not captured by the other three positions. For this reason, SHL 2019 is the most challenging task for our ML and DL pipelines. SHL 2020 used data collected from the different users. While the testing position (Hips) is unknown, but is included in the training data, which provides the data collected at the four positions (Hips, Bag, Torso, Hand). The testing position (Hips) is easier than the one (Hand) in the previous year, as it involves less human interactions. For this reason, SHL 2020 is easier than SHL 2019, but more challenging than SHL 2018.

Similar to other submissions, DL outperforms ML, achieving 6–15 ppt (percentage point) higher F1 score in the three challenges. It seems that the high-level features extracted by DL from the data outperforms the hand-crafted features. On the other hand, the computational complexity of DL is much higher than ML. The training time of DL is about 50 times ML, while the testing time is about twice.

TABLE 5 | Configuration of the convolutional neural network.

Input layer	size: (753, 1)
Conv1/Conv2/Conv3	number: 100; size: (15,1); stride: (1,1); padding: (0,0)
FC1/FC2/FC3	nodes: 300
Dropl/Drop2/Drop3	50%
FC4	nodes: 8
Norm1-6	mini-batch: 500

Similar to other submissions, data augmentation plays an important role to increase recognition performance. For both SHL 2019 and 2020, the classifier trained with training and validation data (AV) outperforms the one trained with training data alone (AO). In particular, DL can improve the performance by 6.3 ppt in SHL 2019, and by 10.8 ppt in SHL 2020. In contrast, the improvement achieved by ML is much less, being 3.3 and 1.4 ppt for SHL 2019 and 2020, respectively. It seems that DL can better exploit augmented data to improve the recognition performance in challenging scenarios. In addition, both ML and DL baselines struggle to distinguish vehicle activities, i.e. train/subway and car/bus. This is the same as what we observed at the participants of the three challenges (**Figure 10**).

Figure 16 depicts the confusion matrices of the ML and DL baseline systems for SHL 2018–2020. The two baseline pipelines perform robustly across the three tasks. In comparison with **Figure 9**, the performance of the DL baseline is slightly higher than the second place in SHL 2018 (92.9 vs 92.4%)[11], the fifth place in SHL 2019 (66.6 vs 66.2%), and third place in SHL 2020

[10] Note that the computation time only considers the training and testing time of the classifier. The feature computation and data loading and writing time are not taken into account.

[11] The DL baseline achieves an F1 score of 92.9% after applying post-filtering within the 1-min segment (Wang et al., 2018).

TABLE 6 | Baseline performance for SHL 2018–2020. AO: the classifier is trained using the training data only; AV: the classifier is trained using both the training and validation data.

	Raw data frames (5 s)		ML			DL		
	Training	Testing	Training time (s)	Testing time (s)	F1 (%)	Training time (s)	Testing time (s)	F1 (%)
SHL 2018	375,130	68,376	94	3.4	76.6	4,604	7.5	82.5
SHL 2019	558,216 (AO)	55,811	160	2.3	56.9	7,066	6.1	60.3
	600,393 (AV)	55,811	171	2.9	60.2	8,810	6.7	66.6
SHL 2020	784,288 (AO)	57,573	239	2.7	61.9	11,936	6.3	68.0
	899,444 (AV)	57,573	304	3.3	63.3	13,110	6.8	78.8

