Towards a Practical Pedestrian Distraction Detection Framework using Wearables

Nisha Vinayaga-Sureshkanth[1,*], Anindya Maiti[2], Murtuza Jadliwala[1], Kirsten Crager[2], Jibo He[2], Heena Rathore[3]

[1]The University of Texas at San Antonio, Texas, USA [2]Wichita State University, Kansas, USA [3]Hiller Measurements, Texas, USA

***Abstract*—Pedestrian safety continues to be a significant concern in urban communities with distraction being one of the main contributing factor behind serious accidents involving pedestrians. The advent of sophisticated mobile and wearable devices, equipped with high-precision on-board sensors capable of measuring fine-grained user movements and context, provides a tremendous opportunity for designing effective pedestrian safety systems and applications. Accurate recognition of pedestrian distractions in real-time given the memory, computation and communication limitations of these devices, however, remains a key technical challenge in the design of such systems. Earlier research efforts in this direction have primarily focused on achieving high distraction detection accuracy, resulting in techniques that are either resource intensive and unsuitable for implementation on mainstream mobile devices, or computationally slow and not real-time, or require specialized hardware and thus less likely to be adopted by most users. Our goal in this paper is to design a pedestrian distraction detection technique that overcomes some of these shortcomings (of existing techniques) and achieves a favorable balance between computational efficiency, detection accuracy, and energy consumption.**

***Index Terms*—Pedestrian, distraction, mobile, wearables.**

I. INTRODUCTION

Pedestrian safety has become a critical concern as the number of serious and fatal injuries due to pedestrian-related accidents continue to steadily rise every year [1]. As one of the major causes of such pedestrian-related accidents, distracted driving has received significant attention over the past decade [2], which has resulted in a host of techniques to detect and overcome distraction during driving. However, nearly 50% [3] of all traffic related pedestrian deaths can be attributed to distraction among pedestrians (for example, inattentiveness while crossing roads and failure to obey traffic signs) rather than distracted drivers, which highlights the significant role pedestrian distraction plays in these accidents [4], [5]. Besides this, distracted pedestrians are also susceptible to other non-traffic hazards in indoor and outdoor environments, such as, falling over the edge of a subway platform, walking into obstacles, falling down a stairway, colliding with other pedestrians, and falling into an uncovered sewer manhole [6]. It is evident that distracted pedestrians pose a significant threat not only to their own safety, but also to the safety of other pedestrians (and drivers), and effective systems and mechanisms to overcome this threat are critically needed.

A pedestrian safety system typically comprises of two main components (Figure 1): (i) a *distraction* or *hazard detection* component, and (ii) an *accident prevention* component. The advent of mobile and wearable devices (e.g., smartphones and smartwatches), equipped with a variety of high-precision sensors capable of capturing fine-grained user movements and context, provides a great opportunity to design sound distraction detection and recognition techniques. However, designing techniques that are accurate, efficient and real-time, is not straightforward, due to the memory, computation and communication limitations of these devices.

Several recent research efforts in the literature have attempted to improve pedestrian safety by detecting hazardous contexts (e.g., incoming vehicles, obstacles, uncovered manholes, etc.) with the help of data available from users' smartphone camera [7]–[10] or from specialized sensors (e.g., ultrasonic sensors or depth cameras) attached to the phones [11]–[13]. In addition to shortcomings such as reliance on smartphone camera feed or other specialized sensors and devices which limits their functionality, several of these schemes employ computationally-intensive data processing techniques that are challenging to implement on resource-constrained mobile and wearable devices. More importantly, the above techniques fail to generalize the problem of pedestrian distraction detection by not considering a diverse range of complex and concurrent activities that commonly resemble distraction, for example, detecting when users are walking, running or descending staircases and simultaneously reading, eating or drinking [14], [15]. As a result, the above solutions are unable to recognize a wide variety of distracting activities.

The key to designing a pedestrian safety system that has broad application and usage is to first generalize the problem of detecting distracted pedestrians as a *concurrent activity recognition* (or *CAR*) problem. Several robust and accurate CAR frameworks that detect and recognize a variety of human activities, and their complex combinations, by using data available from commercial mobile and wearable device sensors have already been proposed in the literature [16]–[19]. However, the applicability of these models for pervasive pedestrian distraction detection applications is unclear and has not been well-studied. It appears that a majority of these CAR models proposed in the literature, owing to their use of computationally expensive data processing and analysis techniques, could be challenging to implement and/or efficiently operate on consumer-grade mobile and wearable devices that possess limited computational and energy resources.

*Corresponding author. E-mail: vsnisha@ieee.org.

[1]Nisha and Murtuza were with Wichita State when this work was done.

Research reported in this publication was supported by the United States National Science Foundation (NSF) under award number 1637290.

These shortcomings necessitate further investigation in two directions, which will be pursued by us in this paper: (i) *is it possible to design a generic pedestrian distraction detection approach that can operate on existing commercial mobile and wearable devices and achieve a favorable balance between computational efficiency, detection accuracy, and energy consumption?* and (ii) *how do existing concurrent activity recognition frameworks perform in a pedestrian distraction detection scenario?* In line with these objectives, we first design a novel complex activity recognition technique, called *Dominant Frequency-based Activity Matching (DFAM)*, which employs a lightweight frequency matching approach on motion (accelerometer and gyroscope) data available from users' mobile and wearable devices to accurately and efficiently detect and recognize a wide variety of complex pedestrian distraction related activities. Next, we undertake a comprehensive comparative evaluation of the proposed technique with well-known complex activity recognition approaches in the literature by means of data collected from real human subject pedestrians.

