HAL
open science

A Framework for the Identification of Human Vertical Displacement Activity Based on Multi-Sensor Data

Ajaykumar Manivannan, Elias Willemse, Balamurali T., Wei Chien Benny Chin, Yuren Zhou, Bige Tuncer, Alain Barrat, Roland Bouffanais

To cite this version:

Ajaykumar Manivannan, Elias Willemse, Balamurali T., Wei Chien Benny Chin, Yuren Zhou, et al.. A Framework for the Identification of Human Vertical Displacement Activity Based on Multi-Sensor Data. IEEE Sensors Journal, In press, 22 (8), pp.8011. 10.1109/JSEN.2022.3157806 . hal-03604682

HAL Id: hal-03604682

https://hal.science/hal-03604682v1

Submitted on 10 Mar 2022

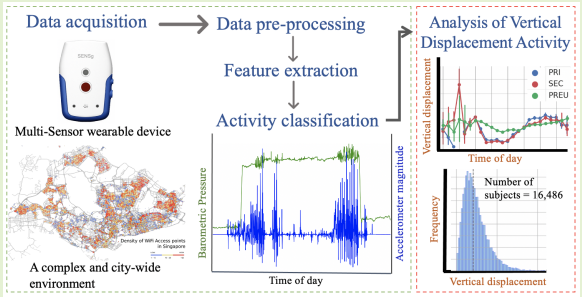
HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Framework for the Identification of Human Vertical Displacement Activity Based on Multi-Sensor Data

Ajaykumar Manivannan, Elias J. Willemse, Balamurali B. T., Wei Chien Benny Chin, Yuren Zhou, Bige Tunçer, Alain Barrat and Roland Bouffanais, *Member, IEEE*

***Abstract*— With new cities increasingly expanding vertically, there is a pressing need to shed light on human vertical mobility, which can readily be achieved with existing sensor technology. To date, the methodology to track and identify vertical movement from large-scale unstructured data sets is lacking. Here, we design and develop such a framework to accurately and systematically identify the sparse human vertical displacement activity that is typically buried into the predominantly horizontal mobility. Our framework uses sensor data from barometer, accelerometer and Wi-Fi scanner coupled with an extraction step involving a combination of feature engineering and data segmentation. This methodology is subsequently integrated into a machine-learning-based classifier to automatically distinguish vertical displacement activity from its horizontal counterpart. We confirm the high accuracy of this approach by a thorough validation and testing showing a 98% overall accuracy and a 92% F1-score in classifying vertical displacement activity.We illustrate the potential of the developed framework by applying it to an unstructured large-scale data set associated with over 16,000 participants going about their daily activity in the city-state of Singapore. This gives us access to all the vertical movements of this large population, and we investigate the statistical distribution of vertical activity, both in terms of number of events and size of vertical jumps, and their temporal heterogeneity across the day. The approach developed here could be used in massive human experiments to uncover the hidden patterns of human vertical mobility. This new knowledge would have significant ramifications for the architectural design of vertical cities.**



***Index Terms*— Wearable sensors, multi-sensor identification, human activity recognition, vertical displacement activity.**

I. Introduction

THE urbanization of our planet is rapidly increasing with 55% of the world population now living in cities [1]. In 2030, this number is projected to increase to 60% [1]. In the face of this unabated urbanization trend, cities are struggling to accommodate the population influx through urban sprawl alone, primarily because of land scarcity and the induced strain on transportation networks. An alternative to urban sprawl—currently predominant in the rapidly urbanizing Asia—consists in increasing the density of the built environment, inevitably leading to cities with vertically dominated landscapes and singular skylines.

Such vertical cities exhibit a very distinct urban landscape, manifesting a sprawl of an upward nature with a very high density of high-rise buildings—not necessarily limited to skyscrapers. This vertical growth of cities is reflected in the increasing market demand around the world for vertical transportation systems like elevators and escalators, with approximately $100,000$ units installed in 2019, and a forecast for $250,000$ units to be commissioned in 2024 alone [2]. The Asia-Pacific region, where most of the fastest growing cities in the world are present [1], is said to have the highest growth in demand (85%) for vertical transportation systems [2].

Today's urban planning has greatly benefited from extensive studies of human mobility over the last half century. Over the last two decades, this area of research has experienced significant growth due to the convergence of several technological factors: (1) the development of new sensors enabling more accurate tracking of human mobility, (2) the very rapid and massive adoption of mobile phones globally, and (3) the so-called "Big Data" effect. In addition, complexity scientists

The NSE 2016 was supported by the Singapore National Research Foundation (NRF), Ministry of Education (MOE),and the SUTD-MIT International Design Centre.

Balamurali B.T, Wei Chien Benny Chin, Yuren Zhou, and Bige Tunçer are with the Singapore University of Technology and Design (SUTD), 8 Somapah Rd, Singapore 487372.

Elias J. Willemse is with Waste Labs Pte. Ltd. 36 Carpenter Street, Singapore, 059915

Alain Barrat is with CNRS, CPT, Aix Marseille Univ, Université de Toulon, 13009 Marseille, France and Tokyo Tech World Research Hub Initiative (WRHI), Tokyo Institute of Technology, Yokohama 226-8503, Japan.

Ajaykumar Manivannan and Roland Bouffanais are with the Department of Mechanical Engineering, University of Ottawa, 161 Louis Pasteur, Ottawa, ON K1N 6N5, Canada (e-mail: roland.bouffanais@uottawa.ca).

have developed several new frameworks to analyze and identify specific patterns of mobility. Those patterns hidden in troves of high-resolution mobility data have been uncovered thanks to large-scale experiments or from massive commercial databases—e.g., call detailed record (CDR) [3]. However, it is important to note that these human mobility studies are limited to horizontal movements, i.e. based on a two-dimensional representation of the urban landscape. In these studies, roads, railway networks, and pedestrian pathways are modeled on a planar surface.

As already mentioned, the rapid change in the topology of cities in the developing world, especially in Asia, is prominently three-dimensional [4]. The development of sustainable and livable cities therefore heavily depends on studies of human mobility across all three dimensions. While brand-new skyscrapers are being erected every day globally, we know surprisingly little about vertical human mobility. For instance, some recent studies propose to study vertical displacements in vertically integrated mixed-use developments as a means to identify key spatial connectors that have a direct influence on social interactions [5]. Moreover, a systematic study of vertical human mobility would benefit urban/infrastructure planning in many ways: e.g., targeted facility allocation in high-rise buildings, optimal placement of vertical nodes based on the typology of the building and the estimated vertical transportation load, effective vertical integration of a building in its neighborhood, etc. [6].

Horizontal mobility has been extensively studied using a wide variety of data sources: e.g., census data, travel surveys, CDR, location-based social network services, GPS [7], and smart travel/transit cards [8]. However, none of these approaches and sensors can be used effectively to track vertical displacements. Interestingly though, the sensor technology required to accurately track such human vertical mobility is readily available. What is missing is a methodology that enables the accurate identification of various types of possible vertical displacements from the output of large-scale human experiments with sufficient statistical significance.

Here, we report a contribution towards that goal by introducing and validating a methodology to accurately and systematically identify the sparse human vertical displacement activity (VDA) [9] that is deeply embedded within the predominantly horizontal displacement activity. This particular methodology is then integrated into a machine-learning-based classifier capable of dealing with large-scale data sets collected in free-living and unstructured urban environments. Classically, barometers have been the primary type of sensor used to track motion in the vertical direction. Indeed, barometric pressure—possibly augmented by other sensors—is commonly used in the field of Human Activity Recognition (HAR) to recognize the particular VDA class, which is of prime interest to us [9]. Specifically, VDA is a particular human activity class that deals with the vertical displacement of individuals in the built environment through commonly available modes of vertical mobility such as stairs, escalators, elevators, or slopes. In this work, the term VDA is intended to solely encompass human movements in vertically built structures. That means that we are discarding changes in elevation associated with any vehicle motion (motor vehicle, train, bicycle, cable car, etc.).

Our methodological advancement is thoroughly tested and validated using a big data set obtained from a large-scale human experiment carried out in Singapore: the so-called National Science Experiment (NSE). The NSE was a city-scale experiment that involved 50,000 students in Singapore between 2015 and 2017. The wearable devices specifically designed for this large-scale experiment were carried by students continuously for 5 days, and contained several sensors including a barometer and an accelerometer. By fully understanding the complex interplay of factors that influence barometric pressure, we develop several preprocessing methods to alleviate the effects of those factors, and with the end goal of achieving the highest possible accuracy in the VDA identification process. Moreover, the VDA extraction process must be robust enough to handle inherent limitations associated with such large-scale human experiments—i.e., low sampling rate, heterogeneity in devices and participant population, missing data, sensor errors, etc. As part of this process, we label manually a large number of training data (81 subjects, 81 devices, for a time period of 24 hours). This step is followed by a validation using a short-term video-annotated data set (2 subjects, 5 devices, for a time-period of 6 hours). We integrate the developed VDA identification methodology into a machine-learning based classifier, and subsequently applied to the large-scale NSE data set to extract some unique features of human vertical mobility associated with the student population participating in the NSE.

The main contributions of this paper can be summarized as follows (see Fig. 1):

- A novel and accurate multi-sensory identification of vertical displacement activity is developed and validated against a sparse data set from a large-scale human experiment involving over $16,000$ individuals going about their daily activity within a densely urban environment.
- The accuracy of this VDA identification process is found to be strongly dependent on a number of constraints associated with the sensing of key physical quantities. Specifically, we design a feature extraction step involving a combination of feature engineering and data segmentation. In addition, the properties of the sensors and how they are used in such large-scale experiments create a number of challenges, which are identified and addressed.
- A machine-learning based classifier using our novel VDA identification process allows us to carry out the first large-scale analysis of human vertical mobility in a city-scale experiment. Interestingly, our results reveal a highly heterogeneous distribution of vertical activity, both in terms of the number of events and of the size of vertical jumps. These results have far-reaching implications for the architectural design of dense urban environments.

II. Related works

In the field of HAR, sensor data requires pre-processing followed by the application of recognition models to classify the activity classes of interest [11], [12]. Algorithms for identifying VDA from sensor data range from simple threshold-based models to sophisticated Machine Learning

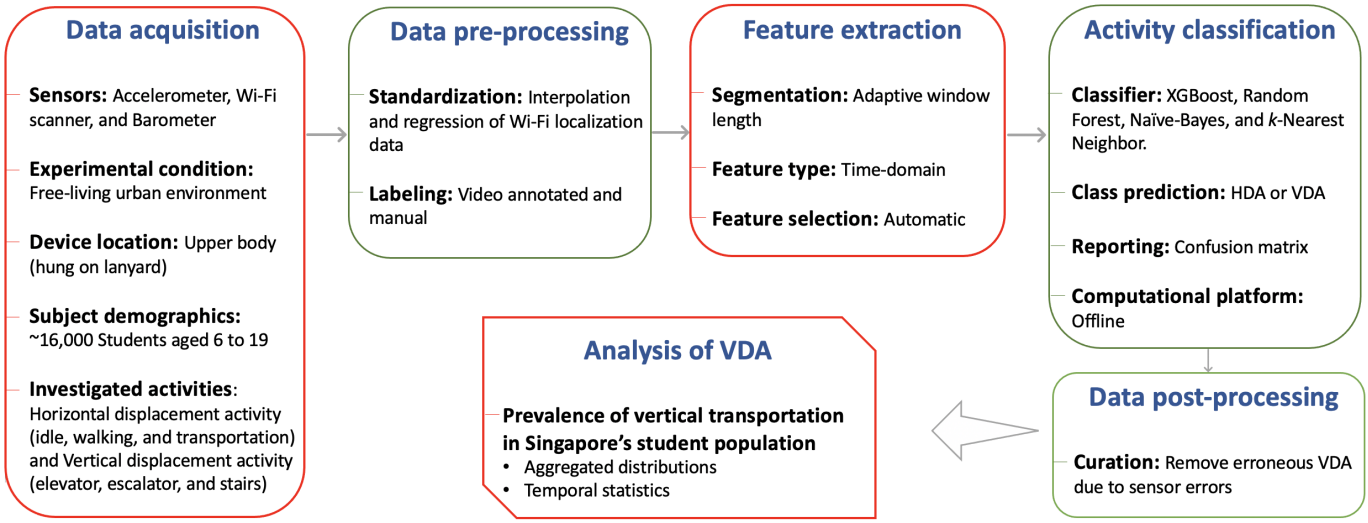


Fig. 1: Human activity recognition (HAR) process in the identification of VDA [10]. This flowchart describes the overall process developed to achieve our VDA analysis represented in the last rectangular box. The key contributions of this work are in the boxes with a red outline.