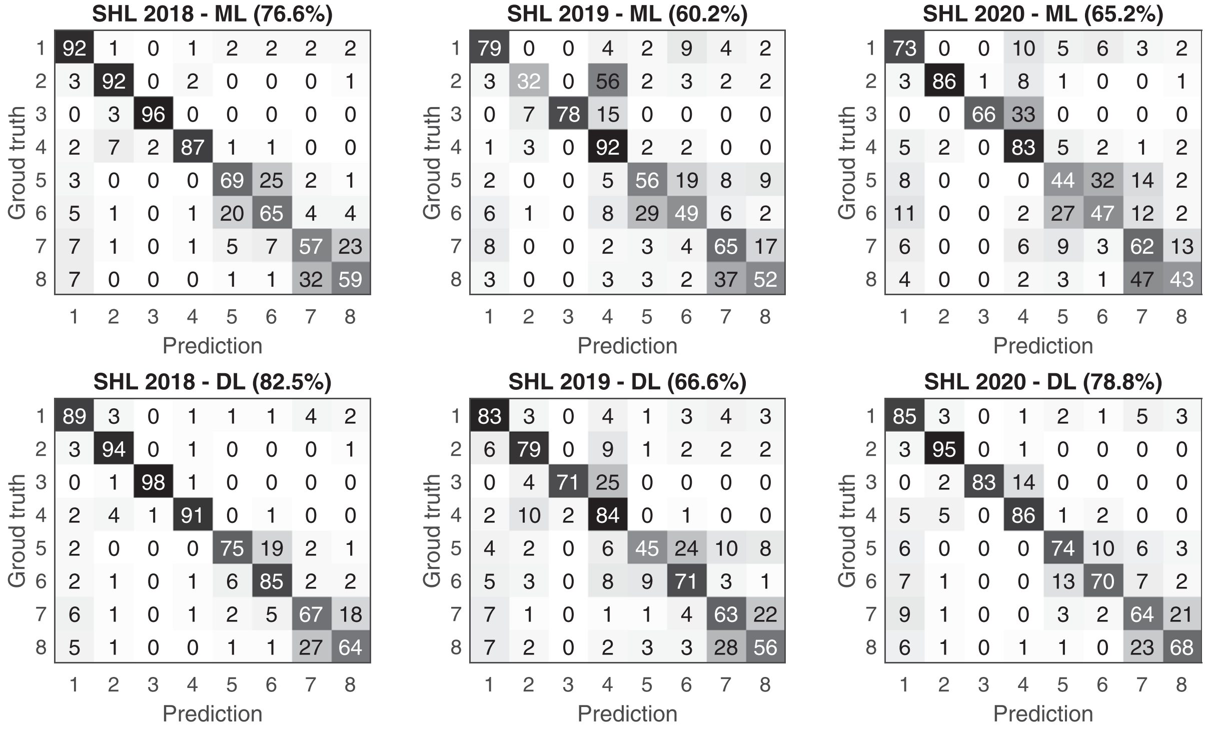


FIGURE 16 | Confusion matrices of the ML and DL baseline systems for SHL 2018–2020 with a 5-s decision window. The eight class activities are: 1—Still; 2—Walk; 3—Run; 4—Bike; 5—Car; 6—Bus; 7—Train; 8—Subway.

(78.8 vs 77.9%). In fact, the baseline systems were just retained without fine-tuning. We can use this pipeline (Wang et al., 2018) as a comparative reference in future challenge events.

6 REFLECTIONS ON THE CHALLENGE ORGANIZATION

We offer some comments on what might motivate people to participate to our challenge.

Initially we were wondering whether teams might participate solely for the financial motivation of winning one of the three prizes awarded each year (800, 400 and 200 GBP for the first to third prize), similarly to how non-academic teams compete on other data science challenges (e.g., on Kaggle). Such a motivation could have been detrimental to our challenge which aims to capture a snapshot of the state of the art, rather than just seeking solutions reaching the highest performance. To avoid this issue, we advertised clearly that the challenge requires the submission of a paper to the HASCA workshop organised at Ubicomp, which we peer reviewed for quality. This may have weeded out teams not interested in the scientific publication exercise, but we also recognize that the prizes awarded are small in contrast to other data science challenges (e.g., on Kaggle). In the end, the vast majority of participants came unsuprisingly from academia, which may further have been enhanced by our advertising the challenge only on academic mailing lists (e.g., CHI, Ubicomp). Overall, the financial incentive is certainly beneficial but does not appear to be the main driver for participation. Notably, some of the winning teams were rather large, which further reduces the individual financial incentive if winning teams shared the prize among members. We noticed teams participating for 3 years in a row despite not winning any of the prizes. Informal discussions with participants revealed that the "challenge" aspect seemed a key motivation.

