2018

# Design, Implementation and a Pilot Study of Mobile Framework for Pedestrian Safety Using Smartphone Sensors

Aawesh Man Shrestha
*South Dakota State University*

DESIGN, IMPLEMENTATION AND A PILOT STUDY OF MOBILE FRAMEWORK FOR PEDESTRIAN SAFETY USING SMARTPHONE SENSORS

BY

AAWESH MAN SHRESTHA

A thesis submitted in partial fulfillment of the requirements for the

Master of Science

Major in Computer Science

South Dakota State University

2018

DESIGN, IMPLEMENTATION AND A PILOT STUDY OF MOBILE FRAMEWORK FOR PEDESTRIAN SAFETY USING SMARTPHONE SENSORS

AAWESH MAN SHRESTHA

This thesis is approved as a credible and independent investigation by a candidate for the Master of Science degree and is acceptable for meeting the thesis requirements for this degree. Acceptance of this thesis does not imply that the conclusions reached by the candidate are necessarily the conclusions of the major department.

Myounggyu Won, Ph.D.
Thesis Advisor — Date

George H Hamer, Ph.D.
Head, Department of EECS — Date

Kinchel C.Doerner, Ph.D.
Dean, Graduate School — Date

## ACKNOWLEDGEMENTS

I cannot express enough thanks to my research advisor Dr. Myounggyu Won of Department of Computer Science at South Dakota State University. Dr. Won always encourage me and give me advices to increase quality of my thesis. He consistently allowed my research to be my own work while continuously steering me in the right direction.

I would also like to thank the committee members for my thesis especially Dr. Shin and Dr. Salenhia who gave me valuable feedback during my preliminary presentation. Their constructive comments substantially helped me in improving the quality of my thesis. Most importantly, I am grateful to my parents (Lalit Man Shrestha and Tara Shrestha), without whose love, support and encouragement, I would not be where I am today.

Finally, I would like to thank the Department of Electrical Engineering and Computer Science and all the graduate faculty of the department for their continuous support and encouragement I have received during the course of my degree program. The courses I took from them helped me greatly to learn about fundamental materials that made my research activities possible.

# TABLE OF CONTENTS

# LIST OF FIGURES

# ABBREVIATIONS

| | |
|---|---|
| AES | Advanced Encryption Standard |
| AP | Access Point |
| API | Application Program Interface |
| CCMP | Counter Mode with Cipher Block Chaining Message Authentication Code Protocol |
| CDF | Cumulative Distribution Function |
| DHCP | Dynamic Host Configuration Protocol |
| DSRC | Dedicated Short Range Communication |
| geo | Geodetic |
| GO | Group Owner |
| GPS | Global Positioning System |
| HMM | Hidden Markov Model |
| IEEE | Institute of Electrical and Electronics Engineers |
| LOS | Line-of-sight |
| MAD | Mean Absolute Deviation |
| NAT | Network Address Translation |
| NLOS | Non-line-of-sight |
| NOA | Notice of Absence |
| OPS | Opportunistic Power Save |
| PIN | Personal Identification Number |

| | |
|---|---|
| PSK | Pre-Shared Key |
| P2P | Peer to Peer |
| QoS | Quality of Service |
| RAM | Random Access Memory |
| VR | Virtual Reality |
| V2P | Vehicle to Person |
| V2V | Vehicle to Vehicle |
| WiFi | Wireless Fidelity |
| WPA | Wi-Fi Protected Access |
| WPS | Wi-Fi Protected Setup |