(ML) algorithms like deep-learning. In the present work, we employ a ML-based classification model and we review here several prior studies concerning the use of ML to classify VDA events.

One of the pioneering works on identifying VDA was performed in 1998 by Sagawa et al. [13], using accelerometer and barometer data to identify VDA. The study used a threshold-based model with a small training sample (83 minutes and 6 subjects). Since then, the combined effects of the digital revolution, the ubiquity of mobile devices, and the advances in sensors, Big Data and ML have paved the way for the possibility of real-time recognition of activity classes of individuals evolving in complex environments.

Accelerometers are the most widely used sensors to track human activities [14], [15]. In studies using an accelerometer as the stand-alone sensor, the prevalent mode of tracked vertical mobility is stairs climbing [11], [16]–[20], although the classification accuracy of stairs climbing tends to be lower than other activity classes [21], [22]. This led researchers to include additional data sources to improve the accuracy of VDA detection, such as gyroscope, magnetometer and barometer data [23]–[25], and also to consider other modes of vertical mobility such as escalator and elevator rides. For instance, Liu et al. [23] added barometer data to a model that used accelerometer, gyroscope, and magnetometer sensor data, notably improving the classification accuracy from $\sim 80\%$ to $\sim 90\%$ (number of subjects: 10).

Additionally, some studies have acknowledged that accelerometers are effectively less robust than barometers for VDA classification. Muralidharan et al. [26] compared the VDA recognition performance using an accelerometer versus a barometer, and showed that their VDA classification performances were similar — with the barometer-based framework performing slightly better at nearly 100% accuracy (number of subjects: 2). However, the accuracy of accelerometer-based framework dropped drastically when the mobile device was used to take calls or play games [26]. Similarly, Vanini et al. [27] showed that the classification performance of VDA was comparable for accelerometer-only and barometer-only study ($\sim 99\%$), but that the barometer was more energy efficient and less dependent from the on-body position than accelerometers (number of subjects: 10).

Our study focuses on the recognition of VDA as a general class of activity, and uses the following sensor data : (1) location data derived from Wi-Fi Access Points (APs), (2) magnitude of 3-axis accelerometer, and (3) barometric pressure data.

The magnitude of the 3-axis accelerometer signal is orientation independent [28], [29], and does not require complex data post-processing [11]. On the other hand, raw barometric sensor data can entail noise introduced by random sensor errors, limited sensor resolution, and high sampling frequency (> 2 Hz). Filtering techniques like moving average filters [30]–[32], Finite Impulse Response (FIR) filters [33], and Infinite Impulse Response (IIR) filters [24], [33]–[35] are commonly used to alleviate the noise effects. To increase precision in extracting elevation changes, signal modeling such as sinusoidal fitting model [36] and sigmoidal nonlinear fitting [37] are also used. In the present study, the spectral resolution of the sensor data collection is 0.06 Hz. Such low sampling rate allows the system to side-step noise appearing at high sampling rate that particularly affects precise extraction of elevation changes. Therefore, the barometric pressure sensor data we consider here are not filtered, and other sources of noise due to sensor resolution are used to quantify the uncertainty in the magnitude of the predicted VDA.

Barometric pressure data is usually converted to several common feature types such as : (1) statistical [38], [39], (2) spectral [38], [39], (3) temporal [38], [39], and (4) wavelet-based features [40], [41]. The most commonly used fea-

tures are the rate of change of pressure (vertical velocity or slope) [23], [30], [38], [42], [43] and differential pressure (dp) [42], [44].

Vanini et al. [27] used barometric pressure data alone to recognize VDA using the rate of change of pressure along with the standard deviation of differential pressure as features and found that Long Short-Term Memory (LSTM) neural network framework produces a 99% accuracy compared to a decision tree approach (96%) and naïve Bayes classifiers (93%). However, their data collection was of short duration (30 minutes for each class) and conducted in limited environments. Muralidharan et al. [26] detected floor changes with an accuracy of 99% using the J48 decision tree model. Even though several factors that affect barometric pressure are considered, the data collected were of short duration (few minutes), conducted in limited structured environments, and lacked any entanglement with transportation modes. Liu et al. [23] classified vertical displacement activities from horizontal displacement activities (HDA) using inertial measurement units (IMU, including magnetometer), and barometer sensor data, with barometric sensor features derived from the standard deviation of pressure and rate of change of pressure. By training various classifiers such as Random Forest, J48 decision trees, Artificial Neural Networks (ANN), SVM and Naïve Bayes, they obtained that Random Forest classifiers produced the highest accuracy of 92%. Also in this study, each activity class was performed only for a few minutes and limited to ambulation [23].

The review of the literature on HAR [11], [12], [45]–[47] makes it clear that no classifier can be considered as the best one in a universal way, i.e. without considering the context in which it is used. As each data set comes with its own set of distinct characteristics, the classifier working best for a particular data set and activity type might not have the best performance for a distinct problem or different circumstances (i.e., not generalizable) [11]. In this study, we have chosen two ensemble models (XGBoost and Random Forest), a Bayesian model (Naive-Bayes), and an instance-based nearest neighbor model (k-Nearest Neighbor) to evaluate and compare the performances of each model on our data set (see Fig. 1).

It is common in the HAR literature to use short-duration training data collected in segments that contain only one or two activity classes and are performed in semi-natural or laboratory conditions, with limited variability in environments. However, real-life activities of human occur in complex and unstructured environments, with a wide range of possible sequences, spanning heterogeneous activity classes with heterogeneous durations. Our study collocates itself in such a framework, as it uses a long-term (5 days) data set collected in a large-scale student population ($\sim 50,000$ students) during their routine weekdays. Hence, it requires a different approach than those reported in the literature.

Indeed, long-term monitoring of human activities requires a thorough understanding of all the factors that affect the sensor data in different static and dynamic environments. In particular, the factors that influence barometric pressure data are climate and weather, air velocity during motion, built environment, altitude, and sensor accuracy [9]. For a detailed review on the use of barometers to track human activity and the many factors that affect barometric pressure, we refer the reader to the recent review paper [9].

III. Description of Data

A. National Science Experiment data

The National Science Experiment (NSE) was designed and commissioned by the Singapore National Research Foundation (NRF) and the Singapore University of Technology and Design (SUTD), with other private and government bodies in Singapore [48]. The primary objective of this island-wide science experiment carried out by Singapore students—themed "Step Out for Science"— was for students to monitor and evaluate their own carbon footprint, travel mobility patterns, amount of time spent indoors and outdoors, and more.

Almost $50,000$ students from 92 schools distributed nationwide participated in the NSE in 2016 (Table I). Each student carried a wearable device called SENSg (see Fig. 2), which consisted of built-in environmental, motion sensors and communication units. The devices were able to record and transmit the sensed data related to the Activities of Daily Living (ADL) of the students to a cloud server [49]. The data was recorded every $13 \sim 18$ seconds over weekdays, from Monday to Friday, for 8 different weeks in 2016. To reduce battery consumption, the device goes to a sleep mode when the processed IMU signal shows no user movement. The SENSg devices were handed over to the students on Mondays and collected back on Fridays. Hence, the full time scale of daily ADL is only available on Tuesdays, Wednesdays and Thursdays. Table II shows the cleaned NSE 2016 database after removing devices based on two criteria (a) data coverage for less than 6 hours per day (b) percentage of missing location data larger than 50% in a day.

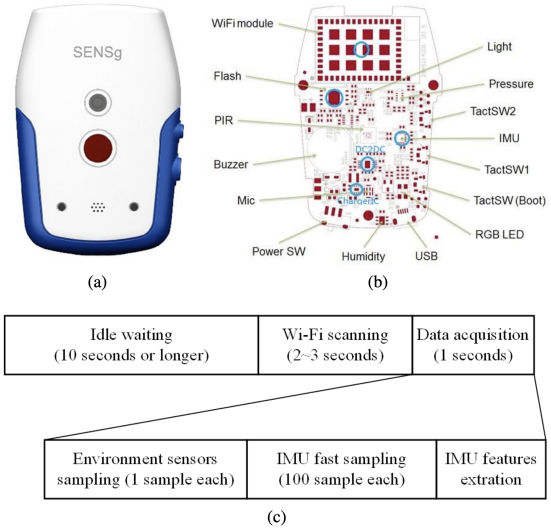


Fig. 2: NSE SENSg device details: (a) outside look, (b) internal structure [49], (c) working cycle. (Picture Courtesy: [50])

Our study uses the accelerometer, barometer and Wi-Fi scanners embedded in the SENSg devices to detect the Vertical Displacement Activites (VDA) of students during their ADL.

TABLE I: **NSE 2016 - Device delivery information**

Dates	No. schools	No. devices	Primary (6–12 y.o.)	Secondary (12–16 y.o.)	Pre-University (16–19 y.o.)
Apr 11–15	7	3,002	1,590	1,192	220
Apr 18–22	7	3,290	2,300	990	–
May 16–20	10	2,673	820	1,624	229
May 23–27	2	840	–	220	620
July 11–15	32	10,751	3,834	6,677	240
July 18–22	23	7,934	5,233	2,181	520
July 25–29	8	15,029	220	890	13,919
Aug 17–19	3	6,000	–	–	6,000
Total	92	49,519	13,997	13,774	21,748

TABLE II: **Curated NSE 2016 database**

Dates	No. schools	No. IDs	Primary (6–12 y.o.)	Secondary (12–16 y.o.)	Pre-University (16–19 y.o.)
Apr 11–15	7	1,797	1,049	608	140
Apr 18–22	7	1,793	1,265	528	–
May 16–20	10	700	239	386	75
May 23–27	–	–	–	–	–
July 11–15	33	4,387	2,048	2,114	73
July 18–22	21	3,086	2,292	666	128
July 25–29	8	3,345	–	93	3,057
Aug 17–19	3	1,380	–	–	1,380
Total	89	16,581	6,939	4,395	4,853

These sensor types are widely available in modern smartphones, rendering them ideal for this particular HAR. The SENSg device comprises the IMU sensor MPU9250 from InvenSense, and the barometer sensor BMP280 from Bosch Sensortech. The device also collected and stored up to a maximum of 20 Wi-Fi Access Points (AP) with the highest Receiver Signal Strength Indication (RSSI). We used Skyhook, a mobile location service from Boston, Massachusetts, that has geolocation of billions of Wi-Fi APs around the world, to convert the Wi-Fi APs to location coordinates [51]. The location coordinates (latitude and longitude) have a typical location accuracy of ± 100 meters. The location accuracy is increased by applying regression to the time series of location data (see Appendix B). When the Wi-Fi APs are sparse or absent, location data is considered as missing. Interpolation is then applied to the time series data to predict these missing values (see Appendix A). The full sensor characteristics are shown in Table III [49]. The embedded barometer sensor is capable of detecting up to 1 meter changes in height, i.e. ± 12 Pa [49]. The raw values measured by the accelerometer along its three axis were processed on-board the SENSg, and only descriptive statistics of these raw data were recorded: (1) $\max(M_{\text{acc}})$—the maximum value of the accelerometer's signal magnitude (2) $\text{std}(M_{\text{acc}})$—the standard deviation of accelerometer's magnitude, both sampled at 100 Hz during the one-second data acquisition temporal window, which occurred in its turn with frequency ~ 0.0625 Hz. The SENSg device along with its working cycle are shown in Fig. 2.