We understood that for the supervisors of some research groups, encouraging PhD students to participate in a well-delimited challenge is a good way to induce new PhD students in research skills and collaboration within the team. We gave a certificate of participation to all teams. This was requested explicitly by participants the first year of the challenge. We received additional feedback in the following years that such a certificate was valuable to participants. We speculate this may be useful to show engagement with an international event, which may be valued by the university or by future employers. Another motivation is clearly the opportunity to have the work of the participants published to the HASCA workshop organised alongside Ubicomp. As a lesson learned, awarding a certificate and a venue for publication seems to be a good recipe to follow for other challenge organisers.

We widely advertised through multiple mailing lists, with a particular effort to widening participation (e.g., Women in Machine Learning). Each year saw about 30 expressions of interest, out of which 50% led to actual submission of a competition entry. Asking for expression of interest at the start of the challenge period is helpful to organise the workload once the competition entries are received and may be another advice to other challenge organisers.

Finally, we initially debated whether to organise the challenge on an established platform. Eventually, we decided to roll-out our own challenge process with a manual submission process (*via* email) but with a well define file submission format, which allows an automated evaluation at the end. This worked well for our challenge, at the expense of not having niceties such as real-time scoreboards that other platforms might have. However, such scoreboards may also lead to participants over-fitting their methods, and we decided for a "blind" submission to prevent this instead. Participants obviously could split our part of the training data for their own validation approach and in some cases we indicate some parts of already-public data could be used as validation. The take home message is to communicate a clear file format for the submission. We never experienced submissions using an incorrect format. We did provide an example of such result file format as a reference, and this may have helped ensuring the correct format.

7 CONCLUSION

We surveyed the state of the art in transportation and locomotion recognition from smartphone motions sensors, as captured by the achievements obtained during the three SHL recognition challenges 2018–2020, which aimed to recognize user's transportation and locomotion mode from smartphone motion sensors.

In total, 21, 14 and 15 submissions were received for the three challenges, respectively. SHL 2018 achieves the highest F1 score 93.9% and an average F1 score of 86.9% (top 10 teams) for time-independent evaluation and 1-min decision window (Wang et al., 2018). SHL 2019 achieves the highest F1 score of 78.4% and an average F1 score of 66.5% (top 10 teams) for position independent evaluation and 5-s decision window (Wang et al., 2019). SHL 2020 achieves the highest F1 score of 88.9% and an average F1 score of 69.5% (top 10 teams) for user-independent evaluation and 5-s decision window (Wang et al., 2020). Because the approaches are implemented by different research groups with varying expertise, the conclusions drawn will be confined to the submissions of the SHL challenges. We additionally provide a general baseline solution that can be applied to the three challenges. The baseline results confirm the challenging level of the three challenges being ranked as: SHL 2018 being the easiest, followed by SHL 2020, and SHL 2019 being the hardest.

The submissions can be broadly divided into ML and DL pipelines. We observe that DL is becoming more popular with the evolution of the challenges, starting with 42% of the submissions in 2018 to 60% of the submissions in 2020. Overall, DL approaches outperformed the ML approaches in the three challenges, but are more computationally complex. The downside is that in the worst case DL may perform much less well than ML approaches, which we attribute to the complexity of effectively deploying deep learning approaches and optimizing architecture and hyper parameters effectively within a time-constrained challenged. Various schemes have been employed by the participant teams to tackle the challenge of the variation of time, user and position, including robust representation, data augmentation, tackling over-fitting, and post-processing. These methods provide a good insight for developing novel algorithms for activity recognition in real life.