II. Related Work

Pedestrian Safety Systems: Several research efforts in the literature have employed mobile and/or wearable devices to improve pedestrian safety by detecting hazardous contexts using users' smartphone camera [7]–[10]. One significant drawback of all these proposals is that they employ costly and resource-intensive image capture and processing techniques, which can adversely impact the performance and battery-life of mobile devices, thus diminishing their chances of being adopted by users. Reliance on a smartphone's camera also restricts the ability of these techniques to operate when the camera is obstructed, for example, in a user's pocket. Techniques for aiding pedestrian safety that do not rely on the camera input, but rather on a smartphone's microphone [20] and GPS [21] have also been proposed. However, these are useful only in detecting outdoor traffic-related hazards scenarios. Furthermore, techniques that employ specialized devices and sensors for improving pedestrian safety have also been proposed. *Lookup* [22] uses information from specialized motion sensors attached to pedestrians' shoes to profile step and slope in order to detect curbs, ramps and other obstructions. Similarly, Ramos and Irani [11] used a depth camera (paired with a smartphone), while Ahn and Kim [12] and [13] employed an ultrasonic sensor for detecting pedestrian hazards and/or for guided navigation. Besides relying on specialized sensors, these systems attempt to address pedestrian safety by detecting obstacles or other potential hazards (to pedestrians). *In this paper, we take an orthogonal approach to pedestrian safety by attempting to detect inattentiveness among pedestrians; after all if pedestrians are not distracted they will be able to easily navigate away from obstacles and other hazards.*

Concurrent Activity Recognition (CAR): The problem of detecting distracted pedestrians can be generalized as a concurrent activity recognition or CAR problem where the goal is to detect concurrent pedestrian activities of being mobile (e.g., walking, running or climbing/descending stairs) and being distracted (e.g., texting, eating or reading). CAR techniques that can distinguish different combinations of elementary activities have been extensively used in the literature for complex human activity recognition. For instance, Shoaib et al. [17], [23] used *multi-source* and *multi-sensor motion* data, from two smartphones, one in trouser pocket and the other on the wrist, to recognize activities that involve hand gestures, such as smoking, eating, drinking coffee and giving a talk. Liu et al. [18] also employed multi-sensor time series data to recognize sequential, concurrent, and generic complex activities by building a dictionary of time series patterns (called *shapelets*) to represent atomic activities. However, several shortcomings in these approaches, as outlined below, prevent them from being effectively used in pedestrian safety applications. For instance, [23] requires the system to keep track of time segments that precede and follow the current one and thus unsuitable for pedestrian safety applications that require real-time operation and feedback. Others are not suitable for implementation on resource-constrained mobile and wearable devices, primarily due to their use of complex feature sets and classification functions. As discussed before, one of the main functional requirement for a mobile/wearable device based CAR framework for pedestrian safety is computational and energy efficiency. Earlier research efforts in energy-aware recognition mechanisms [19] have achieved a favorable balance between classification accuracy and energy consumption, but these schemes have been successful in recognizing only simple activities, such as, standing, walking and sitting, but not concurrent activities.

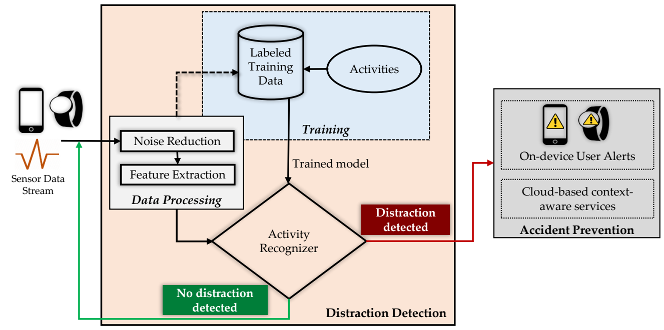


Fig. 1: A generic pedestrian safety system.

III. Pedestrian Distraction Detection

As outlined earlier, any pedestrian safety system typically comprises of two main components (Figure 1): (i) a *distraction* or *hazard detection* component, and (ii) an *accident prevention* component. In this paper, we primarily focus on the former. Figure 1 depicts the design of a generalized learning based framework which is the main building block for pedestrian distraction detection in such systems. As shown in the figure, the distraction detection framework comprises of: (i) a *data processing module* (includes, noise removal, segmentation, and feature generation), and (ii) a *CAR model building phase* (includes design of an appropriate activity classification function and training it using processed labeled training data). Once a trained CAR model is available, it can be used to recognize (or classify) distracted pedestrian activities. Such a design of the distraction detection framework is commonly employed in the literature (and in practice) for pedestrian safety and other applications, and will also be employed by us in this paper. Our distraction detection framework relies on multi-sensor data obtainable from multiple mobile devices carried by the pedestrians, specifically, motion (including, data from accelerometer and gyroscope sensors) and contextual information from the pedestrian's smartphone and smartwatch. The data processing module in our framework filters this multi-sensor data (to eliminate errors and inconsistencies), segments it into fixed-size blocks or windows, and extracts relevant features from it. It should, however, be noted that the data processing task may vary depending on the chosen CAR technique. The extracted features are then used to train appropriate CAR models within a supervised learning paradigm. These trained CAR models are utilized by our framework for distracted activity classification or recognition tasks. As an appropriate CAR technique is central to the design of a pedestrian distraction detection framework that can attain a practicable balance between computational efficiency, detection accuracy and energy consumption, in this section we focus on designing such a technique. Technical details of our proposed CAR technique, referred to as DFAM, are presented next. We also outline other well-known techniques that have been employed in the literature [17], [23] for similar activity classification tasks, as we later empirically compare the performance of our DFAM technique against these classical activity classification techniques.