# ABSTRACT


DESIGN, IMPLEMENTATION AND A PILOT STUDY OF MOBILE FRAMEWORK FOR PEDESTRIAN SAFETY USING SMARTPHONE SENSORS

AAWESH MAN SHRESTHA

2018

Pedestrian distraction from smartphones is a serious social problem that caused an ever increasing number of fatalities especially as virtual reality (VR) games have gained popularity recently. In this thesis, we present the design, implementation, and a pilot study of WiPedCross, a WiFi direct-based pedestrian safety system that senses and evaluates a risk, and alerts accordingly the user to prevent traffic accidents. In order to develop a non-intrusive, accurate, and energy-efficient pedestrian safety system, a number of technical challenges are addressed: to enhance the positioning accuracy of the user for precise risk assessment, a map-matching algorithm based on a Hidden Markov Model is designed; to minimize energy consumption, an adaptive scheme is developed that dynamically activates the GPS module of a phone according to pedestrian walking speed and the locations of nearby crosswalks; to suppress false alarms, a novel algorithm is developed to accurately identify the user-phone-viewing activity so that collision probability assessment is triggered only when the pedestrian is walking while viewing his or her phone. The prototype of the proposed framework is implemented on an Android platform for a pilot study to evaluate feasibility, reliability, and validity of WiPedCross. Extensive experiments are performed in a parking lot and

the results demonstrate that WiPedCross assesses the collision probability efectively and provides warning to the user in a timely manner. The system modules of the proposed framework are expected to benefit numerous other pedestrian safety apps.

# Chapter 1

# Introduction

## 1.1 Motivation

The number of accidents concerning distracted pedestrians have increased substantially, especially due to recent release of Virtual Reality (VR) games such as Pokemon Go [3]. A study has shown that about one third of the pedestrians use mobile phones while crossing streets [2]. In fact, in 2015, 5,376 pedestrians were killed which accounts for an increase of 9.5% compared with pedestrian fatalities in 2014, and it was the highest number of fatalities since 1996 [5]. The statistics indicate that every 1.6 hours a pedestrian was killed, and was injured every 1.6 minutes. There is a pressing need for development of new technology to address this significant societal issue.

## 1.2 Limitations of Prior Art

With recent advances in mobile computing technologies for transportation [34][33][27][37], various approaches have been proposed to improve pedestrian safety. Image-based solutions utilize phone cameras and image processing techniques to detect vehicles posing danger to pedestrians [30]. However, these solutions raise the privacy issue and consume too much energy for running image processing algorithms. Dedicated Short Range Communication (DSRC)-based solutions use the 802.11p standard to enable communication of safety information between a car and a pedestrian [38][26]. However, implementing the DSRC protocol on a phone requires significant modifications to the host system, and specialized equipment is required to implement it for a vehicle. Some approaches utilize a backend server via a cellular network to analyze pedestrian safety [28]. Unfortunately, these approaches incur not only increased message delay, but high cost for exchanging data on the cellular network.

Recently, WiFi has been actively adopted to implement various transportation applications such as traffic monitoring [36][39], traffic management [35][32], localization [11], and pedestrian safety [12][22][16]. An effective pedestrian safety app, however, must meet the following conditions: (1) Non-intrusiveness: the app must operate seamlessly on off-the-shelf phones without requiring modifications to the original hardware and protocol stack; (2) Interactivity: both drivers and pedestrians must be alerted of hazardous situations for improved safety; (3) Sustainability: the energy consumption of the app must be minimized for extended operation time; (4) Independence: the app must run independently without relying on external servers and specialized hardware; (5) Timeliness: alert messages must be sent to drivers and pedestrians only when it is needed to minimize driver distraction. Unfortunately, we found that existing WiFi-based approaches do not meet one or more of the above conditions.