The challenges showcased an increased sophistication in state of the art methods. Transportation and locomotion mode recognition based only on motion sensors was able to distinguish most modes with relative ease, and to a more limited extent was able to distinguish subtly distinct modes, for example between train and subway (rail transport) or between bus and car (road transport). However there are still evident disadvantage when using only one sensor modality. Future work would be to exploit the multi-modal sensor data [e.g. GPS (Wang et al., 2019), image (Richoz et al., 2019), sound (Wang and Roggen, 2019), and multimodal fusion (Richoz et al., 2020)] to improve the robustness to position and user variation[12].

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: http://www.shl-dataset.org/.

AUTHOR CONTRIBUTIONS

All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

FUNDING

The dataset used in this work was supported by HUAWEI Technologies within the project "Activity Sensing Technologies

[12] At the time of writing this paper, the SHL 2021 is going on, which aims at transportation mmode recognition using GPS and radio sensors (Wang et al., 2021).

for Mobile Users". The funder was not involved in the study design, collection, analysis, interpretation of data, the writing of this article or the decision to submit it for publication. This work received support from the EU H2020-ICT-2019-3 project "HumanE AI Net" (project number 952026). All authors declare no other competing interests.

REFERENCES

Ahmed, M., Antar, A. D., and Hossain, T. (2019). "POIDEN: Position and Orientation Independent Deep Ensemble Network for the Classification of Locomotion and Transportation Modes," in Proc. 2019 ACM International Joint Conference and 2019 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, London, United Kingdom, September 2019, 674–679. doi:10.1145/3341162.3345570

Akbari, A., Wu, J., Grimsley, R., and Jafari, R. (2018). "Hierarchical Signal Segmentation and Classification for Accurate Activity Recognition," in Proc. 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, Singapore, September 2018, 1596–1605. doi:10.1145/3267305.3267528

Alwan, A., Frey, V., and Lan, G. L. (2019). "Orange Labs Contribution to the Sussex-Huawei Locomotion-Transportation Recognition challenge," in Proc. 2019 ACM International Joint Conference and 2019 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, London, United Kingdom, September 2019, 680–684. doi:10.1145/3341162.3344860

Anagnostopoulou, E., Urbancic, J., Bothos, E., Magoutas, B., Bradesko, L., Schrammel, J., et al. (2018). From Mobility Patterns to Behavioural Change: Leveraging Travel Behaviour and Personality Profiles to Nudge for Sustainable Transportation. *J. Intell. Inf. Syst.* 2018, 1–22. doi:10.1007/s10844-018-0528-1

Antar, A. D., Ahmed, M., Ishrak, M. S., and Ahad, M. A. R. (2018). "A Comparative Approach to Classification of Locomotion and Transportation Modes Using Smartphone Sensor Data," in Proc. 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, Singapore, September 2018, 1497–1502. doi:10.1145/3267305.3267516

Balabka, D. (2019). "Semi-supervised Learning for Human Activity Recognition Using Adversarial Autoencoders," in Proc. 2019 ACM International Joint Conference and 2019 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, London, United Kingdom, September 2019, 685–688. doi:10.1145/3341162.3344854

Biancat, C., and Brighenti, A. (2014). Review of Transportation Mode Detection Techniques. *EAI Endorsed Transportation Ambient Syst.* 1 (4), 1–10. doi:10.4108/amsys.1.4.e7

Brajesh, S., and Ray, I. (2020). "Ensemble Approach for Sensor-Based Human Activity Recognition," in Proc. 2020 ACM International Joint Conference and 2020 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, September 2020, 296–300. doi:10.1145/3410530.3414352

Brazil, W., and Caulfield, B. (2013). Does green Make a Difference: The Potential Role of Smartphone Technology in Transport Behaviour. *Transportation Res. C: Emerging Tech.* 37, 93–101. doi:10.1016/j.trc.2013.09.016