A. Dominant Frequency-based Activity Matching (DFAM)

A majority of the time domain features utilized for concurrent activity recognition [17], [23] are computationally intensive and thus not suitable for real-time pedestrian distraction detection. Moreover, with multi-source (smartphone and smartwatch) data, data fusion (before feature extraction) proves to be a challenging task owing to source (time) synchronization issues in high-precision data streams. Our proposed DFAM CAR technique addresses this problem, and is capable of computing features (frequency domain, particularly dominant frequencies) directly and independently on the different devices, such as smartphone and smartwatch. Moreover, DFAM can utilize multiple *frequency bins* to extract more than one dominant frequency, which can be used to characterize and recognize activities more accurately. As a result, by using frequency domain features, we can avoid resource intensive sample-to-sample synchronization required before computing multi-source time domain features.

DFAM is inspired from the audio matching algorithm proposed by Avery Wang [24]. Proprietary versions of Wang's algorithm are commonly used in popular song searching applications, such as *Shazam*. Due to the significant differences between audio (found in audio files) and motion data (sampled from the smartphone and smartwatches), it is non-trivial to use Wang's audio matching algorithm directly for activity recognition using motion data. In the audio matching application, matching features in the test audio file occur at almost identical relative time offsets from the beginning of the audio file being matched to. In contrast, motion data from pedestrian activities generally does not occur at exactly fixed time offsets, therefore requiring a new matching algorithm. Other differences between motion and audio data include a significantly lower sampling rate of smartphone and smartwatch motion sensors (compared to audio data which is generally sampled at a much higher frequency) and distinctly different *dominant frequency* ranges of both types of data. Recently, Sharma et al. [25] successfully applied *dominant frequency-based activity matching* for simple (non-concurrent) activities, using fixed threshold-based classifiers. In this paper, we use preprocessing techniques used by Wang [24] and extend Sharma et al.'s work significantly in order to recognize concurrent activities related to pedestrian distractions.

DFAM Training: During the training phase, (low-pass) filtered time-series motion data from the smartphone and smartwatch, denoted as T_p and T_w, respectively, corresponding to each activity of interest is first *segmented* into smaller fixed-sized windows of W samples in a sliding fashion with possible overlap for maximum data utilization. Let's assume that this motion data is sampled at a frequency f_s.

$$T_p = \{{}^{1}b_p, {}^{2}b_p, \ldots, {}^{m}b_p\}; \quad T_w = \{{}^{1}b_w, {}^{2}b_w, \ldots, {}^{n}b_w\} \tag{1}$$

$$\text{where} \quad m = \frac{sizeof(T_p)}{W} \quad \text{and} \quad n = \frac{sizeof(T_w)}{W}$$

After this pre-processing step, the frequency response of each window in T_p and T_w is independently calculated using a discrete Fourier transformation technique such as a fast Fourier transform (or FFT [26]). Let the frequency responses corresponding to T_p and T_w be represented as F_p and F_w, respectively.

$$F_p = \{{}^{1}r_p, {}^{2}r_p, \ldots, {}^{m}r_p\}; \quad F_w = \{{}^{1}r_w, {}^{2}r_w, \ldots, {}^{n}r_w\} \tag{2}$$

$$\text{where} \quad {}^{i}r_p = FFT({}^{i}b_p) \quad \text{and} \quad {}^{i}r_w = FFT({}^{i}b_w)$$

Each of the frequency response blocks ${}^{i}r_p \in F_p$ and ${}^{i}r_w \in F_w$ are then analyzed for a dominant frequency in g (empirically determined) frequency bins, with one dominant frequency in each bin:

$$\{f_{(0,u_1)}, f_{(u_1,u_2)}, \ldots, f_{(u_{g-1}, \frac{f_s}{2})}\}$$

All of the observed dominant frequency in each of the g bins are then hashed (or compressed) to create a '*signature*' for the activity. As we are employing multiple devices and sensors, with each sensor possibly outputting measurements across multiple dimensions (e.g., each accelerometer sensor measurement is across three dimensions), each training data point will consist of measurements across multiple dimensions. For example, a dominant frequency analysis on three-dimensional (x, y, z) time series data window will result in a three-dimensional training point $\langle H_x, H_y, H_z\rangle$, where H_x, H_y, and H_z are the hashes of dominant frequencies on respective axes. Now, let us denote the set of all distracted activities as $\mathbb{D}$, and the set of all pedestrian activities as $\mathbb{P}$. For each activity $a_u \in \mathbb{P}$, a training dataset made of equalized data points is created during the training phase, and stored along with the corresponding label a_u. Similarly, for the each concurrent activity $a_v \in \mathbb{P}\times\mathbb{D}$, another training dataset made of equalized data points is created during the training phase, and stored along with the corresponding label a_v.