B. Video-annotated data

As the measure of the barometric pressure behavior is influenced by many factors, it is important to have at our disposal a data set with the corresponding ground truth for validation purposes. To this aim, we collected approximately 6 hours of sensor data annotated by means of video recording. This data set was recorded across different modes of horizontal (walking, idle, train, bus, car, and cycle) and vertical (elevator, escalator, and stairs) activity (Table IV). Data was collected by 2 researchers on different days and time using a SENSg device that was hung using a lanyard similar to the one used by students in the NSE. The video was recorded using a Go Pro Hero 6 mounted on the chest. We will use this data set to validate manual labeling methods described in Sec. IV-A.

TABLE III: **Sensor Characteristics of SENSg device**

Sensor	Model	Range	Accuracy	Units	Poll frequency
Wi-Fi	SN8205	–	–	–	0.062 Hz
Barometer	BMP280	300 to 1,100 hPa	± 12 Pa	Pa	0.062 Hz
Accelerometer	MPU9250	± 2 g	± 80 mg	mg/LSB	100 Hz (for every 0.062 Hz)

TABLE IV: **Video-annotated data**

Mode	Total time (hours)
Car	0.4
Bus	1.5
Train	0.7
Cycle	0.8
Walking	1.5
Idle	0.3
VDA	0.4
Total	5.6

IV. Methodology

As the NSE was conceived as a large scale data collection with relatively high temporal resolution and for long durations, sensor data collection was optimized to save battery life and data bandwidth. This resulted in a compromise in sensor resolution, sampling rate, and type of sensor data collected, which inevitably makes the VDA identification more challenging. This section thus details the machine learning framework that we developed to identify and extract VDAs from continuous temporal segments of the NSE data (Fig. 1). Section IV-A explains the manual labeling techniques, and sections IV-B, IV-C, IV-D, and IV-E encompass the machine learning framework. We refer to the Appendices for details related to the pre-processing of the location data (App. A and B), while App. C describes the classification model parameters, and App. D deals with model tuning.

A. Manual labeling

Accurate annotation of sensor data is highly manpower intensive. One solution is to manually label a small subset of the data based on expertise, and use it to validate the classification framework. We were indeed able to perform such a manual labeling in a subset of the NSE database (see Sec. IV-D), using our general understanding of the factors affecting

barometric pressure [9] and leveraging the data annotated with the help of the ground truth video (Sec III-B). In this manual labeling, we labeled each data point as either VDA or HDA, based on the unique characteristic profile of vertical transportation. As noted in [9], pressure changes caused by factors other than vertical transportation are mostly long-term variations (e.g. diurnal pressure cycle) and/or brief and intense transient spikes (e.g. Indoor-to-outdoor transition).

First, and based on our pilot experiments illustrated in [9], we distinguish VDA from HDA in our manual labeling by monitoring the following characteristics: (1) a low horizontal travel velocity, (2) body movements picked up by the accelerometer, and (3) an increase or decrease in barometric pressure. We note that high magnitude pressure changes that can be mistaken for VDA occur during transportation modes [9], but these transportation modes can be identified by tracking the location data trend, which should then reveal a high horizontal velocity.

For validation of the procedure, the classifier will be trained on the manually labeled data set, and then applied to the data set described in Sec.III-B, for which the ground truth is available.

B. Data segmentation

We segment the time series sensor data into regions of significant and minor pressure changes to allow the classifier to focus on the main characteristic of VDA—i.e., the pressure-altitude relation. First, each data point is considered to a have significant pressure change based on a cut-off value ($dp_{i,\text{cut-off}}$), determined by considering three points: (1) we want to discard small pressure jumps that could correspond to other factors that yield pressure changes of similar magnitude such as slopes, indoor-outdoor transitions, etc. [9], (2) we want to consider pressure changes corresponding to at least 50% of the minimum vertical displacement of a single floor, and (3) the changes need to be consistent with the sensor resolution (± 12 Pa).

The change in pressure (dp_i) for each data point is calculated from the difference of the time series data $dp_i = P_i - P_{i+1}$, where P_i is the pressure datum at instant t_i. The time interval $dt_i = abs(t_i - t_{i+1})$ associated with dp_i should be less than a cut-off value $dt_{i,\text{cut-off}}$ to take into account that data can be missing in times of inactivity, and the resulting variation in pressure values for large values of dt_i might then be due to the diurnal pressure cycle. The consecutive significant pressure changes in the same direction (positive change or negative pressure drop) are then grouped to form a segment, i.e. $S_{P_{i,n+1}} = \{P_i, P_{i+1}, \ldots, P_n, P_{n+1}\}$ for a pressure change sequence of $S_{dp_{i,n+1}} = \{P_i - P_{i+1}, P_{i+1} - P_{i+2}, \ldots, P_n - P_{n+1}\}$. Other features and sensor data are grouped using the same groups of indices as for the segmented pressure sequences.

In the case of manually labeled time series data set, we first label each data point as either VDA or HDA (Sec. IV-A). Therefore, each segmented data might contain both data points labeled VDA and data points labeled HDA. We thus label each segment using a majority rule, and assigning a VDA label in case of a draw. A perfect data segmentation would allow each VDA segment to indicate a complete VDA event with no false positives or false negatives. The choices of $dt_{i,\text{cut-off}}$ and $dp_{i,\text{cut-off}}$ ultimately determine the performance of this method. Hence, we compute the F_1 score (a.k.a. F-measure of balanced F-score) of capturing a complete VDA event in each segment labeled as VDA in the manually labeled training data for a range of $dt_{i,\text{cut-off}}$ ($[30, 50, 90, 120]$ sec) and $dp_{i,\text{cut-off}}$ ($[20, 23, 25, 27, 30]$ Pa) values (Table V). The best F_1 score is obtained for values $dp_{i,\text{cut-off}}$ and $dt_{i,\text{cut-off}}$ equal to 25 Pa and 120 seconds, respectively. We thus perform the data segmentation with these parameter values, and the final classification described below will be performed on these segmented data.

TABLE V: Selection of data segmentation parameters based on F_1 score.

$dt_{i,\text{cut-off}}$	F_1 score (%)				
120	97.68	98.09	**98.15**	96.98	96.98
90	97.65	98.06	98.12	97.94	96.95
60	97.65	97.99	98.06	97.87	96.88
30	97.14	97.39	97.45	97.23	96.22
$dp_{i,\text{cut-off}}$	20	23	25	27	30

C. Feature engineering

Model explainability is a growing focus in Machine Learning. To improve explainability, it is natural to start from features based on domain-specific knowledge. We use here our data exploration and our understanding of the sensor data and the target event to be recognized to design several domain-specific features. Specifically, we compute the following features from accelerometer, barometer and Wi-Fi localization data.

Rate of pressure change dp/dt: It accounts for the pace of the VDA. This distinguishes the elevation change based activities from phenomena that unravel over slow temporal scales such as sensor drift, diurnal pressure cycle, etc.

Modified zero-crossing rate $\widetilde{zcr}$: The zero-crossing rate zcr is a temporal feature that counts the number of sign changes during a particular time window for a given signal. Here, we modified this feature to count changes in sign only if the corresponding magnitude difference in pressure is ≥ 20 Pa. This conditioned $\widetilde{zcr}$ can indeed identify the pressure spikes due to factors such as weather and climate, built environment, air velocity during motion or sensor accuracy, and distinguish them from one-directional pressure changes that occur during changes in elevation.

Horizontal travel velocity dx/dt: as the location data are recorded in terms of latitude and longitude, we use the haversine formula (see Eq (1)) to calculate the great circle distance x between two locations. It is based on the assumption that the Earth is approximately spherical, a valid assumption for small distances such as the ones measured in the NSE data. The horizontal travel velocity dx/dt plays a key role in differentiating significant pressure changes of VDA from transportation based activities [9]. Specifically, the great-circle

distance x between location coordinates (φ_1, λ_1) and (φ_2, λ_2) with φ the latitude, λ the longitude, and r the radius of earth, is given by:

$$x = 2r \arcsin\left(\sqrt{\sin^2\left[\frac{\varphi_2 - \varphi_1}{2}\right] + \cos\varphi_1 \cos\varphi_2 \sin^2\left[\frac{\lambda_2 - \lambda_1}{2}\right]}\right). \quad (1)$$

Statistical features of immediate neighborhood in time series data N_i: By definition, a VDA event is always preceded and followed by a HDA event. However, during vertical mobility modes like elevator (and sometimes during escalator rides), a person is potentially standing with no significant body movement. Likewise, during transportation modes such as car, bus or train travel, the vehicle stops intermittently, leading to regions of low horizontal travel velocity. Hence, calculating statistical features over an immediate neighborhood of each data point can entail a sequence of events during and around an activity of interest. More precisely, for each data point i, we compute on the time window $N = \{i-2, i-1, i, i+1, i+2\}$ (of approximate width ~ 80 seconds) and for both the horizontal travel velocity dx/dt and the accelerometer data (max(M_{acc}) and std(M_{acc})) the ten following statistics: Minimum, Maximum, Average, Median, Mode, RMS (Root Mean Square), MAD (Median Average Deviation), Standard deviation, Variance, and IQR (Inter Quartile Range).

Statistical features of segmented data S_i: Many of the statistical features considered for the immediate neighborhood of time series data are not suitable for segmented data, as the length of the sequences is typically very small for most VDA events—$2 \sim 4$ s or even less ($1 \sim 2$ s) for transportation modes during which the sign of slope dp/dt changes very often. Hence, we compute only the mean, median, and mode of each data segment. This is done both for the original sensor data and for the statistical features of immediate neighborhood in the time series of the modified zero-crossing rate $\widetilde{zcr}$, of the horizontal travel velocity dx/dt, and of the accelerometer data (max(M_{acc}) and std(M_{acc})).

In total, we compute 95 features of the segmented data (Sec IV-B). We refer to this set of features as Feature set-I.

D. Training-Validation-Test data

The NSE 2016 data set is very diverse in terms of the number of students, unique devices and demography of individual participants. Hence, each train-validation-test data set should reflect this diversity. To ensure this, we select the data collected during one day by 81 students from 81 different schools, with an appropriate balance of school types (primary, secondary and pre-university) and weekdays (limited to Tuesday, Wednesday, and Thursday) to form a representative sample (see Table VI. We manually label this data sample according to the steps described in Section IV. This sample data set is then randomly divided into training (80%) and test set (20%), with all the data collected by any single device assigned either to training or testing as a whole (data collected by a single student cannot be split between training or testing). The classifier model is trained on the training set with the model's hyperparameters being tuned by a 5-fold cross-validation. Once the best model parameters are identified, it is then tested against the test set for the final performance evaluation.

In addition, to validate the manual labeling, we use the video-annotated data described in Sec.III-B. We apply the trained classifier model to these data labeled using the corresponding video recording to establish the ground truth.

E. Classification models

We have selected four commonly used classifiers in HAR [11]: (1) Extreme Gradient Boosting (XGBoost or XGB), (2) Random Forest (RF), (3) Naive-Bayes model (NB), and (4) k-Nearest Neighbors (kNN). Both XGBoost and Random Forest are decision-tree-based ensemble learning algorithms. The XGBoost algorithm is based on the boosting method that adds weak learners sequentially to reduce the loss function of the model, while the Random Forest model is based on the bagging method that adds weak learners in parallel, and uses majority voting model to make final predictions. On the other hand, the Naive-Bayes model is a probabilistic learning algorithm based on Bayes' theorem that assumes strong independence between the features. As for the k-Nearest neighbor model, it is non-parametric and uses distance-based measures to find the k-nearest samples, and it uses a majority voting model to assign a class.