### 1.3 Proposed Approach

In this thesis, we present the design, implementation, and evaluation of WiPedCross, a stand-alone pedestrian safety app that accurately assesses the hazardous situations, and provides warning to both drivers and pedestrians in a timely and energy-efficient manner without requiring modifications to the host mobile system. The proposed system utilizes the WiFi Direct technology to allow drivers and pedestrians to exchange safety information, assess the collision probability, and send alert messages to prevent accidents.The details of the WiFi Direct technology is presented in Chapter 3. Although the concept itself is quite simple, a number of technical challenges must be addressed to enable a fully functioning pedestrian safety app. Positioning of drivers and pedestrians is one of the key components of WiPedCross. Unfortunately, however, in urban areas with a large number of skyscrapers where most pedestrian accidents occur, the accuracy of a phone GPS module is substantially degraded resulting in significant location errors. To address this problem, a map matching algorithm based on a Hidden Markov Model and human walking speed is developed to accurately identify a sidewalk segment that the pedestrian is estimated to be located, which is then used to either eliminate potential location outliers, or perform projection of erroneous locations into appropriate points on the sidewalk segment. To save power consumption of a phone, especially concentrating on the significant energy consumption by the energy-hungry GPS module, a dynamic approach is developed to activate the GPS module adaptively depending on the estimated time that the user is expected to be geographically close to a nearby crosswalk. Another challenge is to ensure that the user is alerted only when he or she is viewing their phone while walking to prevent unnecessary interruption to the pedestrian, and minimize driver distraction. Motivated by the observation that when users view their phones while walking, they tend to try to minimize the shaking of the phone to better read the screen, type a message for texting, or watch videos, an algorithm is developed that accurately detects the user-phone-viewing activity.

Consequently, by integrating all the system components that are designed to address numerous challenges, the collision probability is estimated to assess the pedestrian safety level and alert accordingly the user. It should be noted that no apps provide 100% pedestrian safety because it is impossible to perfectly capture the pedestrian's intention, *i.e.,* what the user will do, *e.g.,* continue to walk, look up, stop, and so on. Under this challenging uncertainty, the proposed framework is introduced as a state-of-the-art precautionary tool designed based on the strong interplay of numerous novel system components to significantly reduce pedestrian traffic accidents.

### 1.4 Thesis Organization

This thesis is organized as follows. Chapter 2 provides a literature review on related approaches designed for improving pedestrian safety followed by the technical details of Wi-Fi Direct in Chapter 3. In Chapter 4 we describe an overview of the proposed framework followed by the details of each system component. The performance of the proposed app is evaluated in Chapter 5. We then conclude in Chapter 6.

# Chapter 2

## Related Work

Wang *et al.* developed an app that uses the rear camera of a phone to detect the dangerous situation, and alert the pedestrian [30]. A machine-learning-based image processing algorithm was designed to capture approaching cars for pedestrian safety assessment. This approach, however, raises the privacy issue as it takes photos of cars, and also suffers from the power consumption problem as the image processing algorithm consumes a lot of energy.

Specialized equipment was designed for pedestrian safety. Sensors were adhered to the pedestrians' shoes to find whether the pedestrian is crossing at a crosswalk by detecting the slope between the sidewalk and the roadway [19]. An electronic transponder was attached to the pedestrian's body to determine if the pedestrian is visible or not [13]. These specialized equipment, however, prohibits widespread adoption of the technology.

A cellular network was utilized to allow pedestrians to communicate with cars (*i.e.,* car-mounted navigation system) via a backend server [28][14] [21]. Using a cellular network, however, not only incurs high cost but also results in higher message delay compared with the direct peer to peer communication. Dedicated Short Range Communication (DSRC) is a wireless communication standard specifically designed for vehicle to vehicle communication (V2V). Researchers utilized DSRC as a means to enable vehicle to person (V2P) communication for pedestrian safety [38][26][22]. However, implementing DSRC on a phone requires significant modifications to the host system firmware, and extra device support is needed to operate DSRC for vehicles.

WiFi has been actively considered as an appropriate alternative technology to enable vehicle to pedestrian communication for pedestrian safety [8][12][16][22]. In particular, WiHonk is quite close to our work [12]. However, it was based on the modification of the beacon frame of IEEE 802.11 which requires the root privilege that makes it difficult for common use. Additionally, no consideration was presented on when to exchange messages with cars, potentially resulting in unnecessary network bottleneck [25]. WiSafe is another WiFi-based pedestrian safety system that is close to our work [16]. Our work is different in that the system design involves both the driver and pedestrian while WiSafe utilizes only one-way communication from a pedestrian to cars. Compared with existing WiFi-based approaches, the proposed work is distinctive in that it takes a holistic approach by providing solutions for practical issues on energy-efficiency, positioning accuracy, context awareness, collision probability analysis, and integrating them together to design the first fully-functioning pedestrian safety framework.