DFAM Activity Classification: To correctly classify the current or test user activity (say, a_c), DFAM employs a dominant frequency matching technique using the labeled training data (from the previous phase), as described below. Given a test window with s-axis signatures, the activity is matched using the following scoring function:

$$S_{i,j}(a_c) = \begin{cases} 0 & \text{if } \sum_{k=1}^{s} F({}^{c}H_k, {}^{\text{train}}H_k^{i,j}) = 0 \\ (\frac{1}{s})^s & \text{if } \sum_{k=1}^{s} F({}^{c}H_k, {}^{\text{train}}H_k^{i,j}) = 1 \\ (\frac{2}{s})^s & \text{if } \sum_{k=1}^{s} F({}^{c}H_k, {}^{\text{train}}H_k^{i,j}) = 2 \\ \vdots & \vdots \\ (\frac{s-1}{s})^s & \text{if } \sum_{k=1}^{s} F({}^{c}H_k, {}^{\text{train}}H_k^{i,j}) = s-1 \\ 1 & \text{if } \sum_{k=1}^{s} F({}^{c}H_k, {}^{\text{train}}H_k^{i,j}) = s \end{cases}$$

where $S_{i,j}(a_c)$ is the matching score per training instance j in each activity $a_i \in \mathbb{P} \times \mathbb{D}$, ${}^{c}H_k$ is the current activity signature from k-th sensor axis, ${}^{\text{train}}H_k^{i,j}$ is the signature from k-th sensor axis of j-th training instance of activity a_i, and

$$F(a,b) = \begin{cases} 0 & a \neq b \\ 1 & a = b \end{cases}$$

The above scoring function gives exponentially more weight to multi-dimensional signature matches, which will intuitively result in a higher score when matching with the ground truth activity. Finally, the activity is classified after matching against the entire training dataset of all activities as follows:

$$arg \max_{i} \sum_{j} S_{i,j}(a_c) \quad \forall a_i \in \mathbb{P} \times \mathbb{D} \tag{3}$$

The current activity a_c is then classified as that activity a_i which achieves the maximum aggregated score as shown in Equation 3.

B. Traditional Classifiers

Traditional supervised learning-based classification functions, such as *Naive Bayes (NB)*, *Decision Tree (DT)*, *Random Forests (RF)*, *Support Vector Machine (SVM)*, *k-Nearest Neighbours (k-NN)*, have been successfully used in the literature (and in practice) for detecting complex and concurrent human activities [17], [23]. Given that distracted pedestrian activities are inherently concurrent activities, these supervised learning-based techniques comprise of a suitable candidate set for a comparative performance evaluation with our proposed DFAM technique. Below, we outline how these classification techniques are employed within our pedestrian distraction detection framework, and provide details on the related data pre-processing, feature extraction and model training tasks.

Data Processing: The (low-pass) filtered time-series motion data from the smartphone and smartwatch, denoted as T_p and T_w respectively, corresponding to each activity of interest is first *segmented* into smaller fixed-sized windows, as discussed earlier for DFAM. Each of the motion data stream T_p and T_w comprises of both the accelerometer and gyroscope sensor data sampled along all the three axes at some frequency f_s. A set of time and frequency domain features, as have been employed in the literature [17], [27]–[30] for activity recognition (and briefly outlined below), are then computed from each window of the time series motion data streams.

- Mean, minimum, maximum, standard deviation, variance, along with energy and entropy of discrete FFT components for each of the three axes of both the accelerometer and gyroscope time-series data.
- Root mean square (RMS) correlation measures among the three axes for each of the accelerometer and gyroscope data.
- Mean, median, and maximum of the instantaneous speed (only for the accelerometer data)
- Mean, median, and maximum of roll velocity (only for the gyroscope data)

As the motion time-series of an activity comprises of several windows, features computed for all the windows are combined to create a feature set for that activity. This process (filtering, segmentation, and feature extraction) is repeated for all the considered distraction-related activities in $\mathbb{D}$ and non-distraction activities in $\mathbb{P}$ in the training dataset to create a labeled feature set for all the activities. Such a labeled training (feature) set is then used to train each of the concurrent activity classification models. The above data pre-processing and feature extraction tasks remain the same for all the supervised learning based classification functions considered.

IV. Evaluation and Results

Next, we present a comprehensive comparative evaluation of the performance of the proposed DFAM technique against traditional classification techniques for distracted pedestrian activity recognition.

A. Experimental Setup

We collect motion sensor data of distracted pedestrian activities using a wrist-worn smartwatch and a paired smartphone (Motorola Moto XT1096). To test the versatility of our technique, we test it across two different smartwatches, namely a Sony Smartwatch 3 and a LG Urbane W150. A combination of smartwatch and smartphone was placed on participating pedestrians[1], for a total of four different device placement scenarios. For the same-side placements, either both smartwatch and smartphone are worn on the right wrist and placed inside right hip pocket (RR), or worn on the left wrist and placed inside the left hip pocket (LL). The remaining two scenarios alternate the placements to the opposite sides, i.e., smartwatch on right wrist along with phone in left hip pocket (RL), smartwatch on left with phone in right pocket (LR). Each participant performed a pre-defined but randomized set of activities for one or more of the scenarios. The set of activities consisted of non-pedestrian, pedestrian and distracted-pedestrian related activities outlined in Table I. All concurrent activities except the starred (*) activities form a set of distracted pedestrian activities.