Castignani, G., Derrmann, T., Frank, R., and Engel, T. (2015). Driver Behavior Profiling Using Smartphones: A Low-Cost Platform for Driver Monitoring. *IEEE Intell. Transport. Syst. Mag.* 7 (1), 91–102. doi:10.1109/mits.2014.2328673

Choi, J., and Lee, J. (2019). "EmbraceNet for Activity: A Deep Multimodal Fusion Architecture for Activity Recognition," in Proc. 2019 ACM International Joint Conference and 2019 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, London, United Kingdom, September 2019, 693–698. doi:10.1145/3341162.3344871

Cottrill, C. D., Pereira, F. C., Zhao, F., Dias, I. F., Lim, H. B., Ben-Akiva, M. E., et al. (2013). Future Mobility Survey. *Transportation Res. Rec.* 2354 (1), 59–67. doi:10.3141/2354-07

Dabiri, S., and Heaslip, K. (2018). Inferring Transportation Modes from GPS Trajectories Using a Convolutional Neural Network. *Transportation Res. Part C: Emerging Tech.* 86, 360–371. doi:10.1016/j.trc.2017.11.021

Engelbrecht, J., Booysen, M. J., Rooyen, G. J., and Bruwer, F. J. (2015). Survey of Smartphone-based Sensing in Vehicles for Intelligent Transportation System Applications. *IET Intell. Transport Syst.* 9 (10), 924–935. doi:10.1049/iet-its.2014.0248

Friedich, B., Cauchi, B., Hein, A., and Fudickar, S. (2019). "Transportation Mode Classification from Smartphone Sensors via a Long-Short-Term Memory Network," in Proc. 2019 ACM International Joint Conference and 2019 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, London, United Kingdom, September 2019, 709–713. doi:10.1145/3341162.3344855

Froehlich, J., Dillahunt, T., Klasnja, P., Mankoff, J., Consolvo, S., Harrison, B., and Landay, J. A. (2009). "Ubigreen: Investigating a mobile Tool for Tracking and Supporting green Transportation Habits," in Proc. SIGCHI Conference on Human Factors Computing Systems, Boston, United States, April 2009, 1043–1052. doi:10.1145/1518701.1518861

Gjoreski, H., Ciliberto, M., Wang, L., Ordonez Morales, F. J., Mekki, S., Valentin, S., et al. (2018a). The University of Sussex-Huawei Locomotion and Transportation Dataset for Multimodal Analytics with mobile Devices. *IEEE Access* 6, 42592–42604. doi:10.1109/access.2018.2858933

Gjoreski, M., Janko, V., Rescic, N., Mlakar, M., Lustrek, M., Bizjak, J., Slapnicar, G., Marinko, M., Drobnic, V., and Gams, M. (2018b). "Applying Multiple Knowledge to Sussex-Huawei Locomotion challenge," in Proc. 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, Singapore, September 2018, 1488–1496. doi:10.1145/3267305.3267515

Gong, L., Morikawa, T., Yamamoto, T., and Sato, H. (2014). Deriving Personal Trip Data from GPS Data: A Literature Review on the Existing Methodologies. *Proced. - Soc. Behav. Sci.* 138, 557–565. doi:10.1016/j.sbspro.2014.07.239

Gunthermann, L., Simpson, I., and Roggen, D. (2020). "Smartphone Location Identification and Transport Mode Recognition Using an Ensemble of Generative Adversarial Networks," in Proc. 2020 ACM International Joint Conference and 2020 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, September 2020, 311–316. doi:10.1145/3410530.3414353

Guo, M., Liang, S., Zhao, L., and Wang, P. (2020). Transportation Mode Recognition with Deep forest Based on GPS Data. *IEEE Access* 8, 150891–150901. doi:10.1109/access.2020.3015242

Ito, C., Cao, X., Shuzo, M., and Maeda, E. (2018). "Application of CNN for Human Activity Recognition with FFT Spectrogram of Acceleration and Gyro Sensors," in Proc. 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, Singapore, September 2018, 1503–1510. doi:10.1145/3267305.3267517