The hyperparameters of these classifiers are tuned by means of a grid search using a 5-fold cross-validation on feature set-I. A more detailed description of the model parameters and model tuning can be found in App. C and D respectively.

V. Results and Discussion

The central objective of our study is to obtain the magnitude of vertical displacements of individuals during their daily activities as an indication of vertical movement in a large-scale study of human mobility. By leveraging the relationship between barometric pressure and altitude, our specific aim is to develop pre-processing methods that alleviate the adverse influence of the other factors that affect barometric pressure in order to accurately extract the instances of vertical motion.

We globally consider the following procedure. First, a small representative sample of the cleaned NSE 2016 data (Sec IV-D) is manually labeled as described in Sec IV-A. We then segment the whole cleaned NSE 2016 data set from Table II along with the manually labeled data and video annotated data according to Sec IV-B. The classifier model is then trained on the segmented manually labeled data to achieve a binary classification between Horizontal Displacement Activity (HDA) or Vertical Displacement Activity (VDA). The trained classifier is applied to the video annotated data to validate manual labeling, and finally applied on the cleaned NSE 2016 data set.

A. Performance comparison of classification models

The four classifiers described in sec. IV-E are trained using feature set-I with the respective hyperparameters tuned from the 5-fold cross-validation with grid search (see Sec. IV-E). Their respective classification performances on the test data set are reported in Table. VII. The XGBoost model provides the highest overall accuracy (98 %) and F_1-score for classifying VDA (93 %), closely followed by the Random Forest (overall

TABLE VI: Properties of the Training-Test split data sets

data set	Ground truth	Size in duration	Number of time-series data points	Number of segmented data	Number of devices	Number of individuals	Number of schools	Pri.	Sec.	Pre-U.
Training (80%)	Manual	64 (days)	207,380	7,132	64	64	64	21	20	23
Test (20%)	Manual	17 (days)	55,092	2,166	17	17	17	8	2	7
Video annotated data	Video recording	6 (hours)	1,262	305	5	2	–	–	–	–

accuracy of 97 % and F_1-score for classifying VDA at 89 %). Both the Naive-Bayes model and k-Nearest Neighbor model perform poorly with an F_1-score for classifying VDA standing at 52 % and 17 % respectively. The ensemble learning methods clearly show a superior performance than Naive-Bayes and kNN models, similar to the results reported by Liu et al. [23] in classifying VDA from HDA. Therefore, we select XGBoost as the classifier of choice for this analysis.

TABLE VII: Comparison of classifier performance on the test data set with feature set-I

Model	Overall accuracy (%)	Class	Precision (%)	Recall (%)	F_1-score (%)	Support
XGB	98	HDA	99	99	99	1899
		VDA	93	92	93	277
RF	97	HDA	98	99	99	1899
		VDA	96	83	89	277
NB	80	HDA	97	80	88	1899
		VDA	38	83	52	277
kNN	87	HDA	88	99	93	1899
		VDA	52	10	17	277

B. Validation of manual labeling

To ensure the validity of our manual labeling strategies, the trained XGBoost classifier model is applied to the data set with ground truth annotated from a video recording. The results of the classifier performance is shown in Table VIII. Although the classifier has 100% precision (i.e. it does not capture false positives), it has a recall of 80% (i.e. it only captures 80% of the actual VDA events), thereby yielding an F1-score of 89%.

TABLE VIII: **Classification results of video-annotated data set with feature set-I**

(a) **Performance metrics**

Class	Precision (%)	Recall (%)	F1-score (%)	Support
HDA	95	100	97	246
VDA	100	80	89	59
weighted average	96	96	96	395

(b) **Confusion matrix**

Class	HDA	VDA	Support
HDA	246	0	246
VDA	12	47	59

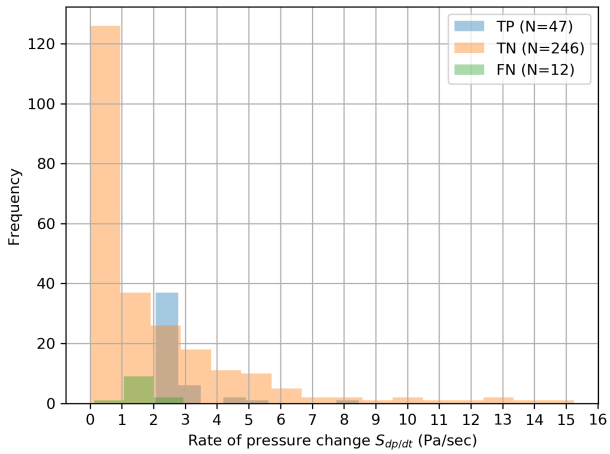


Fig. 3: Distribution of the rate of pressure changes in predicted data in video-annotated data set when using feature set-I.

The recall performance is rather low due to a high proportion of instances such that $dp/dt < 1.9$ Pa/sec in the video-annotated data set (16.9% of all points marked as VDA) compared to the test data set (5.5%) for example. This issue is also responsible for some false negatives in the predicted data, where 77% of the predicted false negatives in video-annotated data set have $dp/dt < 1.9$ Pa/sec (see Fig. 3). This is due to the fact that many of the data points with small pressure jumps dp are not labeled as VDA in the manually labeled data set. Indeed, many instances with similar magnitudes for dp cannot be ruled out with high confidence given the known factors influencing barometric pressure [9].

C. Impact of sensor type and feature importance

Three sensors are employed for this study: Barometer, tri-axis accelerometer, and Wi-Fi scanner (location data). To understand the impact of these sensor types on the classification of VDA, we conduct an ablation study. Since barometric pressure data is used in the pre-processing step (data segmentation) and is vital to the extraction of vertical displacements, features derived from barometer data are not removed in this ablation study. Table IX reports the results of the classification performance with features from the following sensor(s): (a) Barometer only, (b) Barometer and tri-axis accelerometer, and (c) Barometer and Wi-Fi scanner (location data). The F_1-score for classifying VDA is 57% (overall accuracy of

89%) with barometer features only, and it is significantly improved by the addition of the tri-axis accelerometer features (overall accuracy of 96% and F_1-score for classifying VDA at 83%). An even better performance is achieved with the combination of location and barometer features with 97% overall accuracy and 90% F_1-score for classifying VDA. The addition of the accelerometer data to barometer and location data thus only improves the F_1-score by 3%. As noted in Sec. II, previous studies [26], [27] have obtained similar accuracy with accelerometer only or barometer only features, but their training data was collected in limited environment (e.g. ambulation only or transportation only activities, short time period, etc.). However, for a data set such as the one considered here, which was collected in a long term free-living environment, it thus seems that barometer only features are not enough to classify VDA with high accuracy; however, adding the data of even only one additional sensor data allows to reach very good performances even in this case.

TABLE IX: Impact of sensor type on the classifier performance with selected features of feature set-I. Baro - Barometer, Acc - Accelerometer, and Loc - Location data derived from Wi-Fi scanner.

Sensor	Overall accuracy (%)	Class	Precision (%)	Recall (%)	F_1-score (%)	Support
Baro.	89	HDA	94	94	94	1899
		VDA	58	56	57	277
Baro + Acc	96	HDA	98	97	98	1899
		VDA	83	84	83	277
Baro + Loc	97	HDA	98	99	99	1899
		VDA	92	88	90	277
Baro + Loc + Acc	98	HDA	99	99	99	1899
		VDA	93	92	93	277

The XGBoost model (see App. C for a description of the model and its hyperparameters) has an embedded feature ranking method that quantifies the importance of each feature to build the predictive model. The most relevant parameter to quantify relative feature importance is the total gain that measures the improvement in accuracy brought on by the feature for each tree in the model. Based on the total gain, the feature importance of all 95 features is calculated from the tuned model. To understand the impact of using reduced feature sets, we re-tune the hyperparameters and re-train the classifier model using the top 10 (out of 95) and the top 5 (out of 95) of the important features, denoted as the feature set-II (features 1–10 in Table X) and feature set-III (features 1–5 in Table X) respectively, using the same procedure as described in App. D. The reduced feature set size of 10 and 5 are arbitrarily chosen, but it is supported by the fact that the total gain drops drastically after the most important feature ($S_{dp/dt}$) as shown in Table X.

The classification performance of the XGBoost model with feature sets I, II, and III are shown in Table XI. The overall classification accuracy of the model for all feature sets on the test set stands at 98%, and the F_1-score for classifying VDA is found to be slightly lower at 93% for feature set-I and 92% for feature set-II and III. As the number of features are reduced from 95 to 10, and ultimately down to 5, there is a very slight increase in precision with a complementary decrease in recall. The confusion matrix in Table XIb, shows that there is no significant drop in performance when reducing the feature set size. To strike a balance between performance and number of features, we use the XGBoost model results with feature set-II in this section's further analysis.

TABLE X: Feature importance obtained from the tuned XGBoost classifier model. Top 10 results are shown. S stands for segmented data.

No	**Feature**	**Total gain**
1	$S_{dp/dt}$	4460
2	$S_{\min(N_{\text{RMS(std}(M_{acc}))})}$	2614
3	$S_{\min(N_{\text{median}(dx/dt)})}$	988
4	$S_{\min(\widetilde{zcr})}$	421
5	$S_{dx/dt}$	265
6	$S_{\min(N_{\max(M_{acc})})}$	184
7	$S_{\min(N_{\max(dx/dt)})}$	184
8	$S_{\min(N_{\text{avg}(dx/dt)})}$	141
9	$S_{\max(N_{\max(dx/dt)})}$	124
10	$S_{\min(N_{\min(dx/dt)})}$	114

TABLE XI: **Classification performance of the XGBoost model**

(a) **Performance metrics**

Feature set	Overall accuracy (%)	Class	Precision (%)	Recall (%)	F_1-score (%)	Support
Feature set-I	98	HDA	99	99	99	1899
		VDA	93	92	93	277
Feature set-II	98	HDA	99	99	99	1899
		VDA	93	90	92	277
Feature set-III	98	HDA	98	99	99	1899
		VDA	94	90	92	277

(b) **Confusion matrix**

Feature set	Class	HDA	VDA	Support
Feature set-I	HDA	1870	19	1899
	VDA	22	255	277
Feature set-II	HDA	1870	19	1899
	VDA	27	250	277
Feature set-III	HDA	1874	15	1899
	VDA	29	248	277

D. VDA classification performance

Table XIb shows the confusion matrix—i.e. predicted class distributions and corresponding Type-I (False Positives) and Type-II errors (False Negatives). When closely inspecting the feature space in the test data set, one finds that the Type-II error generally occurs when the rate of pressure changes

$S_{dp/dt}$ is small, typically below 1.9 Pa/sec for 65% of the false negatives. This observation may have two possible origins. Either, this is due to the small magnitude of pressure jumps dp (associated to low recording frequency that splits a single VDA event across multiple time intervals). Alternatively, this could also be due to larger time intervals dt (due to irregular recording frequency of $0.076 \sim 0.016$ Hz). In both cases, the low and irregular sampling rate is the key limiting factor. In some cases, a vertical mobility event is following or preceding an horizontal transportation mode—e.g., above-the-ground or underground train travel—, which may lead to misclassification due to the reliance of Wi-Fi localization for the calculation of the travel velocity. This is the case for instance when the APs are sparse.