We developed a custom Android application using Android Studio IDE v2.2.3 running on Java 8 platform, to record activity related motion sensor data from the Moto T1096 running on Android 6.0 and the smartwatches running on Android Wear 1.5, at a sampling rate of 50 Hz. The activity data collected includes three-dimensional accelerometer and gyroscope sensor data from the aforementioned devices. Throughout the data collection, we took several precautionary measures to ensure participant safety during certain distracted activities, due to potential falling and injury risks. For example, we placed a safety harness on the participant when descending

[1]A total of 23 participants took part in our study, which was approved by Wichita State University's Institutional Review Board (IRB).

TABLE I: Activities performed by the participants.

Simple Activities	Concurrent Activities	
Standing	Walking + Using Smartphone	Walking + Reading
Walking	Climbing stairs + Eating	Walking + Eating
Climbing stairs	Descending stairs + Eating	Walking + Drinking
Descending stairs	Climbing stairs + Drinking	Standing + Drinking*
Sitting	Climbing stairs + Using Smartphone	Standing + Reading*
Running	Descending stairs + Using Smartphone	Standing + Eating*
	Running + Using Smartphone	Sitting + Using Smartphone*
	Standing + Using Smartphone*	Descending stairs + Reading
	Descending stairs + Drinking	Climbing stairs + Reading

stairs and reading at the same time, while ensuring these safety measures did not interfere with the activities. On an average, each participant took about 2 hours to complete all the activities. The physical demands of our experiments, together with these additional constraints in selecting participants thereby limited our ability to recruit a larger number of participants, or obtain data for all possible device placements from the same participant.

We implemented the proposed DFAM technique using Java on (i) a 64-bit Debian Linux PC with an Intel Core i5 processor and 8 GB RAM and (ii) the Motorola Moto XT1096 smartphone. Implementations for the traditional classifiers were derived from the Weka 3 machine learning toolkit [31] and its Android counterparts. The PC implementation was used to extensively analyze the performance of DFAM, which is not possible on a resource constrained smartphone. On the other hand, the smartphone implementation is helpful in evaluating on-device response times and resource utilization in real-life usage. Next, we explore how different training sets can affect the classification accuracy of the proposed DFAM and other classification schemes. This step is crucial for the pedestrian safety application, because not all users would be willing to setup a personally trained model. In other words, personalized datasets may not be a realistic scenario, and thus the proposed DFAM scheme should work well with unseen test sets.

B. DFAM Performance

We first validate the feasibility of detecting distracted pedestrian using DFAM, by creating personalized models for each participant using their individual datasets, and then performing a Leave-One-Out Cross Validation (LOOCV) using the trained model. In LOOCV, one block is allotted as test data, while the rest remain in the training set. The individual participant accuracies are then averaged out for each group, where the datasets are grouped based on the smartwatch used, and further grouped based on the device placements (Table II). For different window W and bin sizes (g) of the collected data, we evaluate DFAM performance across three different averaging methods – weighted, micro and macro – based on metrics such as classification accuracy, precision, recall and F1 score as shown in Figures 2 and 3a.

Figures 2a and 2b show the precision, recall and F1 scores for same-side (LL, RR) and different side (LR, RL) device placement. We observed that the precision and recall improve with increasing number of frequency bins. The mean F1 score for $g = 3$ was 0.75, compared to 0.66 and 0.73 for $g = 1, 2$, respectively, for all four placement scenarios combined. We did not observe any significant performance difference using weighted, micro and macro averaging methods between the Sony+Moto and LG+Moto datasets. The mean classification

TABLE II: Datasets collected per placement scenario.

	RR	LL	RL	LR	Total
LG+Moto	5	6	4	4	19
Sony+Moto	6	6	5	5	22
Total	11	12	9	9	41

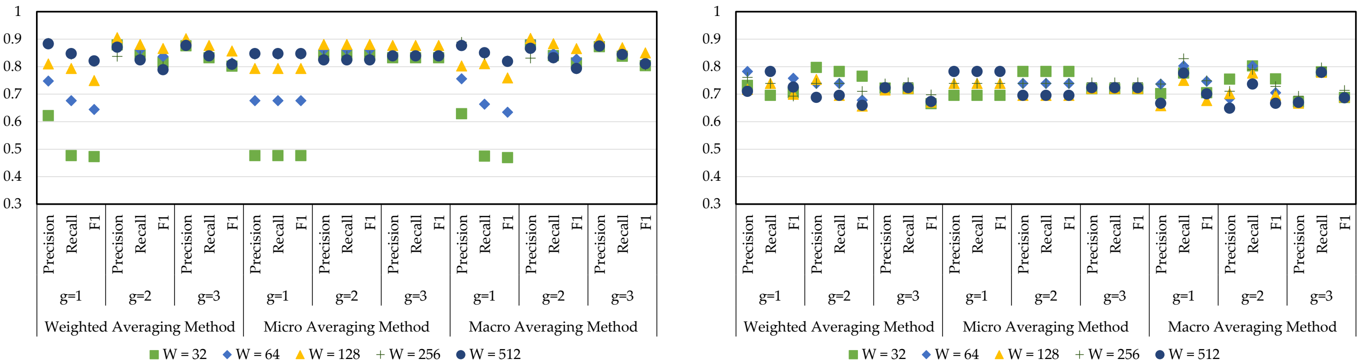
accuracy (for $g = 3$ and $W = \{32, 64, 128, 256, 512\}$) for Sony+Moto and LG+Moto datasets are 0.79 and 0.75, respectively, with a standard deviation of 0.07 and 0.06, respectively. This implies that the proposed DFAM is implementable across different wrist-based wearables. We also observed slight performance difference between the same-side and different-side datasets. The mean classification accuracy (for $g = 3$ and $W = \{32, 64, 128, 256, 512\}$) for same-side and different-side datasets are 0.81 and 0.72, respectively, with a standard deviation of 0.04 and 0.05, respectively. This implies that DFAM works slightly better for same-side smartphone and smartwatch placement.