# Chapter 3

# WiFi Direct

## 3.1 Introduction

Recently, Wi-Fi Alliance has defined a new technology called Wi-Fi Direct to enhance the way the devices communicate via Wi-Fi [10]. In a Wi-Fi Direct technology, the devices must find each other through scanning process and then can connect to each other forming a group of devices. This is not the first technology to enable the device to device connectivity. IEEE 802.11 standard already made it possible through *ad hoc* network [10]. However, the *ad hoc* could not flourish in the market because it neither had optimized power saving technology nor best Quality of Service (QoS) capabilities. The devices which possess Wi-Fi Direct technology are referred to as P2P Devices and clusters of connected P2P devices are called P2P group. Functionally, these P2P groups are similar to old Wi-Fi networks. Typically, in conventional Wi-Fi networks, the devices connect to each other through Access Points (APs) as shown in Figure 3.1. However, in case of Wi-Fi Direct, the P2P devices automatically take the role of either an AP or a client where both roles have a variety of functionalities. Wi-Fi Direct communication eliminates the need for an AP [20] where the roles of client and server are specified dynamically through the negotiation. Each Wi-Fi Direct devices are equipped with the implementation of both the logical role of the client as well as the role of an AP. There are several scenarios where a single device has to execute both roles as shown in Figure 3.3 which is referred to as the concurrent mode [6]. Figure 3.1 shows the difference between the Wi-Fi and the Wi-Fi Direct. The left part of the figure shows the possibility of two-way communication between the P2P GO (Peer to Peer Group Owner) and the P2P Clients directly without the need for an external AP. However, the right part of the figure shows that in a conventional Wi-Fi network, the devices must connect through an AP. Hence, Wi-Fi Direct is more usable and beneficial for peer to peer connection.

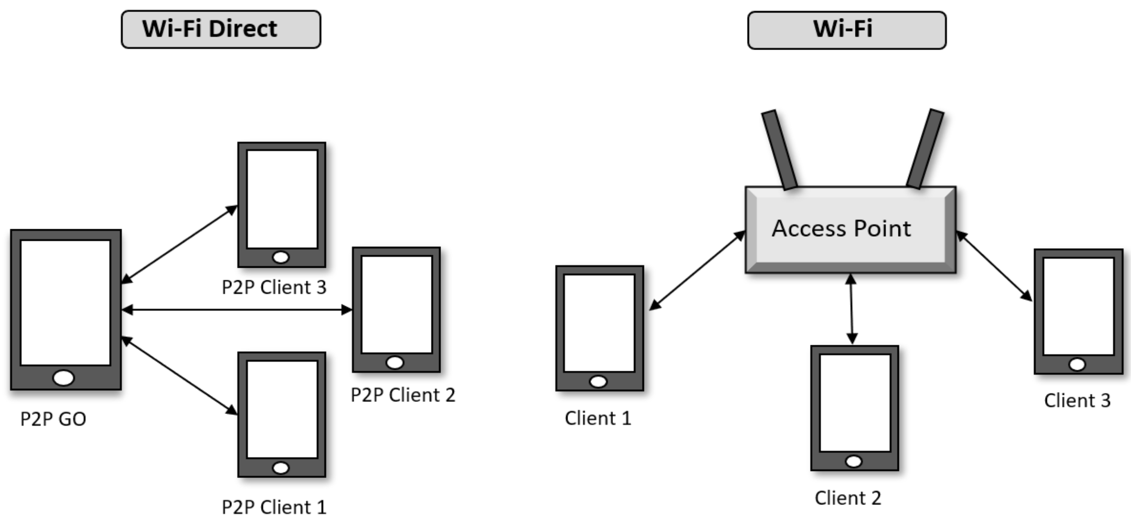


Figure 3.1: Wi-Fi Direct vs Wi-Fi.

