

VehicleSense: A Reliable Sound-based Transportation Mode Recognition System for Smartphones

Sungyong Lee
ECE, UNIST
Ulsan, South Korea
sungyonglee@unist.ac.kr

Jinsung Lee
Samsung Electronics
Suwon, South Korea
js81.lee@samsung.com

Kyunghan Lee
ECE, UNIST
Ulsan, South Korea
khlee@unist.ac.kr

Abstract—A new transportation mode recognition system for smartphones, VehicleSense that is widely applicable to mobile context-aware services is proposed. VehicleSense aims at achieving three performance objectives: high accuracy, low latency, and low power consumption at once by exploiting sound characteristics captured from the built-in microphone while being on candidate transportations. To attain high energy efficiency, VehicleSense adopts hierarchical accelerometer-based triggers that minimize the activation of the microphone of smartphones. Further, to attain high accuracy and low latency, VehicleSense makes use of non-linear filters that can best extract the transportation sound samples. Our 186-hour log of sound and accelerometer data collected by seven different Android smartphone models confirms that VehicleSense achieves the recognition accuracy of 98.2% with only 0.5 seconds of sound sampling at the power consumption of 26.1 mW on average for all day monitoring.

I. INTRODUCTION

We live in the world of smartphones where people interact with their smartphones for a significant time of a day, e.g., [1]. As people spend more time on their smartphones, they expect higher intelligence from the smartphones. In order to meet this expectation, researchers have come up with the concept of recognition systems that bring intelligence to smartphones, with which the surrounding contexts are comprehended to provide timely services to the smartphone users. There exist diverse recognition systems such as activity recognition [2], exercises recognition [3], transportation mode recognition [4], and touch-based identity recognition systems [5].

Among them, we give our special attention to the transportation mode recognition system that aims to figure out when a user gets on and off a vehicle and to further figure out what type of the vehicle it is. There are various applications that benefit from this information. One technical application is found in the architecture design of 5G core networks driven by 3GPP [6]. In 5G networks, cellular network operators are eager to optimize their signaling and data path to user devices by precisely and instantly predicting the mobility patterns of their subscribers (e.g., by discovering the transportation modes that the subscribers use) for the provision of next generation services such as ultra-low latency. Another example is found in a real life situation of the parents of a kindergartener. The parents can set up a transportation mode recognition service at the smartphone given to their child, which immediately gives

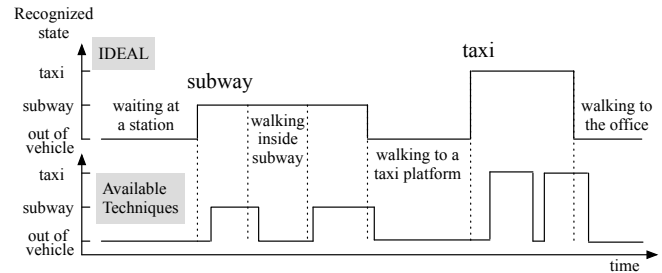


Fig. 1: A sample output from an ideal transportation mode recognition system is compared with that from available systems that mostly have performance issues in recognition accuracy and latency.

an alarm to the parents when the child gets on an unexpected vehicle (e.g., a car or a subway) at an unexpected moment. This can potentially prevent the child from being kidnapped.

Through a series of studies, there exist a number of transportation mode recognition systems such as [7], [8], [9]. However, those are not viable to run the aforementioned services because their recognition performance is not fully reliable in the wild. For instance, suppose there is a usual commute scenario where a person first takes a subway and changes to a taxi, and then walks to the office as in Fig. 1. The figure illustrates the outputs from both an ideal recognition system and available systems for the given scenario, which clarify three major performance issues of available techniques. The first issue is delayed recognition. Whenever the person makes a ride or ends a ride, most available systems experience non-negligible amount of recognition delay that easily spans to a few minutes. Such a long delay can cause critical misbehavior of services that rely on the transportation mode recognition system, such as unnecessary repeated scans to reform the tethering group although the group is no longer available for the person who got off a vehicle. The second issue is inaccurate recognition. As available systems show their best accuracy at around 90%, misjudgement happens frequently as illustrated in the taxi ride of Fig. 1. If the recognition system misjudges, the services on top get confused. Confusions always lower the service quality but some confusions can be even more critical such as a false data charge. The third is incomplete recognition. As depicted in the subway ride of Fig. 1, if the person makes a move inside a subway, e.g., to find a better seat, most available systems conclude that the person is getting off the subway. If the person is again determined to be inside a subway after that

happening, the person is considered to take another ride. This type of misjudgement comes from the incomplete design of recognition systems that fail to cover all possibilities. This can cause severe performance drop at some services such as the wireless channel prediction.

For a reliable transportation mode recognition system that overcomes all those performance issues, we propose VehicleSense, a sound-based transportation mode recognition system which is accurate, immediate, and low-powered. VehicleSense makes a recognition decision based on its learning about the characteristic spectral sound features of popular transportation modes involving subways, buses, and taxis. Our system takes hierarchical accelerometer based triggers as well as sound pre-processing modules into consideration, which leads to highly reliable performance of VehicleSense. Our 186-hour log of sound and accelerometer data collected by seven different Android smartphone models confirms that VehicleSense achieves the recognition accuracy of 98% with only 0.5 seconds of sound sampling, while consuming only about 26.1 mW on average for all day monitoring. It also deals with the privacy concern that may arise from using the microphone by suggesting a privacy protection mode that completely precludes VehicleSense from extracting a single word from the sampled sound.

II. RELATED WORK

In accordance with our work, we brief previous transportation mode recognition systems classified by the sensors they mainly use as follows:

A. Accelerometer only

Because of its relatively low power consumption compared to other sensors, 3-axis accelerometer has been the most popular sensor for the activity recognition. Hemminki et al. [7] tackled important issue in accelerometer based recognition, which is so-called gravity factor elimination. In a nutshell, it is the problem of subtracting the gravity value (i.e., 9.8 m/s^2) which is dynamically distributed to axes while moving. They proposed and implemented the concept of peaks of accelerometers characterizing acceleration and deceleration patterns of motorized transportation and achieved about 85% of accuracy in distinguishing transportation modes of bus, car, subway, train and tram. But the computational complexity involved in the proposed gravity factor elimination is demonstrated to increase the power consumption as high as 85 mW. Shafique et al. [10] recently proposed another accelerometer only system that distinguishes walk, bicycle, car, and train. By using a classifier, random forest search over various accelerometer features, they insisted that achieving over 95% recognition accuracy is possible. However, because they have not given their focus on distinguishing motorized vehicles in detail and the samples for vehicles are relatively insufficient, it remains unconfirmed if its performance is reliably achievable in practice.