Next, we investigate the effect of combining datasets from different participants from the same group to obtain a trained model, and validate it using k-fold cross validation where $k = 10$. In 10-fold cross validation, the dataset is split into 10 equal parts, one of which becomes the testing set, and the remaining nine folds constitute the training set. We compute the classification accuracies for different window W and bin g sizes as shown in Figure 3a. DFAM achieved classification accuracy of 0.70 for $g = 3$ and $W = 32$ (0.64 seconds at 50 Hz). As intuitively expected, classification accuracy improves as the window size is increased to $W = 512$ (10.24 seconds at 50 Hz), although at a cost of increased detection time as we evaluate later in Section IV-D.

In a real world implementation, it may not be practical to combine data exclusively from participants having the same hardware. Moreover, not all wearables may have both accelerometer and gyroscope sensors, compelling us to examine whether DFAM can classify the activities in the absence of either gyroscope (GYR) or accelerometer (ACC) data as shown in Figure 3b. We observed that classification accuracy dropped to 0.57 when using only accelerometer data, for $g = 3$ and $W = 32$ (compared to 0.70 in ACC+GYR datasets). Similarly, classification accuracy was 0.66 when using only gyroscope data, for $g = 3$ and $W = 32$. This implies that DFAM works better in the presence of both accelerometer and gyroscope sensors. We also reinforce our earlier observation that classification accuracy is highest for $g = 3$. As a result, we set bin size $g = 3$ for all following experiments.

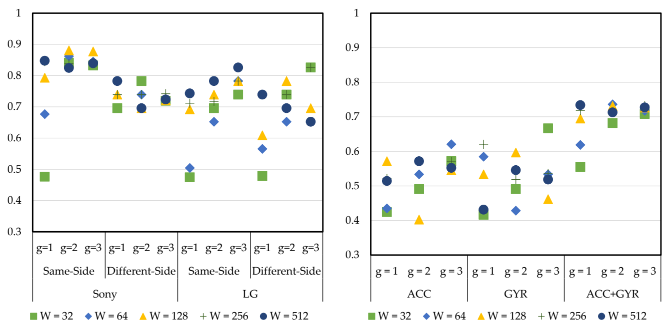
C. Comparison with Traditional Classifiers

A realistic setting involves using already trained models to recognize activities of a previously unseen participant. We evaluate and compare DFAM in such a setting by leaving out one participant's dataset for testing purposes and training using the rest. This Leave-One-Subject-Out (LOSO) approach validates the generalization performance of the CAR schemes. We compare the classification accuracies of DFAM, and other CAR schemes for different window sizes as shown in Table III. Results show that DFAM's classification accuracy is comparable with SVM, DT, RF, NB and 1-NN. However, 2-NN



(a) (b)

Fig. 2: DFAM performace for datasets of (a) same-side using Sony+Moto, (b) different-side using Sony+Moto.



(a) (b)

Fig. 3: Classification accuracy of DFAM for (a) combined datasets, and (b) individual sensors.

TABLE III: Classification accuracy of DFAM compared with traditional classifiers.

	DFAM	SVM	DT	RF	NB	1-NN	2-NN	3-NN
W = 32	0.45	0.43	0.47	0.51	0.52	0.52	0.47	0.51
W = 64	0.48	0.44	0.57	0.58	0.46	0.64	0.52	0.64
W = 128	0.54	0.48	0.42	0.51	0.48	0.52	0.52	0.52
W = 256	0.51	0.48	0.49	0.67	0.54	0.51	0.52	0.52
W = 512	0.54	0.54	0.54	0.52	0.54	0.54	0.54	0.53

and 3-NN performs better than DFAM in most cases, but they also impose higher resource utilization as we evaluate next.