Ito, C., Shuzo, M., and Maeda, E. (2019). "CNN for Human Activity Recognition on Small Datasets of Acceleration and Gyro Sensors Using Transfer Learning," in Proc. 2019 ACM International Joint Conference and 2019 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, London, United Kingdom, September 2019, 724–729. doi:10.1145/3341162.3344868

Janko, V., Gjoreski, M., De Masi, C. M., Rescic, N., Lustrek, M., and Gams, M. (2019). "Cross-location Transfer Learning for the Sussex-Huawei Locomotion Recognition challenge," in Proc. 2019 ACM International Joint Conference and 2019 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, London, United Kingdom, September 2019, 730–735. doi:10.1145/3341162.3344856

Janko, V., Rescic, N., Mlakar, M., Drobnic, V., Gams, M., Slapnicar, G., Gjoreski, M., Bizjak, J., Marinko, M., and Lustrek, M. (2018). "A New Frontier for Activity Recognition - the Sussex-Huawei Locomotion challenge," in Proc. 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, Singapore, September 2018, 1511–1520. doi:10.1145/3267305.3267518

Johnson, D. A., and Trivedi, M. M. (2011). "Driving Style Recognition Using a Smartphone as a Sensor Platform," in Proc. IEEE Conference on

Intelligent Transportation Systems, Washington, United States, October 2011, 1609–1615. doi:10.1109/itsc.2011.6083078

Kalabakov, S., Stankoski, S., Rescic, N., Kiprijanovska, I., Andova, A., Picard, C., Janko, V., Gjoreski, M., and Lustrek, M. (2020). "Tackling the SHL Challenge 2020 with Person-specific Classifiers and Semi-supervised Learning," in Proc. 2020 ACM International Joint Conference and 2020 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, September 2020, 323–328. doi:10.1145/3410530.3414848

Lane, N., Miluzzo, E., Lu, H., Peebles, D., Choudhury, T., and Campbell, A. (2010). A Survey of mobile Phone Sensing. *IEEE Commun. Mag.* 48 (9), 140–150. doi:10.1109/mcom.2010.5560598

Lu, H., Pinaroc, M., Lv, M., Sun, S., Han, H., and Shah, R. C. (2019). "Locomotion Recognition Using XGBoost and Neural Network Ensemble," in Proc. 2019 ACM International Joint Conference and 2019 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, London, United Kingdom, September 2019, 757–760. doi:10.1145/3341162.3344870

Matsuyama, H., Urano, K., Hiroi, K., Kaji, K., and Kawaguchi, N. (2018). "Short Segment Random forest with post Processing Using Label Constraint for SHL challenge," in Proc. 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, Singapore, September 2018, 1636–1642. doi:10.1145/3267305.3267532

Mukhopadhyay, S. C. (2015). Wearable Sensors for Human Activity Monitoring: A Review. *IEEE Sensors J.* 15 (3), 1321–1330. doi:10.1109/jsen.2014.2370945

Nakamura, Y., Umetsu, Y., Talusan, J. P., Yasumoto, K., Sasaki, W., Takata, M., and Arakawa, Y. (2018). "Multi-stage Activity Inference for Locomotion and Transportation Analytics of mobile Users," in Proc. 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, Singapore, September 2018, 1579–1588. doi:10.1145/3267305.3267526

Naseeb, C., and Saeedi, B. A. (2020). "Activity Recognition for Locomotion and Transportation Dataset Using Deep Learning," in Proc. 2020 ACM International Joint Conference and 2020 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, September 2020, 329–334. doi:10.1145/3410530.3414348

Richoz, S., Ciliberto, M., Wang, L., Birch, P., Gjoreski, H., Perez-Uribe, A., and Roggen, D. (2019). "Human and Machine Recognition of Transportation Modes from Body-Worn Camera Images," in Proc. Joint 8th International Conference on Informatics, Electronics & Vision and 3rd International Conference on Imaging, Vision & Pattern Recognition, Spokane, United States, June 2019, 67–72. doi:10.1109/iciev.2019.8858537