B. Accelerometer and GPS

Shah et al. [11] with the GPS support constructs a database of transit route information from the data published by various transportation agencies and detects the type of vehicle in

use by performing a geographical pattern matching. [11] is shown to obtain about 90% of accuracy in distinguishing route-determined public transportations such as bus and subway, but accompanies severe battery drain from continuous location information acquisition.

C. Accelerometer and Barometer

Unlike others, [8] paid attention to a low-powered transportation mode detection using an embedded barometer in combination with the accelerometers. It is shown to consume 32 mW less compared to an accelerometer-only approach in detecting idle, walking, or being in a vehicle. However, the answers on how to continuously calibrate the barometer and how to improve the detection accuracy which is reduced down to 81% by the impreciseness of the embedded barometer are left as future work.

D. Accelerometer and Microphone

There exists a recent study [12] that exploits the microphone to detect when a person gets on and off a bus by letting a smartphone listen to the beeping sound from IC card readers of buses. However, because the original goal of the system is to improve the prediction quality on the bus arrival time from crowd-sourcing, its generalization to other transportations has not been considered. A more relevant work [9] utilizes the microphone of a smartphone along with the assistance of the accelerometers to recognize the type of the vehicle in use. The proposed system shows relatively good accuracy, 92%, but it is incapable of detecting when the user gets off from a vehicle. Also, distinguishing whether a movement is made inside or outside a vehicle is out of its scope.

III. MOTIVATION

In order for various context-aware services to best utilize the transportation mode recognition system, the system should satisfy the followings: 1) *high accuracy*, 2) *instant recognition*, and 3) *low power consumption*. In each of these three perspectives, we explain why the existing systems have difficulties in meeting the requirements.

A. Recognition Accuracy

Most activity recognition system use 3-axis accelerometers to interpret the sensor reading as the owner's activity. However, in transportation recognition, using accelerometers is prone to make mispredictions because the reading is affected by the mixture of human and vehicle motions. Thus, recognizing the vehicle is challenging when a person is walking or budging compared to sitting or standing still on a vehicle. In practical systems, this causes a recognition accuracy drop as people frequently make movements inside a vehicle. A quick detour to this accuracy drop problem is to monitor the situation with accelerometers for a longer duration and make a delayed decision. However, this detour leads to wrong decisions when the duration of a ride is very short or when a person makes continuous movement. More fundamentally, it gets always confused when the smartphone is in use, so it gets persistent wiggling from handshaking and touch actions. If the

accelerometers are assisted by a barometer which continuously tells the altitude variation of a mobile device, the accuracy may be improved but not much because of the frequent confusions caused by the geographical structure of a city (e.g., ups and downs) and the weather condition. Therefore, it is non-trivial to design a system that achieves over 90% of accuracy with the accelerometer and the barometer [8]. GPS can largely assist those sensors conceptually but its weak reception in downtown areas and no reception in underground makes its practicality restricted.

B. Recognition Latency

Recognition latency is a key performance metric of a recognition system especially designed for applications that require timely services. If we define a timely recognition as a decision made within 3 seconds from the moment at which some action happened (e.g., getting on/off a vehicle), there is no existing system that satisfies this tight requirement. The typical latencies of accelerometer-based recognition systems range from 100 to 200 seconds [13], [7]. There had been several proposals that use only a few seconds of accelerometer sampling, however they turned out to suffer from poor accuracy, and thus are no longer popular. If a recognition system is designed to detect abrupt deceleration and acceleration from breaking and driving [7], its recognition latency could reach as high as several minutes. GPS-based recognition also needs at least several tens of seconds and sometimes needs much longer given that a vehicle makes no move at the red light during one minute or two.

C. Energy Efficiency

Energy efficiency is another key performance metric for a recognition system as the system needs to run in the background all the time. Given this necessity of continuous sensing, GPS which is known to consume huge power is certainly a bad choice. Using the Monsoon power monitor [14], we measure that GPS in Samsung Galaxy S6 consumes about 317 mW. Thus, GPS can be regarded useful only when it is intermittently used to support decisions made by low power sensors such as the accelerometers and the barometer that consume only about 5.23 mW and 2.3 mW when sampled at 50 Hz in Samsung Galaxy S6. However, although the accelerometers and barometer are known as low-powered, recognition systems that perform continuous processing over sampled data tend to consume high power by keeping the mobile AP (application processor) awakened. This is unavoidable unless a trade-off is made between the power consumption and the decision making interval that affects both latency and accuracy.

As explained, most of the sensors used in the literature have their own sets of limitations. Therefore, instead of combining such sensors, we opt to exploit the built-in microphone readily available in every smartphone. Considering that a person is highly likely to recognize what the transportation mode is while closing her eyes in a vehicle before any acceleration or deceleration happens, the microphone can be the most informative sensor that achieves high recognition accuracy

along with nearly immediate recognition. Also, the power consumption of the microphone is manageable as it takes 174.51 mW in Samsung Galaxy S6, which is substantially lower than that of GPS and can be further optimized by the help of accelerometers. In the following sections, we explain in detail how and why our proposed system relying on the microphone can achieve high accuracy, low latency, and low power consumption all at once.

IV. CHARACTERISTICS OF TRANSPORTATION SOUND

A. Frequency-Domain Characteristics

How can a person sense if she is in either of bus, taxi, or subway only by the sound? What are the characteristics of the unique sound of each transportation? We try to answer these questions by performing a power spectral density analysis [15]. As will be explained later in Section VII, we collected the empirical sound data of 186 hours captured from bus, taxi, and subway in South Korea and this data is used for the analysis. Fig. 2 shows the power spectral density of the sound captured from a bus, a taxi and a subway over time. Fig. 3 compares the power spectral density over frequencies of an announcement made at a bus and the sound samples from transportations. As shown in Figs. 2 and 3, each transportation holds its own distinctive spectral pattern, which is consistent irrespective of whether the transportation is standing or moving.