D. Response Time and Resource Utilization

We next evaluate the response time, CPU, RAM and power consumption of the CAR models on the Motorola XT1096 smartphone paired with the Sony Smartwatch 3. The XT1096 with a 2300mAh Li-ion battery was running Android 6.0, while the Smartwatch 3 with a 420mAh Li-ion battery was running Android Wear 1.5. For this analysis, we use the same participant data and pre-trained classification models from Section IV-B. However, the signature (or feature) generation and matching (or classification) is executed on the mobile and wearable device, unlike the previous evaluation (Section IV-B) where they were executed on a PC. Table IV compares the response time and resource utilization of DFAM compared with traditional classifiers, with ranges signifying varying window sizes. The response time excludes the communication delays, time taken to obtain a data block and consumed time not related to generation of block-related features, which would be same for all the techniques. The time taken to obtain the data blocks remained constant across the different CAR models for a window size W, along with the time taken to generate features across the traditional CAR models. The CPU utilization, power consumption and RAM utilization were also recorded in these trials over a period of two minutes and repeated 10 times. The RAM usage is measured in megabytes (MB), whereas the CPU utilization is in percentage indicating the fraction of available processing power used.

Results show that DFAM has significantly lower response time compared to traditional classifiers, which is beneficial for alerting distracted pedestrian in real-time. CPU utilization, power consumption and RAM utilization are also on the lower side for DFAM, which means users will notice minimal impact on performance of their smartphone. Notably, 2-NN and 3-NN which achieved slightly better classification accuracy earlier, also have the highest response times and generally consumes more system resources. Good classification response time is vital in determining the effectiveness of any pedestrian safety framework, because any delay in alerting distracted pedestrians can be decisive in potential accident preventions. Our evaluation results so far positions DFAM as a sound and most suitable CAR technique for pedestrian distraction detection in a pedestrian safety system that employs motion data from mobile devices.

V. Discussion and Future Work

Accident Prevention: In this work, we limit ourselves to solely studying the viability of using a CAR-based framework for improving pedestrian safety. In order to prevent unwanted distraction related injuries and fatalities, we plan to develop an *on-device alert module* for users' mobile and/or wearable devices, to remind distracted pedestrians that they should pay more attention to their surroundings while they are in motion. Additionally, we plan to implement a *cloud-based alert module* that will employ crowd-sourced contextual information from distracted users to alert other users in the vicinity about the presence of distracted pedestrians. The design of these alert modules, however, is not trivial and requires a careful analysis of the associated human-factors issues. An alert mechanism that is not carefully designed may annoy users with frequent notifications, who may in turn decide not use it anymore, or may become a source of distraction themselves. We plan to accomplish this as part of our future work.

Other Future Work: In this work, we validated the performance of the proposed CAR technique (DFAM) and

TABLE IV: Average response time and resource utilization of DFAM compared with traditional classifiers.

	DFAM	SVM	DT	RF	NB	1-NN	2-NN	3-NN
Response Time (ms)	640-1150	2000-6000	1200-4480	2800-10630	890-2980	2800-8600	1900-8600	2900-6600
CPU Utilization	0.5-4.5%	1.3-8%	0.3-1.6%	0.5-7.4%	0.7-2.4%	1.5-3.8%	1.2-3%	1.2-2.9%
Power Consumption (mW)	33.3-129.5	33.3-188.7	33.3-85.1	85.1-222	40.7-96.2	85.1-214.6	85.1-188.7	85.1-218.3
RAM Utilization (MB)	20-24	26-53	17-67	56-108	19-29	15-26	29-53	30-92 MB

its Android implementation across different wearable device hardwares, i.e., smartwatches. It will also be interesting to study how the performance of the proposed CAR technique and its implementation varies across different smartphone hardwares. Also, we reckon that the CAR techniques can be employed in a hierarchical CAR model, where the viability and efficiency of integrating these techniques would depend on their feature sets used (for classification) in each state of the model. As part of future work, we plan to conduct a comprehensive comparative evaluation of one such hierarchical model.

VI. Conclusion

We outlined and comprehensively evaluated a novel framework that detects and recognizes distracted pedestrian activities using motion data available from users' mobile and wearable devices. As part of our framework, we designed and evaluated a novel dominant frequency matching based concurrent activity recognition model, called DFAM, and compared the performance and execution efficiency of the DFAM model with other well-known learning-based classification functions, such as Random Forests, SVM, k-NN, Naive Bayes and Decision Trees. Our evaluation results showed that the proposed DFAM model is a suitable candidate for detecting concurrent activities, such as that of distracted pedestrians, and that it has reasonable concurrent activity recognition accuracy compared to traditional classification functions. We also observed that DFAM has comparatively lower power consumption rates and quicker response time(s). In summary, we have not only comprehensively evaluated the efficacy and feasibility of various concurrent activity recognition techniques for detecting and recognizing pedestrian distraction, but have also proposed a novel concurrent activity recognition technique that achieves a good balance between recognition accuracy and alert response time, while being energy efficient.

References

[1] A. Williams, "Pedestrian traffic fatalities by state," *Washington, DC: Governors Highway Safety Association*, 2013.

[2] K. Young, M. Regan, and M. Hammer, "Driver distraction: A review of the literature," *Distracted Driving*, 2007.

[3] T. J. Bungum, C. Day, and L. J. Henry, "The association of distraction and caution displayed by pedestrians at a lighted crosswalk," *Journal of Community Health*, 2005.

[4] I. E. Hyman, S. M. Boss, B. M. Wise, K. E. McKenzie, and J. M. Caggiano, "Did you see the unicycling clown? inattentional blindness while walking and talking on a cell phone," *Applied Cognitive Psychology*, 2010.