Richoz, S., Wang, L., Birch, P., and Roggen, D. (2020). Transportation Mode Recognition Fusing Wearable Motion, Sound and Vision Sensors. *IEEE Sensors J.* 20 (16), 9314–9328. doi:10.1109/jsen.2020.2987306

Sekiguchi, R., Abe, K., Yokoyama, T., Kumano, M., and Kawakatsu, M. (2020). "Ensemble Learning for Human Activity Recognition," in Proc. 2020 ACM International Joint Conference and 2020 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, September 2020, 335–339. doi:10.1145/3410530.3414346

Siraj, M. S., et al. (2020). "UPIC: User and Position Independent Classical Approach for Locomotion and Transportation Modes Recognition," in Proc. 2020 ACM International Joint Conference and 2020 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, September 2020. doi:10.1145/3410530.3414343

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a Simple Way to Prevent Neural Networks from Overfitting. *J. Machine Learn. Res.* 15, 1929–1958.

Tseng, Y., Lin, H., Lin, Y., and Chen, J. (2020). "Hierarchical Classification Using ML/DL for Sussex-Huawei Locomotion-Transportation (SHL) Recognition Challenge," in Proc. 2020 ACM International Joint Conference and 2020 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, September 2020, 346–350. doi:10.1145/3410530.3414347

Vaizman, Y., Ellis, K., and Lanckriet, G. (2017). Recognizing Detailed Human Context in the Wild from Smartphones and Smartwatches. *IEEE Pervasive Comput.* 16 (4), 62–74. doi:10.1109/mprv.2017.3971131

Wang, L., Ciliberto, M., Gjoreski, H., Lago, P., Murao, K., Okita, T., and Roggen, D. (2021). "Locomotion and Transportation Mode Recognition from GPS and Radio Signals: Summary of SHL Challenge 2021," in Adjunct Proc. 2021 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proc. 2021 ACM International Symposium on Wearable Computers, September 2021. doi:10.1145/3460418.3479373

Wang, L., Gjoreski, H., Ciliberto, M., Lago, P., Murao, K., Okita, T., and Roggen, D. (2019). "Summary of the Sussex-Huawei Locomotion-Transportation Recognition challenge 2019," in Adjunct Proc. 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proc. 2019 ACM International Symposium on Wearable Computers, London, United Kingdom, September 2019, 849–856. doi:10.1145/3341162.3344872

Wang, L., Gjoreski, H., Ciliberto, M., Lago, P., Murao, K., Okita, T., and Roggen, D. (2020). "Summary of the Sussex-Huawei Locomotion-Transportation Recognition challenge 2020," in Adjunct Proc. 2020 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proc. 2020 ACM International Symposium on Wearable Computers, September 2020, 351–358. doi:10.1145/3410530.3414341

Wang, L., Gjoreski, H., Ciliberto, M., Mekki, S., Valentin, S., and Roggen, D. (2018). "Benchmarking the SHL Recognition challenge with Classical and Deep-Learning Pipelines," in Proc. 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, Singapore, September 2018, 1626–1635. doi:10.1145/3267305.3267531

Wang, L., Gjoreski, H., Ciliberto, M., Mekki, S., Valentin, S., and Roggen, D. (2019). Enabling Reproducible Research in Sensor-Based Transportation Mode Recognition with the Sussex-Huawei Dataset. *IEEE Access* 7, 10870–10891. doi:10.1109/access.2019.2890793

Wang, L., Gjoreski, H., Murao, K., Okita, T., and Roggen, D. (2018). "Summary of the Sussex-Huawei Locomotion-Transportation Recognition challenge," in Proc. 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, Singapore, September 2018, 1521–1530. doi:10.1145/3267305.3267519