E. VDA recognition limitations

The magnitude of the altitude change in a given VDA event is derived from the barometric pressure change of the segmented data S_{dp} using Eq (2). The accuracy of this magnitude is limited by the barometric sensor resolution, which is ± 12 Pa for our SenSg device (see Table III). Some of the VDA events with lower values of $S_{dp/dt}$ are not appropriately classified, as stated in Sec. V-D. For a typical sampling frequency of 0.062 Hz, this corresponds to an altitude change of 2.5 meters (30 Pa). Hence, it is reasonable to assume that vertical displacements smaller than this value are not properly captured, and thus the vertical moves can only be accurate with a vertical resolution of ~ 2.5 meters.

The classification performance when recognizing VDAs in this study is thus limited by the type of sensor data, sensor resolution, and sampling frequency. Location data from GPS with an accuracy of ± 10 meters would outperform the often inaccurate Wi-Fi localization, which is only accurate within ± 200 meters. Similarly, a higher sampling rate of barometric pressure to the tune of 1 Hz would be ideal compared to the lower sampling frequency of ~ 0.062 Hz in this data set. Moreover, the use of accurately annotated data set—e.g., by means of video recording—can markedly improve the training performance.

VDA can be further sub-classified into different modes of vertical transportation like elevator, escalator, or stairs. This would however require large amounts of video-annotated training data set with high temporal and spatial resolution. This is because the rate of change of vertical displacement—a key metric in distinguishing between these modes [9]—needs to be sampled at a fairly high resolution, especially to be able to differentiate between stairs climbing and escalators riding. Furthermore, the classification between stairs/escalators and elevators will only be accurate for floor jumps larger than $2 \sim 3$ if the sampling frequency is low, such as that found in our data set. For these reasons, we do not consider VDA sub-classification in this study. However, with the use of sensors with a higher sampling rate, such sub-classification of VDA should be attainable as shown in previous works [23], [26], [27], [52].

F. Prevalence of vertical mobility in Singapore's student population

We have applied the trained XGBoost model with featurre set-II to the curated NSE data described in Table II. The data was collected for $N = 16,486$ students from 89 schools aged 6–19 whose residences and schools were spread throughout the island-state of Singapore. For each subject, the data contains at least 6 hours of coverage during a day. The results presented in this section are aggregated over the duration of a single day for each participant—the selected day corresponds to the one with the largest among of data points collected during their week of carrying the SENSg device as part of the NSE program. The predicted VDAs are postprocessed to remove vertical displacements (less than 9% of total) that are accompanied by significant pressure fluctuations ($dp > 20$ Pa) that may have been caused by sensor errors (see [9]).

The total number of predicted VDAs in the entire data set reaches $182,841$ events after postprocessing. Some statistics of VDA for each subject over a day are calculated and shown in Fig. 4, namely the number of VDAs, the cummulated vertical displacement and the total time spent in the VDA mode. On a daily average, a subject was found to be engaged in 10 events of vertical mobility, traveled vertically ~ 83 meters, and spent a total of 4 minutes per day in this mode. As an element of comparison, Americans are found to spend on average $\sim$ 65 minutes per day eating [53]. The distributions are however rather heterogeneous, with individuals who traveled as much as 140 meters vertically in a single VDA event, moved up to 60 times in vertical direction and spent up to 24 minutes per day in VDA. The highest vertical displacement in our predicted data (140 meters) is nearly half the size of the highest building in Singapore—the Guoco tower stands at 290 meters.

Fig. 4 also shows the breakdown of the distributions of VDA for primary (PRI), secondary (SEC) and pre-university (PREU) students. Interestingly, the vertical mobility footprint of PREU students is markedly higher than that of PRI or SEC students, even if the descriptive statistics of single VDA events are very similar: the median values are 5.2 m for PRI students, 5.1 m for SEC students, 5.8 m for PREU students; the mean values are respectively 9.3 m (PRI), 8.3 m (SEC), 8.1 m (PREU) and the maximum values are 116 m (PRI), 120 m (SEC), 140 m (PREU). The fact that pre-university students are more active in terms of cumulated VDA can in particular be explained by the fact that they are more likely to travel through public transportation like trains that are either above or below the ground road level [54].

The timeline of VDA reveals a rich structure. In highly vertical cities, people experience significant waiting times during their vertical mobility due to their use of shared public transportation such as elevators or slow pedestrian movements in escalators and stairs. In Singapore, these waiting times can be similar in magnitude to the waiting times between trains or buses ($2 \sim 5$ minutes). Actually, businesses operating on appointment only often remind their customers to include the waiting time associated with elevator rides when planning their arrival. While these time scales can still be considered small, the ongoing trend of vertical integration of multi-purpose

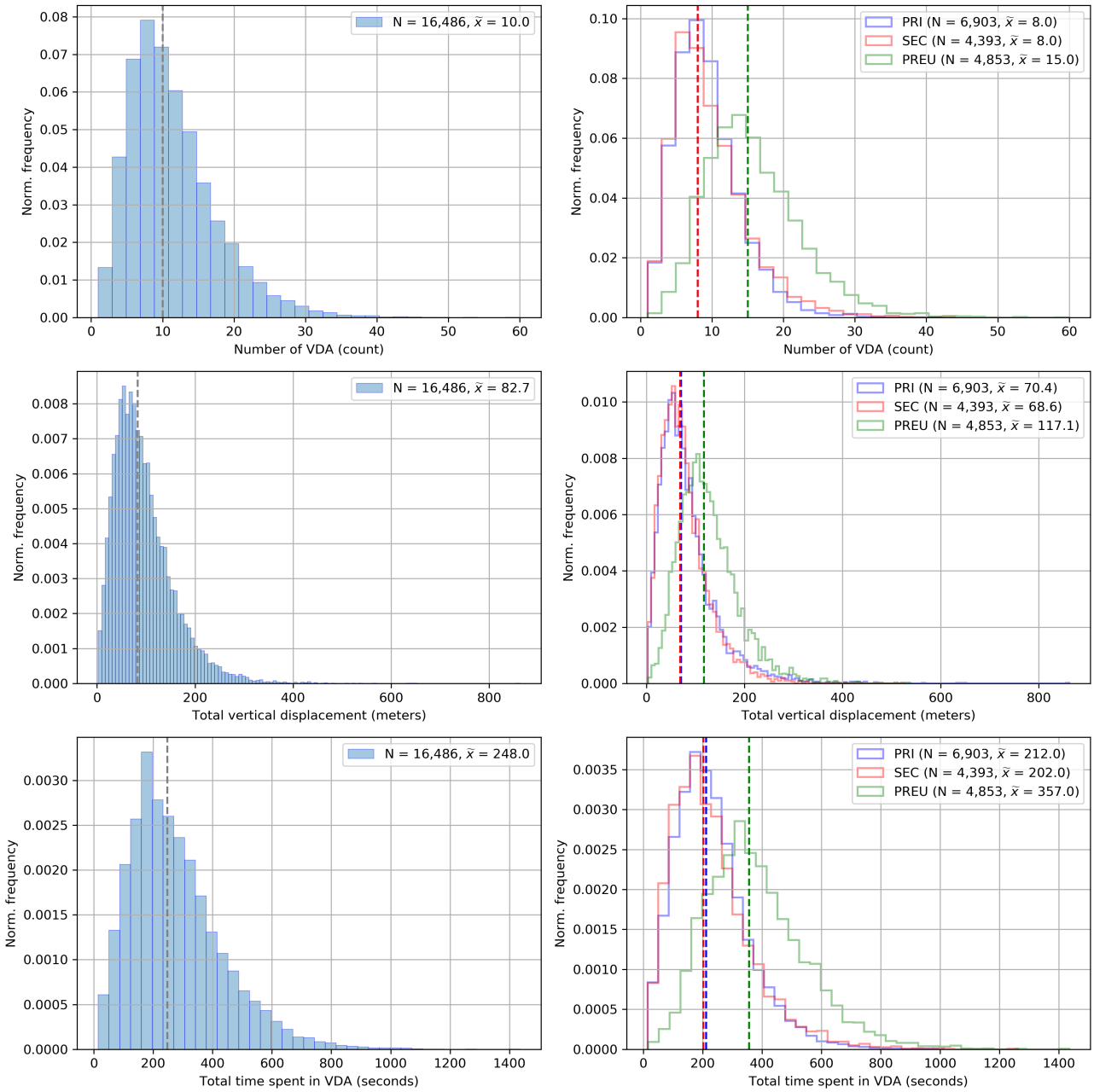


Fig. 4: Aggregated distribution of VDA for a population of $N = 16,486$ individuals. The aggregation period is set to 1 day. N is the number of subjects and $\tilde{x}$ denotes the median value (dashed lines). Top row: number of VDA events per subject; Middle row: total cummulated vertical displacement (in meters) per subject; bottom row: total time spent in VDA (in seconds) per subject; left column: total population; right column: subjects grouped by school types.

buildings will underscore the importance of understanding congestion in terms of vertical transportation and mobility. Understanding congestion per land use type and time is important to inform better transportation planning and this point will apply as well to vertical transportation.

Fig. 5 shows the hourly statistical distribution of VDA according to the three group types under investigation (PRI, SEC and PREU). The group-based total number of subjects with VDA (i.e., active subjects, Fig. 5(a)) and total count of VDA (Fig. 5(b)) exhibit similar trends; both indicate that different daily rhythms of individuals occur between the three groups, i.e,. the peaks in activity for PRI are at 7 a.m., 10 a.m., 1 p.m., and 4 p.m., SEC at 6 a.m., 10 a.m., 2 p.m., and 6 p.m., and PREU students at 8 a.m., 12 p.m., and 5 p.m. In addition, a large number of activities of PREU students is observed during evening and night times, which can readily be explained by the fact that some schools like the Institute of Technical Education (ITE), operate primarily in the evening and by the active after-school life of the late-teen population.

Due to the low number of active users between 12 a.m. and 5

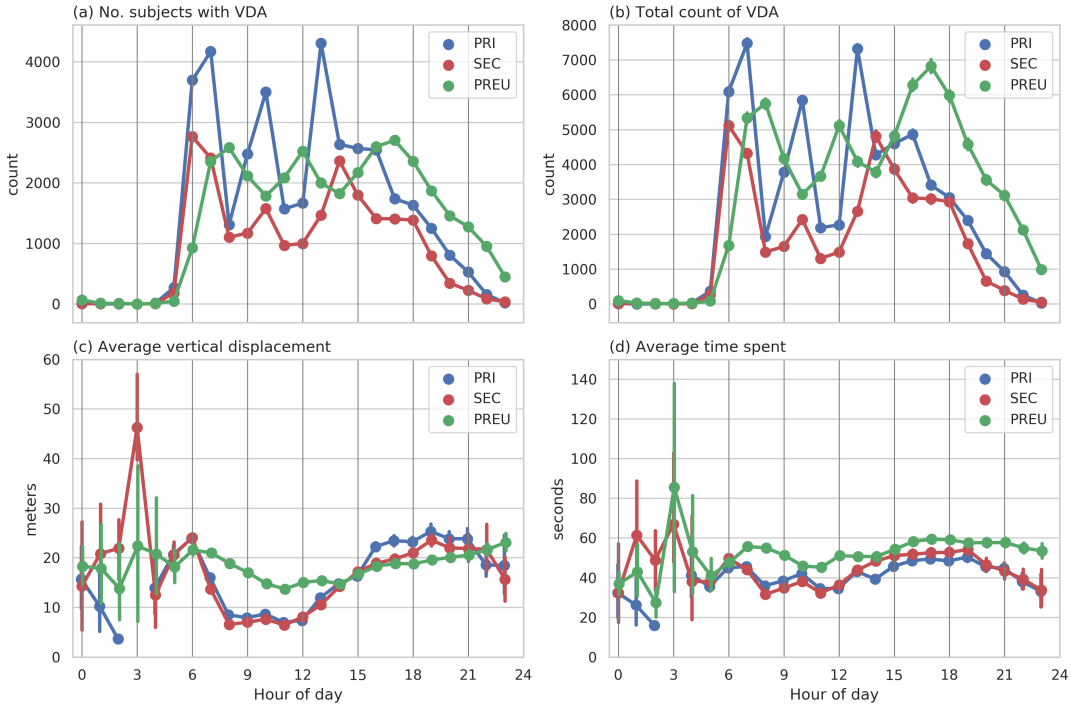


Fig. 5: Temporal statistics of VDA according to group types: (a) number of subjects with VDA (active subjects) by hours, (b) total VDA count by hour, (c) average vertical displacement per active subject by hour, and (d) hourly average time spent in VDA per active subject by hour. The errorbars in (c) and (d) indicate the confidence intervals estimated using a bootstrap approach ($n = 1000$).

a.m., the average vertical displacement (Fig. 5(c)) and average time spent in VDA (Fig. 5(d)) shows large uncertainty within that particular time period, and become stable again after 5 a.m. In terms of per-subject average vertical displacement and time spent, the primary and secondary students show similar patterns, while pre-university students yield markedly different patterns. For instance, the primary and secondary students' average vertical displacement is high during 5 a.m.–6 a.m. time window, drops to 9 meters between 8 a.m.–12 p.m., then peaks up again to $\geq$ 20 meters between 4 p.m.–9 p.m., and decreases after that. On the other hand, the pre-university students average vertical displacement is about 20 meters between 6 a.m.–7 a.m., slightly decreases to 14 meters around 11 a.m., then slowly increases until 11 p.m. These differences in the morning could be explained by the style of classes between the primary/secondary and pre-university students: primary and secondary classes are usually fixed and students do not need to change classes between courses, whereas pre-university students will need to move to other classes for different courses.