As shown in Figs. 2 (a) and 3, a bus has high spectrum density in the range of 250~400 Hz while moving. Whenever it stops at a red light or at a bus station, the dispersion of spectrum density over frequency domains narrows down and gets its peak at around 300 Hz. Our experimental sound recording from various spots of a bus reveals that this characteristic in low frequency sound comes from the engine of the bus. In South Korea, almost all city buses are equipped with a variety of 11,000cc (11 liter) CNG (compressed natural gas) engines with about 110 $kg \cdot m$ torque from several manufacturers. Most taxis in South Korea are with LPG (liquefied petroleum gas) based engines and produce relatively low noise. As expected, we observe spectrum density dispersed over a wide frequency range with minor peaks as shown in Figs. 2 (b) and 3. Low frequency components at 200~ 300 Hz are higher than other components, but they are modest. According to [16], the subway trains powered by distributed electric motors are deployed across more than 55 countries. The subway trains of this type that also run in South Korea produce unique electric noise patterns at about 100 Hz and 200 Hz as shown in Figs. 2 (c) and 3. When the subway train stops at a station, the pattern is still consistent with smaller magnitude.

B. Robustness of Sound Sensing

One concern about using sound information to distinguish transportation modes arises at its robustness over vocal noises such as conversations, radio broadcasting, and station announcement. It is known that human voices range widely from 20 to 5000 Hz, but are mostly concentrated at 100 ~ 200 Hz [17]. Fig. 3 confirms that the dispersion of vocal noises largely deviate from the patterns of transportations. Thanks to this

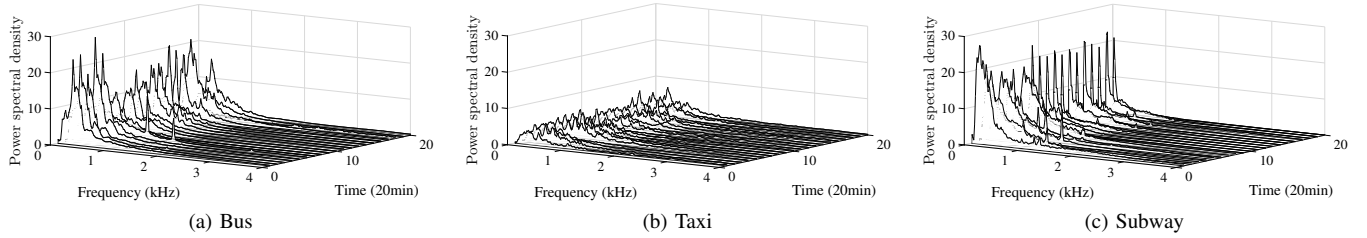


Fig. 2: The power spectral densities of the sound obtained from three different transportations. Each transportation has its unique spectral pattern and the pattern is persistent over time.

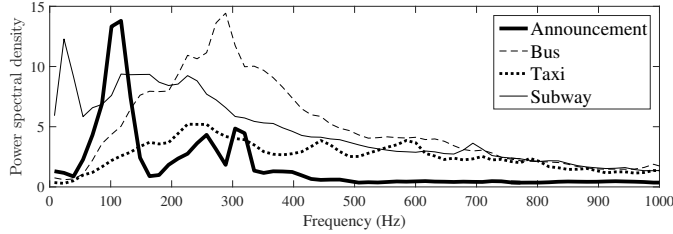


Fig. 3: The comparison of power spectral densities of an announcement made at a bus and the sample sound captured in a bus, a taxi, and a subway.

observation, we later confirm that our proposed system does not suffer from vocal noises in achieving high accuracy by its evaluation with 186 hours of sound data collected in the wild where various vocal noises prevail.

V. SYSTEM OVERVIEW

In order to achieve all three objectives in the motivation section at once, we design a new system, VehicleSense. We explain the details of the system design in two aspects: 1) low power architecture and 2) accuracy improvement.

A. Low Power Architecture

VehicleSense aims at minimally utilizing the microphone for energy efficiency, while not losing any small detail of the transitional behaviors related to transportation modes. In order to do so, we find the exact moments of capturing the sound. The moments for capturing the sound can be exemplified by the following scenarios: 1) A person is walking into a vehicle and then stopped, and 2) a person was determined to be inside a vehicle and now keeps walking to somewhere. The first example explains the situation where she is getting on a vehicle and then sits or stands at a place (e.g., in a bus or in a subway). The second example visualizes the situation where she starts to walk out from the vehicle or starts to relocate inside the vehicle. We ensure that these two cases are the only moments to capture the sound. This makes sense considering that it is of no use to continuously capture the sound while she keeps staying inside a vehicle or outside a vehicle. To realize this selective and timely sound capture, we design hierarchical triggers as shown in Fig. 4, which monitor the accelerometers and signal to the microphone whether or not to capture the sound. The hierarchical triggers have two levels whose first level trigger recognizes whether or not a person has strong kinetic energy representing that the person is walking/running or being stationary. The first level trigger works with a simple

thresholding method over the norm-2 magnitude from the multi-axis accelerometers. The threshold can be learned from activity data, for instance, our data logs and it is known by [18], [19] that the thresholding is precise enough to distinguish the kinetic states of a person. The computational complexity of the first trigger is minimal, thus it is even possible to run this trigger in any low power co-processors such as Qualcomm Hexagon 680 with a single core running at 1 GHz, which is equipped in Samsung Galaxy S7 series. When the first level trigger detects that the kinetic energy of a person has changed from high to low which we call by *Go&Stop*, the second level trigger kicks in and recognizes if the person is on a transportation or not, irrespective of what the transportation is. It is known by [7] that the accelerometers experience small perturbation while being at a transportation and this subtle but persistent fluctuation in the accelerometer readings can distinguish whether the person is being stationary on a vehicle or at a static place like an office or home. If it is indeed possible to detect if the person is on a vehicle by accelerometers, it is surely the right moment to turn on and capture the sound for the detailed recognition of what the type of the vehicle is. This is how VehicleSense mainly utilizes the hierarchical triggers. The only exception happens when the person was determined to be on a vehicle and is now walking. The reason why this can be of an exception is that although the second level trigger is not activated by the movement, VehicleSense should be continuously determining by the microphone whether the person is moving inside the vehicle or moving out from the vehicle. Thus, we let the first level trigger signal to the microphone upon detecting continuous walking over a certain time period (e.g., 5 seconds) and let the system make decisions repeatedly until the person becomes stationary again. Note that this exception doesn't incur high power consumption, which will be verified in Section VII. All these operational cases are depicted in Fig. 4.