Wang, L., and Roggen, D. (2019). "Sound-based Transportation Mode Recognition with Smartphones," in Proc. IEEE International Conference on Acoustics, Speech and Signal Processing, Brighton, United Kingdom, May 2019, 930–934. doi:10.1109/icassp.2019.8682917

Widhalm, P., Leodolter, M., and Brandle, N. (2019). "Ensemble-based Domain Adaptation for Transport Mode Recognition with mobile Sensors," in Proc. 2019 ACM International Joint Conference and 2019 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, London, United Kingdom, September 2019, 857–861. doi:10.1145/3341162.3344857

Widhalm, P., Leodolter, M., and Brandle, N. (2018). "Top in the Lab, Flop in the Field? Evaluation of a Sensor-Based Travel Activity Classifier with the SHL Dataset," in Proc. 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, Singapore, September 2018, 1479–1487. doi:10.1145/3267305.3267514

Widhalm, P., Merz, P., Coconu, L., and Brandle, N. (2020). "Tackling the SHL Recognition challenge with Phone Position Detection and Nearest Neighbour Smoothing," in Proc. 2020 ACM International Joint Conference and 2020 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, September 2020, 359–363. doi:10.1145/3410530.3414344

Wu, J., Akbari, A., Grimsley, R., and Jafari, R. (2018). "A Decision Level Fusion and Signal Analysis Technique for Activity Segmentation and Recognition on Smartphones," in Proc. 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, Singapore, September 2018, 1571–1578. doi:10.1145/3267305.3267525

Xia, H., Qiao, Y., Jian, J., and Chang, Y. (2014). Using Smart Phone Sensors to Detect Transportation Modes. *Sensors* 14 (11), 20843–20865. doi:10.3390/s141120843

Yaguchi, K., Ikarigawa, K., Kawasaki, R., Miyazaki, W., Morikawa, Y., Ito, C., Shuzo, M., and Maeda, E. (2020). "Human Activity Recognition Using Multi-Input CNN Model with FFT Spectrograms," in Proc. 2020 ACM International

Joint Conference and 2020 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, September 2020, 364–367. doi:10.1145/3410530.3414342

Yu, M.-C., Yu, T., Wang, S.-C., Lin, C.-J., and Chang, E. Y. (2014). Big Data Small Footprint. *Proc. VLDB Endow.* 7, 1429–1440. doi:10.14778/2733004.2733015

Yuki, Y., Nozaki, J., Hiroi, K., Kaji, K., and Kawaguchi, N. (2018). "Activity Recognition Using Dual-ConvLSTM Extracting Local and Global Features for SHL Challenge," in Proc. 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, Singapore, September 2018, 1643–1651. doi:10.1145/3267305.3267533

Zhao, B., Li, S., and Gao, Y. (2020). "IndRNN Based Long-Term Temporal Recognition in the Spatial and Frequency Domain," in Proc. 2020 ACM International Joint Conference and 2020 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, September 2020, 368–372. doi:10.1145/3410530.3414355

Zheng, L., Li, S., Zhu, C., and Gao, Y. (2019). "Application of IndRNN for Human Acivity Recognition - the Sussex-Huawei Locomotion," in Proc. 2019 ACM International Joint Conference and 2019 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, London, United Kingdom, September 2019, 869–872. doi:10.1145/3341162.3344851

Zhu, Y., Luo, H., Chen, R., Zhao, F., and Su, L. (2020). "DenseNetX and GRU for the Sussex-Huawei Locomotion-Transportation Recognition challenge," in Proc. 2020 ACM International Joint Conference and 2020 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, September 2020, 373–377. doi:10.1145/3410530.3414349

Zhu, Y., Zhao, F., and Chen, R. (2019). "Applying 1D Sensor denseNet to Sussex-Huawei Locomotion-Transportation Recognition challenge," in Proc. 2019 ACM International Joint Conference and 2019 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, London, United Kingdom, September 2019, 873–877. doi:10.1145/3341162.3345571

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.