VI. Conclusion

Over the past two decades, advancements in sensor technology and complexity science have enabled the dissection, with unprecedented accuracy, of the fine details of human mobility in urban areas from large-scale data. Mining these "Big Data" revealed the burstiness and relative predictability of human spatial movements [55], [56], as well as highly dense spatial areas in cities [57]. However, these studies, albeit illuminating, were mainly carried out on large groups of people living and moving about in mostly flat cities, and were therefore limited to horizontal movements in two dimensions. The present study explored the application of identifying and monitoring VDA to highlight the prevalence of vertical mobility at city-scale, and its potential to add to the understanding of human mobility in general.

Previous studies on VDA identification focused on the feasibility of using barometric pressure sensing, hence the associated experiments were designed for and carried out in highly controlled conditions, e.g., structured environments, short-term activities, simplified barometric pressure behavior, limited activity classes, small-population samples, and small-scale areas. In the real world, human vertical movements are typically unstructured due to the nature of human living patterns as well as the constraints imposed by the built environment. Using our understanding of the barometric pressure data behavior and considering a more general scope corresponding to real-world conditions, we have developed an accurate VDA extraction methodology, integrated into a machine learning strategy to precisely identify daily life VDA in "Big Data" sets obtained from a city-scale experiment.

This "Big Data" set consisted of 16,581 students between the age of 6 and 19, from 89 schools spread throughout the city-state of Singapore. The relatively long-term tracking (5 days) and high recording frequency (every $\sim$ 16 seconds) has provided high spatiotemporal resolution data sufficient

to identify VDA in diverse conditions. The classification performance of the model is similar or closely aligned with the results reported in the literature (Sec. II)—with 98% overall accuracy and 92% F_1-score in classifying VDA—while significantly generalizing its applicability to almost all possible real-world conditions.

The recognition of VDA is instrumental to the effectiveness of a range of critical applications in the fields of indoor positioning and navigation [26], [31], [58], [59], estimation of energy expenditure, and health monitoring [34], [41]. The present study markedly expands the scope of applications by rendering possible the performance of large-scale human experiments with the ultimate goal of shedding light on the patterns of human mobility in dense urban areas, and its impact on urban planning.

Our study is ultimately limited by the participants age group and size, the sensor characteristics (primarily resolution and sampling rate), and the training data collection methods (video annotation versus manual labeling). Future studies can most likely achieve more significant results by improvements in one or more of these categories to obtain a more accurate and general detection of VDA.

Lastly, the analysis of the aggregated VDA data by population groups/segments has revealed rich details about the anticipated differences in the profiles of daily activity between late-teens and younger children. The uncovered patterns of vertical human mobility enabled us to accurately quantify, for the first time, the distance traveled and time spent in vertical transportation in a densely built urban environment. It has also revealed some unique patterns of activity related to vertical transportation that is present in many aspects of our human lives. There is no doubt that a more systematic analysis of the non-aggregated data would provide substantial new details and unique information about the dynamics of vertical mobility across several dimensions, including gender, age, socioeconomic status, etc.

APPENDIX

A. Localization & Interpolation

The location data is using an API from a third party company called Skyhook based on the available Wi-Fi APs. It requires a minimum number of APs to triangulate a given location. As the density of Wi-Fi APs varies from place to place, accurate localization may not always be achievable with insufficient data. This is especially pronounced in transportation modes—such as underground subway rides, by-pass roads, tunnels, bridges, etc.—where Wi-Fi APs are sparse or even non-existent, even in highly dense cities. In our dataset, a per-person average of 8% data points are missing location information due to failed localization.

To complete the Wi-Fi localization data, an interpolation method is used to predict the missing values. To identify and validate the best interpolation method, successfully localized time series are selected and a fraction of its data is removed randomly. The Root Mean Squared Error (RMSE) value between original and randomly removed data is calculated for three distinct interpolation methods: (1) linear (2) cubic spline, and (3) piece-wise cubic spline. For a fraction ($10 \sim 50\%$) of the data removed, the linear interpolation consistently outperforms other methods under consideration with the lowest RMSE values. For fraction$> 20\%$ of the data removed, RMSE measured in degree of latitude/longitude is ~ 0.0001 for linear interpolation while RMSE (degree)~ 0.005 for other methods.

B. Regression

The Wi-Fi localization data has a low spatial resolution of ± 400 meters. The travel velocity estimation based on this data reveals high local errors. To reduce local fluctuations, a regression model is applied to find the best fit. To analyze the model performance, RMSE is used to estimate the residuals and the R-squared value quantifies the proportion of variance explained by the model. The regression model performs poorly when applied to the entire time series sequence. This is due to the different regions of variability in data. The time series location data is hence segmented into a series of local (temporary) variable and global (long-term) variable sequences. The regression model is then applied with different window sizes or knot placements to each of these sequences, higher number of knots for a global variable and lower number of knots for a local variable.

A local variable sequence is defined as $S_{i,n} = \{i, i+1, \ldots, n\}$ where the location of index $- n$ is at distance $\leq$ distance_cut-off from index $- i$. A global variable sequence is created by combining several local variable sequences of length < 10 (~ 2.5 minutes) to reflect long term changes in location.

Two regression models: (1) piece-wise polynomial and (2) natural piece-wise cubic regression spline are considered as well as three smoothing models: (3) Savitzky-Golay smoothing, (4) LOESS model, (5) Exponential smoothing model. These five options are compared against different window sizes or knot placements (local variable $= [5, 15, 30]$, global variable$= [5, 15, 30]$) and distance_cut_off = $[0.1, 0.3, 0.5]$ km). The natural piece-wise Cubic regression spline and Savitzky-Golay show the lowest RMSE and highest R-Squared values for all knot placements/window sizes and distance_cut_off. Either of these methods will suitably reduce the local errors in location data. A low RMSE and high R-squared value can also point to an over-fitted data, hence, model selection should be supported by additional considerations. The optimal knot placements should be sparse for local variable data to reduce local fluctuations and denser for global variable data where location varies long-term. Here, Natural Piece-wise cubic regression spline method is selected with knot placements at (local variable, global variable)$= (15, 1)$ of the segmented data using distance_cut_off $= 0.3$ km. Essentially, the knot placements are designed to smooth the local variable data while leaving the global variable sequences intact.

C. Classification model parameters

1) XGBoost: Gradient boosting allows an ensemble of weak learners to build models that depend on the gradient descent algorithm to optimize an objective function. The objective function measures the model fitness of the training data

and consists of training loss and regularization [60]. Training loss quantifies the predictive capability of the model and the regularization parameters helps reduce over-fitting by controlling the complexity of the model. The balance between these two terms is commonly known as the bias-variance trade-off [60]. XGBoost classifier model encompass several parameters that are categorized as general, booster, and learning task parameters [61]. The tree booster parameters allows the specification of learning rate (eta), minimum loss reduction to make a split ($gamma$), maximum depth of a tree (max_depth), minimum sum of weights of all observations in a child (min_child_weight), fraction of observations to be sampled randomly ($subsample$), fraction of columns to be sampled randomly in each tree ($colsample_bytree$), control of class imbalance ($scale_pos_weight$), L2 regularization term on weights (reg_alpha), and L1 regularization term on weights (reg_lambda) [60], [61]. Learning task parameters gives the option to choose an objective function ($objective$) for optimization against a specified evaluation metric ($eval_metric$). The number of trees or estimators ($n_estimators$) required depends on the learning rate eta. The optimization of the tree booster parameters based on the evaluation metric allows the objective function to look for values that avoids over-fitting. Model over-fitting can be further controlled by reducing the learning rate and increasing the number of estimators [61].

2) Random Forest: The Random Forest classifier is an ensemble learning algorithm that tries to create a range of uncorrelated trees by randomly selecting features for each tree and randomly (with replacement) assigning training data to each tree. Both of these properties of Random Forest helps control over-fitting [62], [63]. The number of trees are given by the parameter $n_estimators$ and the maximum number of features considered for splitting a node is given by $max_features$. The maximum number of levels in each tree is controlled by the parameter max_depth. The samples given to the tree are controlled by the following parameters: $bootstrap$, min_sample_split, $min_weight_fraction_leaf$, and min_sample_leaf. The parameter $bootstrap$ can be set to $True$ to allow random sampling with replacement. The minimum number of data points assigned to a node before splitting is given by min_sample_split and the parameter min_sample_leaf is used to control the minimum number of data points allowed in a leaf node. The sample weight can be adjusted using the parameter $min_weight_fraction_leaf$. Finally, the class imbalance in a data set can be controlled by setting the parameter $class_weight$ to $balanced$.

3) Naive-Bayes: Based on Bayes' theorem, the Naive-Bayes model is a probabilistic learning algorithm. This study uses the Gaussian Naive-Bayes algorithm that assumes Gaussian distribution for each class [63], [64]. Other Naive-Bayes models such as Multinomial and Bernoulli are defined for discrete data values and hence not suitable for our problem [63]. Gaussian Naive-Bayes model has only two parameters: $var_smoothing$ and $priors$. The prior probability of the classes can be assigned through $priors$. Since there are no prior probability available for our classes, this parameter is unspecified in this work so that the priors can be learnt from the data. The parameter $var_smoothing$ is used to adjust the weight given to data points far from the mean distribution. This is the only parameter tuned for the Naive-Bayes model here.

4) k-Nearest Neighbors: k-Nearest Neighbors classifier is a class of Nearest Neighbors algorithm that identifies k-nearest training data points based on their similarity [63]. The number of neighbors k is given by the parameter $n_neighbors$. The neighbors can be weighed uniformly or differently using the parameter $weight$. The distance metric used can be controlled by the parameter p, which denotes the power of the Minkowski distance ($p = 1$ denotes Euclidean and $p = 2$ denotes Manhattan). The speed of finding the nearest neighbors depends on the parameter $algorithm$ that can be set to brute-force or more faster methods such as tree-based search algorithms. The tree-based search can be controlled by the parameter $leaf_size$ that can be adjusted for faster construction and queries.