B. Accuracy Improvement

In the design of VehicleSense, we choose to use the short-duration sound information whose length is less than 1 second. For maximizing the accuracy for a given sampling duration, we carefully design the pre-processing block, the feature extraction block, and the decision block of the sound classifier shown in Fig. 4. First, the decision block has been tested with various machine learning algorithms such as Naive Bayesian, Multi-dimensional Bayesian, HMM (hidden Markov model), and SVM (support vector machine). The SVM has been also tested

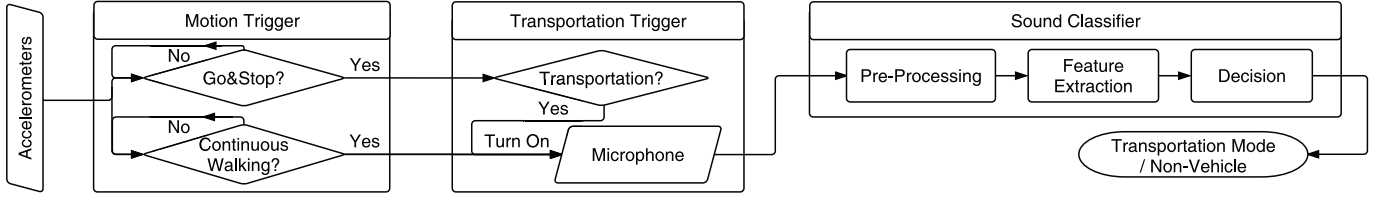


Fig. 4: The block diagram of VehicleSense that consists of two hierarchical accelerometer-based triggers and the sound classifier.

with a number of popular kernel functions such as RBF (radial basis function), polynomial, and sigmoid. Unlike [9] in which HMM was adopted for sound classification, we find that SVM with RBF kernel persistently outperforms available systems throughout extensive folding tests¹ over our sound logs. Based on this observation, we opt to use SVM for the decision block. It is known by [20] that once support vectors for the SVM is learned, every decision is made with little computation, which is nothing but the summation over matrix multiplications. Thus, using SVM can still satisfy the low power requirement. Second, the feature extraction block has been tested with diverse sets of popular sound features such as signal power, variance, peak interval, spectral centroid, spectral flatness and so on. We have also tested the interaction between the feature extraction block and the pre-processing block by applying various pre-processing techniques such as LPF (low-pass filter), BPF (band-pass filter), and *mel-scale* frequency division [21]. From our extensive combinatorial tests, we find that combining vectorized per-frequency features that are pre-processed by a specific form of non-linear pre-processing filter maximizes the accuracy even with the short-duration sound. VehicleSense is our crystallized system made up of these judiciously designed components.

VI. TRIGGERS AND FILTERS

In this section, we explain algorithmic details of triggers and sound filters that are trained offline and provide justifications on why such algorithms are adopted.

A. Motion Trigger

The first level trigger, **motion trigger**, is used to identify either of 1) the moments for the users changing their motions from walking to being stationary (i.e., sitting or standing), or 2) the moments for the users who have walked for longer than a preset duration of time (t_{cont}), for instance 5 seconds.

Algorithm 1 Motion trigger

```

1: while 1 do
2:    $Mag_a(t) := a_x(t)^2 + a_y(t)^2 + a_z(t)^2 - G^2$ 
3:   for every  $\Delta = 200\text{ms}$  do
4:      $Mag_a(t) := \text{average}(Mag_a(t))$  for 200ms
5:      $Mag_a^{MA}(t) = 0.5 \cdot Mag_a(t - \Delta) + 0.5 \cdot Mag_a(t)$ 
6:      $flag_w(t) = (Mag_a^{MA}(t) > T_w) ? 1 : 0$ 
7:      $flag_s(t) = (Mag_a^{MA}(t) < T_s) ? 1 : 0$ 
8:     if  $(flag_s(t) == 1) \&\& (flag_w(t - \Delta) == 1)$  then
9:       Go to Transportation Trigger
10:    else if  $flag_w(t) == 1$  remains for  $t_{cont}$  seconds then
11:      Go to Sound Classifier

```

¹A folding test means a test in which the learning period and the decision making period are separated in the data, and shuffled for more reliable evaluation.

B. Transportation Trigger

The transportation trigger examines if a person is on a vehicle at every moment when Go&Stop event is observed indicating that there exists a possibility that a person gets on a vehicle and finds a place to stay. This trigger is grounded on the fact that transportations have vibration patterns that are not observable at a fixed place. Such patterns are detected by SVM over three features extracted from accelerometer readings during a preset time window (t_{acc}): 1) accumulated magnitude (AM), 2) mean variance (MV), and 3) selected frequency components in the FFT (fast Fourier transform) (SF). We use t_{acc} as 2 seconds, which is empirically shown effective from our traces. Note that t_{acc} does not delay the recognition as the accelerometer readings are continuously made by the motion trigger and the readings stored can be used for the transportation trigger.

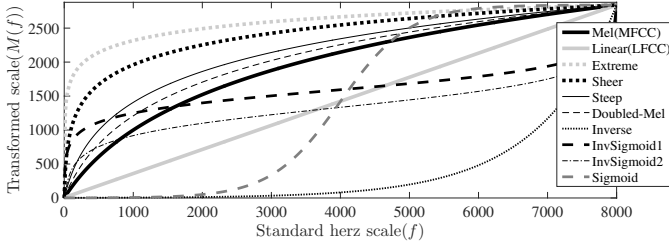
AM quantifies the extent of the accelerometer readings caused by large motions. This is done by accumulating the magnitude values of accelerometers, which are larger than a threshold denoted by T_{AM} , thus eliminating all minor movements during t_{acc} . By its design, AM can catch sudden motions such as standing up or sitting down. We use $96 (m/s^2)^2$ for the T_{AM} , which is also empirically chosen from our traces. MV measures the variation of accelerometer readings from the mean value during the time window, which is obtained by summing all per-axis variances over t_{acc} . SF that analyzes spectral characteristics of accelerometer readings understands regular vibrations dispersed over various frequency components. We let SF focus especially on the spectrum range from 1 to 20 Hz where the most vibration patterns of transportations are concentrated. We train these three features with SVM and let the SVM output one of the following states: being stationary at a fixed place (F), walking (W), being stationary in a vehicle (V). Thus, when V is provided, we invoke the sound classifier.

C. Sound Classifier

The sound classifier consists of three blocks: 1) pre-processing, 2) feature extraction, and 3) decision. We focus on explaining about our pre-processing technique as other blocks mostly involve standard SVM-based machine learning techniques. As soon as a trigger wakes up the microphone, the sound classifier samples a short duration sound data and start pre-processing. Our pre-processing aims at maximizing the differences between the sound characteristics of transportations in the feature space. Typically twisting the feature space to widen the distance between the data points to classify is done by a kernel function in the SVM, but this has a fundamental difference to the pre-processing. That is because once the vector

TABLE I: FILTERS USED AND THEIR SCALE FORMULAS.