[5] D. C. Schwebel, D. Stavrinos, K. W. Byington, T. Davis, E. E. O'Neal, and D. De Jong, "Distraction and pedestrian safety: how talking on the phone, texting, and listening to music impact crossing the street," *Accident Analysis & Prevention*, 2012.

[6] R. D. Lang, "Don't text, talk, and walk: the emerging smartphone defense in personal injury litigation," *Albany Law Review*, 2013.

[7] T. Wang, G. Cardone, A. Corradi, L. Torresani, and A. T. Campbell, "Walksafe: a pedestrian safety app for mobile phone users who walk and talk while crossing roads," in *ACM MCSA Workshop*, 2012.

[8] K.-T. Foerster, A. Gross, N. Hail, J. Uitto, and R. Wattenhofer, "Spareeye: enhancing the safety of inattentionally blind smartphone users," in *ACM MUM*, 2014.

[9] Z. Wei, S.-W. Lo, Y. Liang, T. Li, J. Shen, and R. H. Deng, "Automatic accident detection and alarm system," in *ACM MM*, 2015.

[10] E. Peng, P. Peursum, L. Li, and S. Venkatesh, "A smartphone-based obstacle sensor for the visually impaired," in *International Conference on Ubiquitous Intelligence and Computing*. Springer, 2010.

[11] J. D. Hincapié-Ramos and P. Irani, "Crashalert: enhancing peripheral alertness for eyes-busy mobile interaction while walking," in *ACM CHI*, 2013.

[12] E. Ahn and G. J. Kim, "Casual video watching during sensor guided navigation," in *ACM SIGGRAPH VRCAI*, 2013.

[13] J. Wen, J. Cao, and X. Liu, "We help you watch your steps: Unobtrusive alertness system for pedestrian mobile phone users," in *IEEE PerCom*, 2015.

[14] J. Mwakalonge, S. Siuhi, and J. White, "Distracted walking: examining the extent to pedestrian safety problems," *Journal of traffic and transportation engineering (English edition)*, 2015.

[15] J. Ogden, E. Oikonomou, and G. Alemany, "Distraction, restrained eating and disinhibition: An experimental study of food intake and the impact of eating on the go," *Journal of Health Psychology*, 2017.

[16] J. Korpela, K. Takase, T. Hirashima, T. Maekawa, J. Eberle, D. Chakraborty, and K. Aberer, "An energy-aware method for the joint recognition of activities and gestures using wearable sensors," in *ACM ISWC*, 2015.

[17] M. Shoaib, S. Bosch, O. D. Incel, H. Scholten, and P. J. Havinga, "Complex human activity recognition using smartphone and wrist-worn motion sensors," *Sensors*, 2016.

[18] L. Liu, Y. Peng, S. Wang, M. Liu, and Z. Huang, "Complex activity recognition using time series pattern dictionary learned from ubiquitous sensors," *Information Sciences*, 2016.

[19] Z. Yan, V. Subbaraju, D. Chakraborty, A. Misra, and K. Aberer, "Energy-efficient continuous activity recognition on mobile phones: An activity-adaptive approach," in *ACM ISWC*, 2012.

[20] J. Lee and A. Rakotonirainy, "Acoustic hazard detection for pedestrians with obscured hearing," *IEEE Transactions on Intelligent Transportation Systems*, 2011.

[21] C.-H. Lin, Y.-T. Chen, J.-J. Chen, W.-C. Shih, and W.-T. Chen, "psafety: A collision prevention system for pedestrians using smartphone," in *IEEE VTC-Fall*, 2016.

[22] S. Jain, C. Borgiattino, Y. Ren, M. Gruteser, Y. Chen, and C. F. Chiasserini, "Lookup: Enabling pedestrian safety services via shoe sensing," in *ACM MobiSys*, 2015.

[23] M. Shoaib, H. Scholten, P. J. Havinga, and O. D. Incel, "A hierarchical lazy smoking detection algorithm using smartwatch sensors," in *IEEE Healthcom*, 2016.

[24] A. Wang, "An industrial strength audio search algorithm." in *ISMIR*, 2003.

[25] A. Sharma, A. Purwar, Y.-D. Lee, Y.-S. Lee, and W.-Y. Chung, "Frequency based classification of activities using accelerometer data," in *IEEE MFI*, 2008.

[26] D. Batenkov, "Fast fourier transform," in *Key Papers in Computer Science Seminar*, 2005.

[27] L. Sun, D. Zhang, B. Li, B. Guo, and S. Li, "Activity recognition on an accelerometer embedded mobile phone with varying positions and orientations," *Ubiquitous intelligence and computing*, 2010.

[28] J. P. Varkey, D. Pompili, and T. A. Walls, "Human motion recognition using a wireless sensor-based wearable system," *Personal and Ubiquitous Computing*, vol. 16, no. 7, pp. 897–910, 2012.

[29] A. Parate, M.-C. Chiu, C. Chadowitz, D. Ganesan, and E. Kalogerakis, "Risq: Recognizing smoking gestures with inertial sensors on a wristband," in *ACM MobiSys*, 2014.

[30] T. Ilmjärv *et al.*, "Detecting user reading behaviour using smartphone sensors," Ph.D. dissertation, Tartu Ülikool, 2015.

[31] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: an update," *ACM SIGKDD explorations newsletter*, 2009.