D. Classifier model tuning

The four classifier models follow an overall procedural structure for model tuning. The optimal model hyperparameters are selected by grid search and a 5-fold cross-validation using the metric – Area Under the Receiver Operating Characteristic Curve (ROC AUC). The grid search is initialized with a set of hyperparameter values, which are fine tuned until no further change in the results is noticed. The 5-fold cross-validation is used to reduce the bias and makes use of the training set efficiently to understand the model's predictive power on new data during the tuning procedure. Due to the specific nature of the classifier models, some differences in the procedure exist. For example, all the hyperparameters for the Random Forest, Naive-Bayes, and k-Nearest Neighbors are tuned using a single grid search while the hyperparameters for XGBoost are tuned in a series of grid searches. The XGBoost model is thus tuned by the following steps:

XGBoost model tuning steps: The optimal parameter values are calculated in a series of steps, where each step progresses by estimating the parameter under consideration based on the parameters calculated from preceding step. The steps taken to tune the model are as follows: (1) Set a high learning rate and find the optimal number of estimators, (2) For the given learning rate and number of estimators, find optimal max_depth and min_child_weight, (3) Find optimal value of $gamma$ (4) Re-calibrate the optimal number of estimators, (5) find optimal values for the parameters $subsample$ and $colsample_bytree$, (6) find the optimal values for the regularization parameters reg_alpha and reg_lambda, (7) Re-calibrate the optimal number of estimators, (8) Reduce the learning rate and find the optimal number of estimators [61].

The cross-entropy loss for binary classification is set as the objective function. For an initial learning rate of 0.3, the above steps are followed from (1) to (7). As the final step (8), the learning rate is reduced considerably (from 0.3 to 0.2 for feature set-I, to 0.1 for both feature set-II, and feature set-III) to control over-fitting while keeping the F_1 score of VDA classification higher. The optimal number of estimators is then found to be 43, 125, and 72 for feature set-I, II, and III respectively from 5-fold cross validation for the given learning rate. The class imbalance is acknowledged by setting the parameter $scale_pos_weight$ to 1 for faster convergence.

E. Pressure-altitude relation

The fundamental equation for fluids at rest dictates the relationship between atmospheric pressure and altitude [65]. The change in altitude is indeed given by

$$z_2 - z_1 = -\frac{p_2 - p_1}{\gamma}, \tag{2}$$

where $\gamma = \rho g$ is the specific weight of air with density $\rho = 1.225$ kg/m^3 and g is the acceleration due to gravity $g = 9.81$ m/s^2 at standard sea-level conditions. The fluid is assumed to be incompressible and in isothermal condition. This pressure-altitude equation is valid for data collected for elevation less than 10 km from sea-level [65], which is always the case in our study.

ACKNOWLEDGMENT

The NSE 2016 was supported by the Singapore National Research Foundation (NRF), Ministry of Education (MOE), and the SUTD-MIT International Design Centre. On-time production of the sensors was only possible due to strong support from Delta Electronics DRC, IABG, and Taoyuan Factory. Logistics for the distribution of thousands of sensors was facilitated by the Singapore Science Center, and the infrastructure to securely store the data was built and maintained by the A*STAR Institute for High Performance Computing.

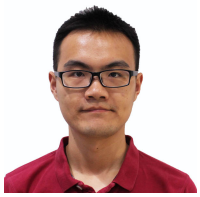
We also would like to thank Francisco Benita, Francesco Scandola, and Garvit Bansal for their assistance with SENSg devices and guidance on data collection. A.M. was supported by an MOE-SUTD PhD fellowship and currently supported by University of Ottawa PhD Admission scholarship.

Ajaykumar Manivannan received the B.Eng. (2013) in Aerospace Engineering from Hindustan University, India and M.Sc. (2016) in Aerospace Engineering from Technische Universität München (TUM), Germany and Nanyang Technological University (NTU), Singapore. He is currently pursuing Ph.D. with the Mechanical Engineering department, University of Ottawa, Canada. His research interest includes computational social science, human mobility, network science, and fluid mechanics.

Elias J. Willemse received the B.Eng. and Ph.D. degrees in Industrial Engineering from the University of Pretoria, South Africa, in 2007 and 2016, respectively. He was a senior lecturer at the University of Pretoria from 2016 to 2019, and he was a Postdoctoral Researcher with the Singapore University of Technology and Design from 2018 to 2020. He is currently the CTO of Waste Labs. His research interests include optimisation modelling, and their application to transportation and waste collection systems.

Balamurali B. T. received his Ph.D. in Electrical and Computer Engineering from the university of Auckland, New Zealand in 2015. He is currently a postdoctoral research fellow working at the Singapore University of Technology and Design. After his Ph.D, he worked as a researcher in gastro intestinal group in Auckland Bio-engineering Institute and also served as a lecturer in Auckland university of technology. Prior to his Ph.D endeavor, he was a senior design and development engineer in Tata Elxsi, India. He is passionate about artificial intelligence and trying to solve a variety of problems related to bio-acoustics processing, detection and classification of bio-signals, automatic speech/speaker recognition, spoofed-speech detection, blacklisted speaker identification, blind source separation, music classification, fluid flow classification, early-dementia prediction, etc

Wei Chien Benny Chin received the B.Sc., M.Sc., and Ph.D. degrees in geography from National Taiwan University, in 2011, 2014, and 2018, respectively. He is currently a Postdoctoral Researcher with the Singapore University of Technology and Design. His research interest includes space-time analysis and complex network analysis. He is also participating in projects associated with complex human movement and the horizontal and vertical dimensions of spatial structures.



Yuren Zhou received the B.Eng. degree in electrical engineering from Harbin Institute of Technology, Harbin, China, in 2014, and the Ph.D. degree in engineering from Singapore University of Technology and Design, Singapore, in 2019. His research interests include data mining, machine learning, and their applications in human mobility, building energy management, and Internet of Things.

Bige Tunçer received her Ph.D. (2009) from Delft University of Technology, her M.Sc. from Carnegie Mellon University (1996), and her B.Arch (1993) from Middle East Technical University. She is now an associate professor at Singapore University of Technology and Design, where she leads the Informed Design Lab. The lab's research focuses on data driven architectural and urban design. She leads and participates in large multi-disciplinary research projects in evidence informed design, IoT, and big data.

Alain Barrat received the B.Sc. and M.Sc. degrees in physics from École Normale Supérieure (Paris, France) in 1992 and 1994, respectively. He received the Ph.D. degree in theoretical physics from UPMC university, Paris, France in 1996. He was a postdoctoral fellow at the Abdus Salam ICTP in Trieste, Italy, and entered the French National Council for Scientific Research (CNRS) as junior researcher in 1998. He is currently CNRS senior researcher at the Centre de Physique Théorique in Marseille, France. He is also Specially Appointed Professor at the Tokyo Tech World Research Hub Initiative (Tokyo, Japan) since April 2019. His research interests span statistical physics and its interdisciplinary applications. He is an expert in the field of complex networks, from fundamental aspects to applications ranging from computational social science to epidemiology.

Roland Bouffanais (Member, IEEE) received the B.Sc. and M.Sc. degrees in physics from École Normale Supérieure (ENS Lyon), Lyon, France, in 1997 and 1999, respectively. He also received the M.Sc. degree in physics from UPMC Paris Sorbonne University, Paris, France, in 1999. He received the Ph.D. degree in engineering from École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland, in 2007. He was a Postdoctoral Fellow and an Associate with the Department of Mechanical Engineering, Massachusetts Institute of Technology (MIT), Cambridge, MA, USA. He is now an Associate Professor with the Department of Mechanical Engineering at the University of Ottawa, Canada. His interdisciplinary research spans a vast breadth of areas and is focused on the design and control of decentralized complex systems, multiagent systems, leader–follower consensus dynamics, and nonlinear dynamical systems.

References

[1] United Nations, *World's cities in 2018—data booklet (st/esa/ser.a/417)*, Available online: https://www.un.org/en/events/citiesday/assets/pdf/the_worlds_cities_in_2018_data_booklet.pdf, (Accessed 2021-03-29), 2 United Nations Plaza, NY 10017 USA, Dec. 2018.

[2] Technavio. (2020). "Elevator and escalator market by product, end-user, and geography - forecast and analysis 2020-2024." (Accessed 2021-01-24), [Online]. Available: https://www.technavio.com/report/elevator-and-escalator-market-industry-analysis.

[3] Y. Zhou, B. P. L. Lau, C. Yuen, B. Tunçer, and E. Wilhelm, "Understanding urban human mobility through crowdsensed data," *IEEE Communications Magazine*, vol. 56, no. 11, pp. 52–59, 2018.

[4] A. G. Yeh and B. Yuen, "Introduction: High-rise living in asian cities," in *High-rise living in Asian cities*, Springer, 2011, pp. 1–8.

[5] S. Gopalakrishnan, D. Wong, A. Manivannan, R. Bouffanais, and T. Schroepfer, "User-driven emergent patterns of space use in vertically integrated urban environments," in *Proceedings of the 21st International Conference on Architectural Research Centers Consortium (*ARCC *2021)*, Arizona, United States, 2021, pp. 215–222. [Online]. Available: http://www.arcc-arch.org/wp-content/uploads/2021/08/ARCC-2021-Proceedings_Digital-Version_Web_PP.pdf.

[6] R. Bouffanais and L. Sun Sun. (2019). "The rise of homo verticalis." (Accessed 2021-10-16), [Online]. Available: https://blogs.scientificamerican.com/observations/the-rise-of-homo-verticalis/.

[7] H. Barbosa, M. Barthelemy, G. Ghoshal, C. R. James, M. Lenormand, T. Louail, R. Menezes, J. J. Ramasco, F. Simini, and M. Tomasini, "Human mobility: Models and applications," *Physics Reports*, vol. 734, pp. 1–74, 2018.

[8] S. Hasan, C. M. Schneider, S. V. Ukkusuri, and M. C. González, "Spatiotemporal patterns of urban human mobility," *Journal of Statistical Physics*, vol. 151, no. 1, pp. 304–318, 2013.

[9] A. Manivannan, W. C. B. Chin, A. Barrat, and R. Bouffanais, "On the challenges and potential of using barometric sensors to track human activity," *Sensors*, vol. 20, no. 23, p. 6786, 2020.

[10] M. Straczkiewicz and J.-P. Onnela, "A systematic review of human activity recognition using smartphones," *arXiv preprint arXiv:1910.03970*, 2019.

[11] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1192–1209, 2013.

[12] A. Bulling, U. Blanke, and B. Schiele, "A tutorial on human activity recognition using body-worn inertial sensors," *ACM Comput. Surv.*, vol. 46, no. 3, 33:1–33:33, Jan. 2014, ISSN: 0360-0300. DOI: 10.1145/2499621.

[13] K. Sagawa, T. Ishihara, A. Ina, and H. Inooka, "Classification of human moving patterns using air pressure and acceleration," in *IECON'98. Proceedings of the 24th Annual Conference of the IEEE Industrial Electronics Society (Cat. No. 98CH36200)*, IEEE, vol. 2, 1998, pp. 1214–1219.

[14] M. Janidarmian, A. R. Fekr, K. Radecka, and Z. Zilic, "A comprehensive analysis on wearable acceleration sensors in human activity recognition," *Sensors (Switzerland)*, vol. 17, no. 3, 2017, ISSN: 14248220. DOI: 10.3390/s17030529.

[15] N. Twomey, T. Diethe, X. Fafoutis, A. Elsts, R. McConville, P. Flach, and I. Craddock, "A Comprehensive Study of Activity Recognition Using Accelerometers," *Informatics*, vol. 5, no. 2, p. 27, 2018. DOI: 10.3390/informatics5020027.

[16] Y. Chen and C. Shen, "Performance Analysis of Smartphone-Sensor Behavior for Human Activity Recognition," *IEEE Access*, vol. 5, pp. 3095–3110, 2017, ISSN: 21693536. DOI: 10.1109/ACCESS.2017.2676168.