Filter	Scale Formula	Accuracy(%)
Inverse	$20 \cdot \exp(f/2595)$	86.0
Mel	$2595 \cdot \log_{10}(1 + f/700)$	94.9
Linear	$0.3550f$	92.7
Extreme	$686 \cdot \log_{10}(1 + 10f)$	97.1
Double-Mel	$2141 \cdot \log_{10}(1 + f/350)$	98.3
Sheer	$1730 \cdot \log_{10}(1 + f/10)$	99.2
Steep	$1818 \cdot \log_{10}(1 + f/175)$	99.3
InvSigmoid	$-0.2855 \cdot \log_{10}(8000/f - 1) + 14$	97.8
InvSigmoid2	$-0.2855 \cdot \log_2(8000/f - 1) + 10$	96.3
Sigmoid	$1/(1 + \exp^{-0.002(f-4000)})$	84.4

Fig. 5: Illustration of the transformed scales ($M(f)$) from the filters in comparison over the standard hertz scale (f).

coordinates of the data points are quantized, the amount of information included in the coordinates is unchanged even with a coordinate transformation. However, the pre-processing can affect the amount of information included in the data coordinates. MFCC (mel frequency cepstral coefficient) [21] is a popular signal processing filter which is widely used in the speech recognition field. The main idea of MFCC is to transform the sound signal in the frequency domain from hertz (Hz) scale to mel scale to emphasize the frequency bands in which the most information of human voice is concentrated. Inspired by MFCC, we design our own scales with diverse non-linear filters that may best highlight the characteristic sounds from transportations. In Fig. 5, we depict our original filters, **Steep**, **Extreme**, **Inverse**, **Doubled-mel**, **Sheer**, **Sigmoid**, **InvSigmoid** and **InvSigmoid2** compared to existing filters, **Mel** (MFCC) and **Linear** (LFCC: linear frequency central coefficient) [22], whose scale formulas ($M(f)$) from the hertz scale (f) are in Table I. All these scales are designed to have the same start and end points and are linearly binned to be vectorized by the feature extraction block. As shown in Table I, the maximum recognition accuracy we obtain is 99.4% with **Steep** so we just take **Steep** as our default pre-processing filter. More detailed results are in Section VII.

VII. EVALUATION

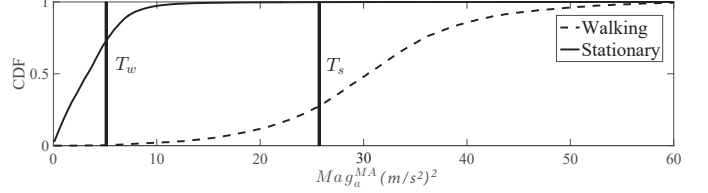
In this section, we verify the performance of VehicleSense with our own collected data in various perspectives. Then, we compare VehicleSense with an advanced accelerometer-based recognition system via trace-driven simulation and for the evaluation of the decision latency and the power consumption of two systems, we implement them on Samsung Galaxy S6 running Android 5.1.1 Lollipop. Also, we evaluate the performance of VehicleSense in a practical environment.

A. Data Collection

For our study, we have collected the microphone and the accelerometer data from seven different smartphone models whose total length is 183 hours, experimented by 8 testers. For

TABLE II: EXPERIMENT TIME (MINUTE) OF EACH TRANSPORTATION BY SMARTPHONE MODEL.

smartphone model	stationary	walking	bus	taxi	subway
LG Optimus G ProS	140	440	440	260	400
LG Vu3	120	420	300	400	400
LG G2	120	150	140	180	90
Samsung Galaxy S3	160	380	960	620	780
Samsung Galaxy Note2	160	380	460	340	320
Pantech Vega No.6	140	360	220	300	260
Pantech Vega LTE-A	160	300	180	220	280

Fig. 6: CDFs of the Mag_a^{MA} values from the accelerometers under stationary (either being inside or outside a vehicle) and walking motions. We also depicted two thresholds, T_w and T_s .

the microphone, we set the sampling rate as 16 kHz and the encoding method as 16 bit PCM (pulse-code modulation). For the accelerometers, we set the sampling rate as 50 Hz. While collecting the data, we have also acquired the activity labels provided by the testers at every moment of changing their motions. Each of the activity labels we have in the dataset is one of the following five motions: being stationary at a fixed place, walking, in a bus, in a taxi, and in a subway. Note that we minimized the labeling effort by the testers by asking them to keep each motion sustained at least for 20 minutes. Table II summarizes the per-device data length in minutes for the candidate activities. In addition to this dataset, we have collected another dataset from the wild, which only recorded the accelerometer readings from 110 randomly selected volunteers of various age bands, jobs, and cities recruited from an Internet community. The data collection lasted for 8 days and the microphone data was excluded for their privacy concern. We exploit this data only for the evaluation of triggers.

B. Performance of Triggers

Accuracy of triggers: We first evaluate the accuracy of the motion trigger. Fig. 6 shows the CDFs of the magnitude values extracted under two motions: stationary (either being inside or outside a vehicle) and walking. We set the stationary and walking thresholds to be $25.7(m/s^2)^2$ within and $5.1(m/s^2)^2$ with which 99.65% of walking and 99.95% of stationary motions are captured. These stationary and walking thresholds produce false positives of 25.5% and 26.8%, but this choice is made by our intention that aims at minimizing the false negatives in the hope of missing out virtually no transportation related events such as getting on and off. With this design, the missing rate of the riding events goes below 0.68%.

We then evaluate the performance of the transportation trigger to distinguish being stationary at a fixed place from being stationary at a transportation with our accelerometer dataset. The detailed classification accuracy values measured by three popular metrics [23], recall, precision, and F_1 score (the harmonic mean of precision and recall) are 89.03%, 99.37% and 93.91% respectively. The most important performance metric as a trigger is again the missing rate which is denoted

TABLE III: POWER CONSUMPTION OF THE MODULES.

	MT	TT	SC
Power consumption (mW)	5.80	178.30	270.92
Activated duration (s)	Persistent	1.02	0.54

by FNR (false negative rate). FNR is nothing but $1 - \text{recall}$. The result confirms that FNR is as low as 0.6%. Note that false positives do not incur critical problems as they only lead to a little more power consumption of the system.