[17] J. Mäntyjärvi, J. Himberg, and T. Seppänen, "Recognizing human motion with multiple acceleration sensors," *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, vol. 2, pp. 747–752, 2001, ISSN: 08843627. DOI: 10.1109/ICSMC.2001.973004.

[18] S. Chung, J. Lim, K. J. Noh, G. Kim, and H. Jeong, "Sensor data acquisition and multimodal sensor fusion for human activity recognition using deep learning," *Sensors (Switzerland)*, vol. 19, no. 7, 2019, ISSN: 14248220. DOI: 10.3390/s19071716.

[19] K. Kunze and P. Lukowicz, "Sensor placement variations in wearable activity recognition," *IEEE Pervasive Computing*, vol. 13, no. 4, pp. 32–41, 2014, ISSN: 15361268. DOI: 10.1109/MPRV.2014.73.

[20] M. Cornacchia, K. Ozcan, Y. Zheng, and S. Velipasalar, "A survey on activity detection and classification using wearable sensors," *IEEE Sensors Journal*, vol. 17, no. 2, pp. 386–403, 2017.

[21] Z. Yan, V. Subbaraju, D. Chakraborty, A. Misra, and K. Aberer, "Energy-efficient continuous activity recognition on mobile phones: An activity-adaptive approach," *Proceedings - International Symposium on Wearable Computers, ISWC*, pp. 17–24, 2012, ISSN: 15504816. DOI: 10.1109/ISWC.2012.23.

[22] L. Bao and S. S. Intille, "Activity Recognition from User-Annotated Acceleration Data BT - UbiComp 2002: Ubiquitous Computing," *UbiComp 2002: Ubiquitous Computing*, vol. 3001, no. Chapter 1, pp. 1–17, 2004.

[23] M. Liu, H. Li, Y. Wang, F. Li, and X. Chen, "Double-windows-based motion recognition in multi-floor buildings assisted by a built-in barometer," *Sensors*, vol. 18, no. 4, p. 1061, 2018.

[24] A. Moncada-Torres, K. Leuenberger, R. Gonzenbach, A. Luft, and R. Gassert, "Activity classification based on inertial and barometric pressure sensors at different anatomical locations," *Physiological measurement*, vol. 35, no. 7, p. 1245, 2014.

[25] A. El Halabi and H. Artail, "Integrating pressure and accelerometer sensing for improved activity recognition on smartphones," in *2013 Third International Conference on Communications and Information Technology (ICCIT)*, IEEE, 2013, pp. 121–125.

[26] K. Muralidharan, A. J. Khan, A. Misra, R. K. Balan, and S. Agarwal, "Barometric phone sensors: More hype than hope!" In *Proceedings of the 15th Workshop on Mobile Computing Systems and Applications*, ACM, 2014, p. 12.

[27] S. Vanini, F. Faraci, A. Ferrari, and S. Giordano, "Using barometric pressure data to recognize vertical displacement activities on smartphones," *Computer Communications*, vol. 87, pp. 37–48, 2016.

[28] S.-M. Lee, S. M. Yoon, and H. Cho, "Human activity recognition from accelerometer data using convolutional neural network," in *2017 ieee international conference on big data and smart computing (bigcomp)*, IEEE, 2017, pp. 131–134.

[29] I. Cleland, M. P. Donnelly, C. D. Nugent, J. Hallberg, M. Espinilla, and M. Garcia-Constantino, "Collection of a Diverse, Realistic and Annotated Dataset for Wearable Activity Recognition," *2018 IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops 2018*, pp. 555–560, 2018. DOI: 10.1109/PERCOMW.2018.8480322.

[30] C. Bollmeyer, T. Esemann, H. Gehring, and H. Hellbrück, "Precise indoor altitude estimation based on differential barometric sensing for wireless medical applications," in *2013 IEEE International Conference on Body Sensor Networks*, IEEE, 2013, pp. 1–6.

[31] G. Pipelidis, O. R. M. Rad, D. Iwaszczuk, C. Prehofer, and U. Hugentobler, "A novel approach for dynamic vertical indoor mapping through crowd-sourced smartphone sensor data," in *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, IEEE, 2017, pp. 1–8.

[32] A. Sabatini and V. Genovese, "A sensor fusion method for tracking vertical velocity and height based on inertial and barometric altimeter measurements," *Sensors*, vol. 14, no. 8, pp. 13 324–13 347, 2014.

[33] B. Ghimire, C. Nickel, and J. Seitz, "Pedestrian motion state classification using pressure sensors," in *2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, IEEE, 2016, pp. 1–6.

[34] F. Bianchi, S. J. Redmond, M. R. Narayanan, S. Cerutti, and N. H. Lovell, "Barometric pressure and triaxial accelerometry-based falls event detection," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 18, no. 6, pp. 619–627, 2010.

[35] M. Voleno, S. J. Redmond, S. Cerutti, and N. H. Lovell, "Energy expenditure estimation using triaxial accelerometry and barometric pressure measurement," in *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, IEEE, 2010, pp. 5185–5188.

[36] F. Massé, R. R. Gonzenbach, A. Arami, A. Paraschiv-Ionescu, A. R. Luft, and K. Aminian, "Improving activity recognition using a wearable barometric pressure sensor in mobility-impaired stroke patients," *Journal of neuroengineering and rehabilitation*, vol. 12, no. 1, p. 72, 2015.

[37] C. M. el Achkar, C. Lenoble-Hoskovec, A. Paraschiv-Ionescu, K. Major, C. Büla, and K. Aminian, "Instrumented shoes for activity classification in the elderly," *Gait & posture*, vol. 44, pp. 12–17, 2016.

[38] M. Elhoushi, J. Georgy, A. Wahdan, M. Korenberg, and A. Noureldin, "Using portable device sensors to recognize height changing modes of motion," in *2014 Ieee International Instrumentation and Measurement Technology Conference (I2mtc) Proceedings*, IEEE, 2014, pp. 477–481.

[39] C. Figueira, R. Matias, and H. Gamboa, "Body location independent activity monitoring.," in *BIOSIGNALS*, 2016, pp. 190–197.

[40] A. Ejupi, C. Galang, O. Aziz, E. J. Park, and S. Robinovitch, "Accuracy of a wavelet-based fall detection approach using an accelerometer and a barometric pressure sensor," in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, IEEE, 2017, pp. 2150–2153.

[41] K. Leuenberger, R. Gonzenbach, E. Wiedmer, A. Luft, and R. Gassert, "Classification of stair ascent and descent in stroke patients," in *2014 11th International Conference on Wearable and Implantable Body Sensor Networks Workshops*, IEEE, 2014, pp. 11–16.

[42] K. Sankaran, M. Zhu, X. F. Guo, A. L. Ananda, M. C. Chan, and L.-S. Peh, "Using mobile phone barometer for low-power transportation context detection," in *Proceedings of the 12th ACM Conference on Embedded Network Sensor Systems*, ACM, 2014, pp. 191–205.

[43] M. Monteiro and A. C. Martí, "Using smartphone pressure sensors to measure vertical velocities of elevators, stairways, and drones," *arXiv preprint arXiv:1607.00363*, 2016.

[44] M. Wu, P. H. Pathak, and P. Mohapatra, "Monitoring building door events using barometer sensor in smartphones," in *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, ACM, 2015, pp. 319–323.

[45] J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, "Deep learning for sensor-based activity recognition: A survey," *Pattern Recognition Letters*, vol. 119, pp. 3–11, 2019, Deep Learning for Pattern Recognition, ISSN: 0167-8655. DOI: https://doi.org/10.1016/j.patrec.2018.02.010. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S016786551830045X.

[46] H. F. Nweke, Y. W. Teh, M. A. Al-Garadi, and U. R. Alo, "Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: State of the art and research challenges," *Expert Systems with Applications*, vol. 105, pp. 233–261, 2018.

[47] F. Attal, S. Mohammed, M. Dedabrishvili, F. Chamroukhi, L. Oukhellou, and Y. Amirat, "Physical human activity recognition using wearable sensors," *Sensors*, vol. 15, no. 12, pp. 31 314–31 338, 2015.

[48] SUTD. (2016). "National science experiment, singapore." (Accessed 2021-03-29), [Online]. Available: https://www.family.sg/singapore-national-science-experiment.html.

[49] E. Wilhelm, S. Siby, Y. Zhou, X. J. S. Ashok, M. Jayasuriya, S. Foong, J. Kee, K. L. Wood, and N. O. Tippenhauer, "Wearable environmental sensors and infrastructure for mobile large-scale urban deployment," *IEEE Sensors Journal*, vol. 16, no. 22, pp. 8111–8123, 2016.

[50] Y. Zhou, J. Wang, P. Shi, D. Dahlmeier, N. Tippenhauer, and E. Wilhelm, "Power-saving transportation mode identification for large-scale applications," 2017. arXiv: 1701.05768.

[51] Skyhook. (2021). "Skyhook location services." (Accessed 2021-03-29), [Online]. Available: https://www.skyhook.com/.

[52] C. Lang and S. Kaiser, "Classifying elevators and escalators in 3d pedestrian indoor navigation using foot-mounted sensors," in *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, IEEE, 2018, pp. 1–7.

[53] K. S. Hamrick and K. J. Shelley, "How much time do americans spend preparing and eating food?" Tech. Rep., 2005.

[54] B. Monnot, F. Benita, and G. Piliouras, "How bad is selfish routing in practice?" *arXiv preprint arXiv:1703.01599*, 2017.

[55] A.-L. Barabási, *Bursts: the hidden patterns behind everything we do, from your e-mail to bloody crusades*. Penguin, 2010.

[56] M. C. González, C. A. Hidalgo, and A.-L. Barabási, "Understanding individual human mobility patterns," *Nature*, vol. 453, no. 7196, pp. 779–782, 2008.

[57] E. J. Willemse, B. Tunçer, and R. Bouffanais, "Identifying highly dense areas from raw location data," in *Proceedings of the 24th International Conference of the Association for Computer-Aided Architectural Design Research in Asia*, CAADRIA, vol. 2, 2019, pp. 805–814.

[58] F. Haque, V. Dehghanian, A. O. Fapojuwo, and J. Nielsen, "A sensor fusion-based framework for floor localization," *IEEE Sensors Journal*, vol. 19, no. 2, pp. 623–631, 2018.

[59] X. Shen, Y. Chen, J. Zhang, L. Wang, G. Dai, and T. He, "Barfi: Barometer-aided wi-fi floor localization using crowdsourcing," in *2015 IEEE 12th International Conference on Mobile Ad Hoc and Sensor Systems*, IEEE, 2015, pp. 416–424.

[60] xgboost. (2020). "Xgboost documentation." (Accessed 2021-03-29), [Online]. Available: https://xgboost.readthedocs.io/.

[61] A. Jain. (2020). "Xgboost parameter tuning guide to parameter tuning in xgboost." (Accessed 2021-03-29), [Online]. Available: https://www.analyticsvidhya.com/blog/2016/03/complete-guide-parameter-tuning-xgboost-with-codes-python/.

[62] T. Yiu. (2019). "Understanding random forest." (Accessed 2021-10-16), [Online]. Available: https://towardsdatascience.com/understanding-random-forest-58381e0602d2.

[63] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[64] A. Sharma. (2021). "Gaussian naive bayes with hyperparameter tuning." (Accessed 2021-10-16), [Online]. Available: https://www.analyticsvidhya.com/blog/2021/01/gaussian-naive-bayes-with-hyperpameter-tuning/.

[65] D. F. Young, B. R. Munson, T. H. Okiishi, and W. W. Huebsch, *A Brief Introduction to Fluid Mechanics*. John Wiley & Sons, 2010.