Energy efficiency of triggers: We here evaluate how much the proposed triggers help in reducing power consumption of VehicleSense in comparison with continuous sensing methods. In order to understand the expected power consumption ($\mathbb{E}[\mathcal{P}]$) from VehicleSense for the realization of all day monitoring, we characterize it by estimating the probabilities of seeing specific events in a day. When we denote the instantaneous power consumption for the motion trigger (MT), the transportation trigger (TT), and the sound classifier (SC) by $\mathcal{P}_{MT}$, $\mathcal{P}_{TT}$, and $\mathcal{P}_{SC}$, the following equation holds:

$$\begin{aligned} \mathbb{E}[\mathcal{P}] = & \mathcal{P}_{MT} + p(W_{cont})\mathcal{P}_{SC}t_{SC} \\ & + p(W \rightarrow S)\{\mathcal{P}_{TT}t_{TT} + p(V|S)\mathcal{P}_{SC}t_{SC}\}, \end{aligned}$$

where $p(W \rightarrow S)$, $p(V|S)$, and $p(W_{cont})$ denote the probability of observing a transition from walking to being stationary, the probability of being on a vehicle given that a person is being stationary, and the probability of observing the motion of continuous walking for $t_{cont} = 5$ seconds. Note that all these probabilities are calculated from the discretized events assessed at every second throughout a day. Also note that t_{TT} and t_{SC} are the activated time duration in seconds, of the motion trigger and the sound classifier for each call. Notice that If we have no trigger system, the power consumption of VehicleSense simply becomes $\mathbb{E}[\mathcal{P}_{no-trigger}] = \mathcal{P}_{SC}$. If we only exploit the transportation trigger, $\mathbb{E}[\mathcal{P}_{1-trigger}] = \mathcal{P}_{TT} + p(V)\mathcal{P}_{SC}t_{SC}$ holds. Given the baseline statistics of MT, TT, and SC provided in Table III and the average event probabilities from our 110 user dataset, $p(W \rightarrow S) = 0.064$, $p(V|S) = 0.569$, and $p(W_{cont}) = 0.032$, the power consumption of VehicleSense becomes $\mathbb{E}[\mathcal{P}] = 26.11$ mW that outperforms $\mathbb{E}[\mathcal{P}_{no-trigger}] = 270.92$ mW and $\mathbb{E}[\mathcal{P}_{1-trigger}] = 183.62$ mW. Figs. 7 (a), (b), and (c) confirm that the $p(W \rightarrow S)$, $p(V|S)$, and $p(W_{cont})$ obtained from 110 users individually are different but have certain coherence. We can estimate the worst case power consumption from these statistics, which becomes $\mathbb{E}[\mathcal{P}] = 66.21$ mW. This is still far less than no-trigger and 1-trigger systems.

C. Performance of Sound Classifier

Recognition accuracy by non-linear filters: In Section VI, we have proposed non-linear filters that pre-process the sound data to maximize a difference between sound characteristics of vehicles. Fig. 8 summarizes the recognition accuracy from non-linear filters in three performance metrics, precision, recall, and F_1 score with 95% confidence intervals. As Fig. 8 clarifies, we can immediately notice that **Inverse** which is the only exponential shape filter shows the worst performance compared to others. Considering that the important frequency features are concentrated in the frequency range under 1 kHz, this result is of no surprise. More interesting observation is on the pattern

of the accuracy change over the logarithmic shape filters that become more convex in the order of **Linear**, **Mel**, **Doubled-Mel**, **Steep**, **Sheer**, and **Extreme**. Also Fig. 8 indicates that the accuracy gradually increases and peaks at **Steep** by showing 99.4% on average, and then again gradually decreases. The well-known pre-processing technique, **MFCC** shows relatively poor performance which is about 6% lower than that from **Steep**. The variants of a sigmoid function, **Sigmoid**, **InverseSig1**, and **InverseSig2** do not surpass the performance of **Steep** as well. Note that we let the sampling duration of the sound data be 0.5 seconds in obtaining the results shown in Fig. 8, but we confirmed that this trend remains the same for different sampling durations.

Recognition accuracy by sampling durations: As discussed in Section III, the recognition latency is one of the most important performance measures in the transportation mode recognition systems. In order to let VehicleSense conclude as quickly as possible, we investigate the trade-off between the sampling duration and the recognition accuracy, which is shown in Fig. 9 with 95% confidence intervals. The tested durations range from 0.1 seconds to 5 seconds. Fig. 9 shows that a longer sampling duration leads to higher accuracy, but at the same time the improvement in the accuracy is marginal beyond 0.3 seconds. A more practically valuable observation from Fig. 9 is that the accuracy is kept very high even at 0.3 seconds of sampling duration, which is much smaller than the sampling duration of 3 seconds suggested in the previous study [9]. By taking both the accuracy and the sampling duration into consideration, we opt to use 0.5 seconds as our default setting for the remaining experiments.

We have also tested VehicleSense with 0.5 seconds of sampling duration on 7 different smartphone models in Table II and confirmed that the average precision, recall, and F_1 score are stably over 98.5%. We omit the detailed values for brevity.

Privacy protection mode for VehicleSense: We briefly describe how we can mitigate the privacy concern from using the microphone. The key idea is to let the microphone alternatively sample sound and put a mute at each 0.1 seconds so that the sampled sound becomes no longer comprehensible. We call this *privacy protection (PP)* mode. For instance, if we use 0.5 seconds of sampling in *PP* mode, say 0.5 (*PP*), each sound sample has three 0.1 seconds of data and two 0.1 seconds of silence. We apply majority voting on three prediction results obtained from three data pieces. To our surprise, as shown in Fig. 9, it gives comparable performance with that of 0.5 seconds of continuous sampling. Note that we could not understand any contents from sound sample of 0.5 (*PP*).

D. Performance Comparison with Existing Systems

We now compare the system performance of VehicleSense with that of the one of the most advanced accelerometer based methods proposed in [7] in which peaks of accelerometers from acceleration and deceleration are carefully processed and characterized to recognize the type of a vehicle.

Recognition accuracy: Fig. 10 summarizes the recognition accuracy with 95% confidence intervals, which are obtained

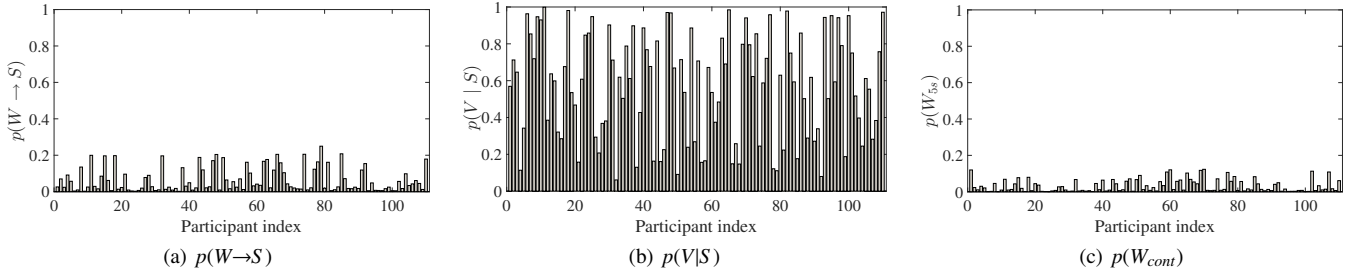


Fig. 7: $p(W \rightarrow S)$, $p(V|S)$, and $p(W_{cont})$ which are evaluated individually from the 110 user traces measured in the wild.

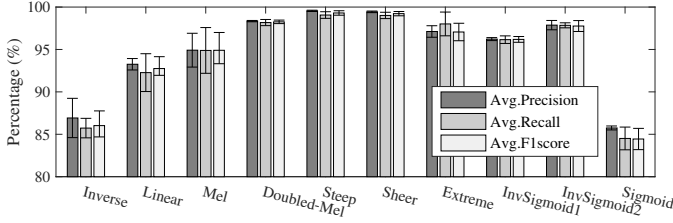


Fig. 8: Precision, recall, and F_1 score from the filters.

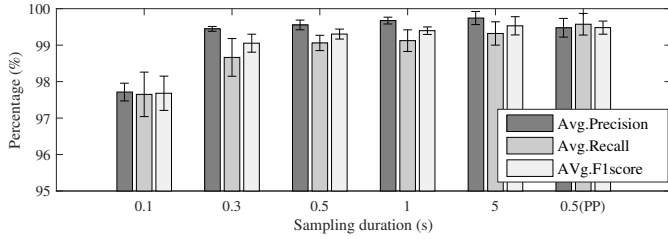


Fig. 9: Precision, recall, and F_1 score from different sampling durations. Privacy protection mode(PP) with 0.5 second sampling duration is also compared.

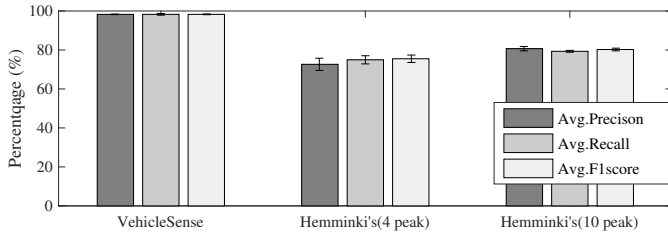


Fig. 10: Comparison of precision, recall, and F_1 score between VehicleSense and Hemminki's system of two different settings.

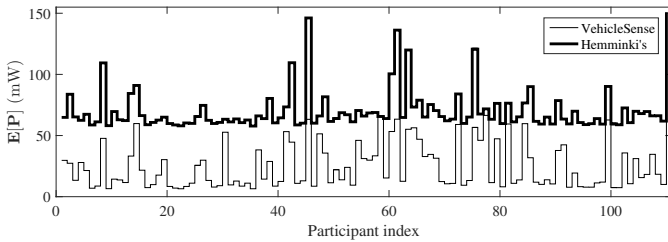


Fig. 11: $\mathbb{E}[P]$ of VehicleSense and Hemminki's, evaluated individually from the 110 user traces measured in the wild.

from VehicleSense and the Hemminki's system of two criteria, 4 and 10 peaks. For all cases, VehicleSense outperforms Hemminki's system. Specifically, the accuracy of VehicleSense reaches 98.2% while that of Hemminki's reaches up to 80.23%. But note that using more peaks leads to better accuracy but longer buffering of the accelerometer data, which in turn leads to longer latency.

Recognition latency: VehicleSense recognizes if a person is on a vehicle by going through the sound classifier followed by the two triggers. The Hemminki's system also has a trigger that senses the level of kinetic energy and invokes the main accelerometer processing module when the energy level is detected high. Given these hierarchical designs with different numbers of triggers, we measure the actual recognition latency from both systems running on the same Android platform. Our measurement on the recognition latency is done over simple scenarios of getting on and getting off a bus. Our measurement reveals that the latency of VehicleSense with 0.5 seconds of sound sampling duration keeps staying in the level of 0.6 seconds while that of [7] ranges from 80 to 140 seconds with 4 peaks. Note that both systems are activated right after the user becomes stationary in a vehicle. Thus, for VehicleSense to make a decision, about a second or up to a few seconds may be required due to the riding motion of a user.

Power consumption: Using the Android implementations of both systems, we measure the power consumptions of the triggers and the main classifiers implemented on Samsung Galaxy S6. Then, we apply these results to 110 user traces to quantify how much of sensing events will occur during a day, and thus to estimate the average power consumption of both systems. Fig. 11 summarizes the individual daily power consumption of both systems. On average, thanks to its very low energy consumption of the motion trigger, VehicleSense that consumes 26.11 mW is far more energy efficient than the Hemminki's system consuming 86.51 mW. This is due to the fact that VehicleSense uses extremely short sampling duration of sound while Hemminki's system requires monitoring of accelerometers for a much longer duration.

E. Real Experiment

In order to investigate the in-situ performance of VehicleSense in a real environment, we asked two users to carry the smartphones running VehicleSense for 48 hours in their real lives, during which they are expected to take transportations multiple times. By the difference in their life styles, one user took subways and buses while the other user took taxis during the experiment. Figs. 12 (a) and (b) illustrate the ground truth of their transportation uses along with the decisions made by VehicleSense over time. VehicleSense shows near-perfect decision performance during the whole experiment in terms of exactly when the users get on and off the vehicles and which vehicles they take except only one misprediction case found

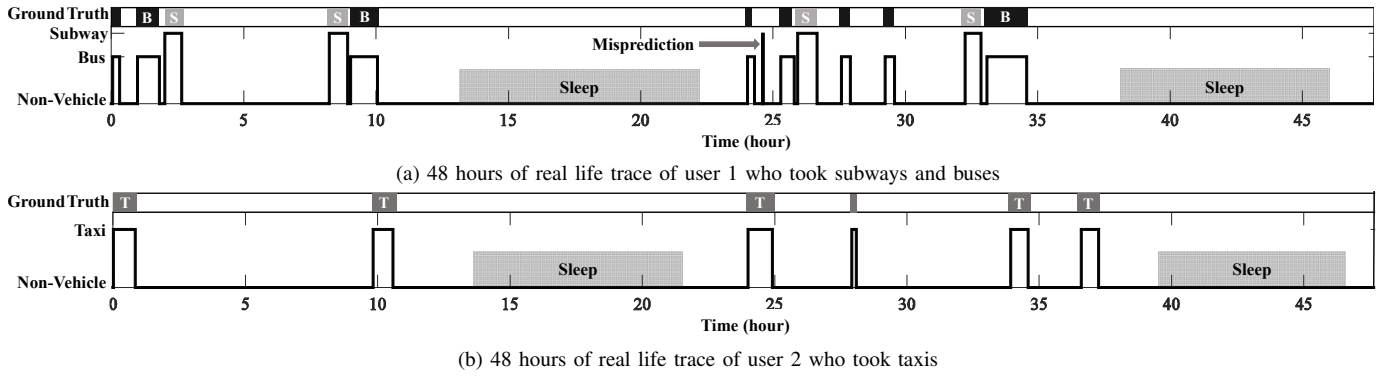


Fig. 12: In-situ performance of VehicleSense measured by two users who took (a) buses and subways and (b) taxis in their real lives. The figures show the ground truth of taking transportations and decisions made by VehicleSense. VehicleSense almost perfectly distinguishes the transportation modes as well as captures the moments of change.

in the first user, which is corrected immediately. Further, we observe that the median value of the total decision latency is 3.7 seconds, each of which is measured from every moment of riding a vehicle in the ground truth. Note that the total decision latency consists of two delay factors: the delay of about a few seconds from the riding motion of a user and the system-wide delay of about 0.6 seconds in VehicleSense.

VIII. CONCLUDING REMARKS

In this work, we proposed and implemented a reliable sound-based transportation mode recognition system called VehicleSense. The main features of VehicleSense are accelerometer-based hierarchical triggers and built-in microphone-based sound classifier. In VehicleSense, those features cooperate closely to achieve high recognition accuracy, low latency and minimal power consumption at the same time. Through extensive simulations with 186-hour log, we verified that VehicleSense achieves over 98.2% of recognition accuracy with only 0.5 seconds of sound sampling and consumes only 26.1 mW on average. We found that these performances are persistently observable in various Android smartphone models and the high accuracy is indeed achieved even in real-life experiments. For the privacy concern raised from sound sampling, we also devised the privacy protection mode of VehicleSense, which still shows comparable performance with VehicleSense using continuous sound sample.

ACKNOWLEDGEMENT

This work was supported by the MSIP of Korea government under Grant NRF-2016R1D1A1B03935042 and Grant IITP No. 2015-0-00278 (Research on Near-Zero Latency Network for 5G Immersive Service).

REFERENCES

- [1] "Average time spent per day with major media by US adults, 2011-2017," eMarketer, Oct 2015.
- [2] O. D. Incel, M. Kose, and C. Ersoy, "A review and taxonomy of activity recognition on mobile phones," *BioNanoScience*, vol. 3, May 2013.
- [3] M. Sundholm, J. Cheng, B. Zhou, A. Sethi, and P. Lukowicz, "Smart-mat: Recognizing and counting gym exercises with low-cost resistive pressure sensing matrix," in *Proc. of UbiComp*, September 2014.
- [4] L. Stenneth, O. Wolfson, P. S. Yu, and B. Xu, "Transportation mode detection using mobile phones and GIS information," in *Proc. of ACM SIGSPATIAL GIS*, Nov 2011.
- [5] T. Feng, J. Yang, Z. Yan, E. M. Tapia, and W. Shi, "TIPS: Context-aware implicit user identification using touch screen in uncontrolled environments," in *Proc. of ACM HotMobile*, Feb 2014.
- [6] "Study on architecture for next generation system," 3GPP, Tech. Rep. 23.799, Sep 2016, <https://portal.3gpp.org/>.
- [7] S. Hemminki, P. Nurmi, and S. Tarkoma, "Accelerometer-based transportation mode detection on smartphones," in *Proc. of ACM SenSys*, November 2013.
- [8] 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 *Proc. of ACM Sensys*, November 2014.
- [9] M. Han, L. T. Vinh, Y.-K. Lee, and S. Lee, "Comprehensive context recognizer based on multimodal sensors in a smartphone," *Sensors*, vol. 12, pp. 12 588–12 605, Sep 2012.
- [10] M. A. Shafique and E. Hato, "Use of acceleration data for transportation mode prediction," *Transportation*, vol. 42, no. 1, Jan 2015.
- [11] R. C. Shah, C. yih Wan, H. Lu, and L. Nachman, "Classifying the mode of transportation on mobile phones using GIS information," in *Proc. of UbiComp*, September 2014.
- [12] P. Zhou, Y. Zheng, and M. Li, "How long to wait?: Predicting bus arrival time with mobile phone based participatory sensing," in *Proc. of ACM MobiSys*, June 2012.
- [13] Y.-S. Lee and S.-B. Cho, "Activity recognition using hierarchical hidden markov models on a smartphone with 3d accelerometer," in *Proc. of HAIT*, vol. 6678, May 2011.
- [14] "Monsoon power monitor," <https://www.msoon.com>.
- [15] "Spectral density," https://en.wikipedia.org/wiki/Spectral_density.
- [16] "Statistics brief world metro figures," Union Internationale des Transports Publics (UITP), Oct 2014.
- [17] V. R. Vijaykumar, P. T. Vanathi, and P. Kanagasapabathy, "Modified adaptive filtering algorithm for noise cancellation in speech signals," *Elektronika ir Elektrotechnika*, vol. 74, pp. 17–20, 2007.
- [18] P. H. Veltink, H. B. J. Bussmann, W. de Vries, W. L. J. Martens, and R. C. V. Lummel, "Detection of static and dynamic activities using uniaxial accelerometers," *IEEE Transactions on Rehabilitation Engineering*, vol. 4, no. 4, Dec 1996.
- [19] S. Trost, P. D. Loprinzi, R. Moore, and K. Pfeiffer, "Comparison of accelerometer cut points for predicting activity intensity in youth," *Medicine and Science in Sports and Exercise*, vol. 43, no. 7, Jul 2011.
- [20] J. Shawe-Taylor and N. Cristianini, *Support Vector Machines*. Cambridge University Press, 2000.
- [21] Z. Fang, Z. Guoliang, and S. Zhanjiang, "Comparison of different implementations of mfcc," *Journal of Computer Science and Technology*, vol. 16, pp. 582–589, Feb 2001.
- [22] X. Zhou, D. Garcia-Romero, R. Duraiswami, C. Espy-Wilson, and S. Shamma, "Linear versus mel frequency cepstral coefficients for speaker recognition," in *Proc. of IEEE ASRU*, Dec 2011, pp. 559–564.
- [23] D. M. W. Powers, "Evaluation: From precision, recall and f-measure to roc, informedness, markedness & correlation," *Journal of Machine Learning Technologies*, vol. 2, pp. 37–63, 2011.