## 3.2 Technical overview

### 3.2.1 Architecture

A Wi-Fi Direct communication is possible only if a P2P group has been formed. An owner device of a P2P Group called P2P GO must have implemented an AP-like functionality but not an AP itself and the client devices (rest of the devices in a group) called P2P clients simply join the group. These P2P groups are similar to a conventional Wi-Fi network which can be joined only when the P2P group has been formed. The legacy clients which are not 802.11b-only devices can also join P2P groups. The P2P GO is visible to the legacy clients like an Access Point (AP) in a conventional Wi-Fi network but they cannot use all the functionality that a Wi-Fi Direct device can support. Wi-Fi Direct supports different architectural deployments because it can act both as a client as well as an AP as shown in Figure 3.2. The top part of the figure shows the possibility of forming two P2P groups. In the group 1, a mobile phone acts as a P2P GO and two laptops act as P2P clients sharing a 3G internet connection from a tower. The second laptop which is a P2P client for the group 1 now forms another group that is, group 2. In this case, the second laptop acts as a P2P GO and printer as a P2P client. The Wi-Fi interface is time shared by the second laptop to go into the concurrent mode, that is, it acts both as a P2P client as well as P2P GO which typically alternates between the roles. In the lower part of the Figure 3.2, a laptop accesses internet through a router and at the same time it also streams the content to a TV set. In this case, the laptop behaves as a P2P GO forming a P2P Group. Only a P2P GO is permitted to cross-connect from one group to external group and this cross-connection is only permitted to a group owner. Network Address Translation (NAT) must be implemented at the network layer to make this type of connection possible. Clients cannot transmit messages to each other directly. The transmission occurs only between the group owner and the clients. Group owners can also be referred to as hosts. In a P2P group, once a device has been assigned as a P2P GO, its role cannot be changed to the P2P client or vice-versa. The group will be dismissed and the connection will be lost and has to be re-established if a P2P GO leaves a P2P group.

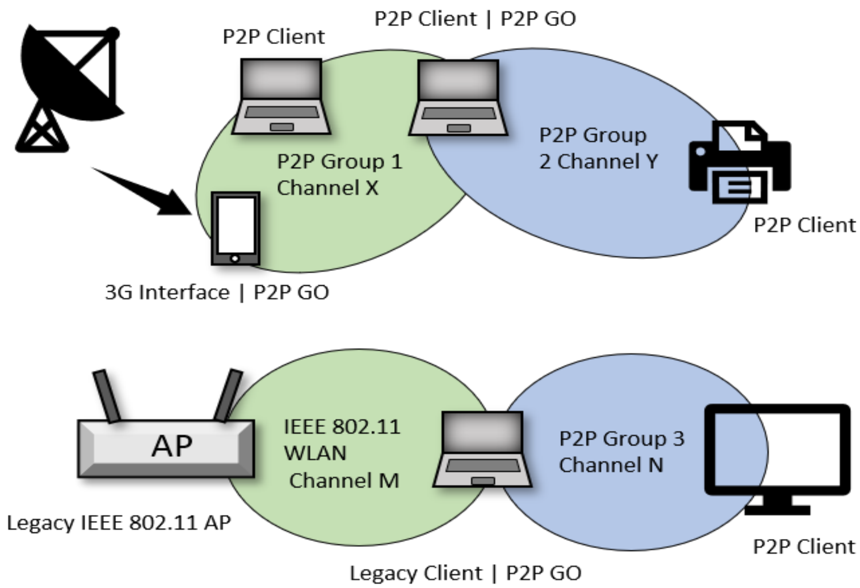


Figure 3.2: Wi-Fi Direct connection scenarios

### 3.2.2 Group formation

Several factors affect the way a P2P group can be formed. In some group formation technique, the role of P2P GO has to be negotiated while in some cases a group owner is selected autonomously. It also depends on whether some security information has been shared already in the previous connection or not. While forming a group, firstly the group owner is either negotiated or selected and then a session is established using proper credentials. There are three types of group formation methods which are described briefly as follows:

- Standard

  In this method, the P2P devices initially find each other and after that consult to end up a P2P GO. Firstly, a conventional Wi-Fi scan is performed by the Wi-Fi Direct devices to find the existing P2P groups. After finding the existing P2P groups, an algorithm is executed to discover the available services. One of the channels (operating in 2.4 GHz band) out of channels 1, 6 or 11 is selected to listen by the P2P devices. There are two states, one is searching and another is listening state. The P2P devices keep on changing these states time and again. In search state, active scanning of the channel is performed by sending probe requests in every channel and in the listening state, the P2P devices listen for the probe requests and respond with the probe response. A P2P device typically spends between 100ms and 300ms in each state randomly. The amount of time to spend in each state is dependent on the implementation that either reduces the service discovery time or enhances the energy savings. Within the service discovery phase, the P2P devices must have found each other and viewed the list of available services. After that, a GO Negotiation phase will commence whose goal is to determine a GO in a group. A three-way handshake which consists of Request/Response/Confirmation is used to implement the GO Negotiation phase. In the handshake process, all the P2P devices will send their intent values to become a P2P GO. The one with the highest value will become a P2P GO. If two or more devices have a desire to be a P2P GO, they end up sending the same numerical value of GO intent. This conflict is removed by attaching a special bit to the GO Negotiation request, setting it arbitrarily each time a request is sent. By this time, the P2P devices will have discovered each other, formed a group and agreed upon the P2P GO. After that, a secure connection is established using Wi-Fi Protected Setup (WPS) which is denoted as the WPS provisioning phase. After that, a DHCP is employed to set up the IP configuration which is shown as 'Address config. phase' in Figure 3.3.

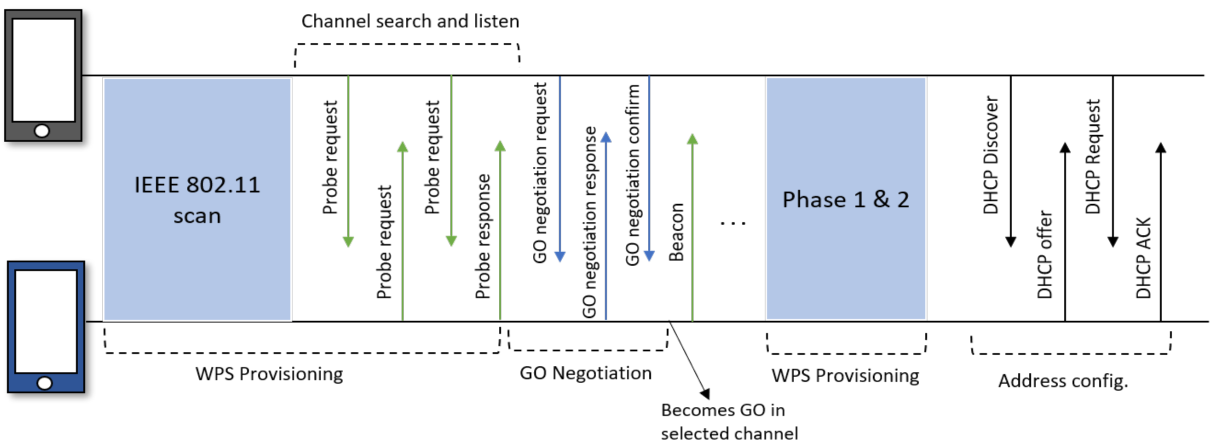


Figure 3.3: Wi-Fi Direct operation.

- Autonomous

  This method of group formation is very simple where a device immediately becomes a P2P GO and starts to beacon on a channel. Rest of the devices can find the formed group using conventional scanning method. After that, they can specifically continue with the security and DHCP stages. Here, the discovery phase is rearranged as compared to the previous strategy by removing the alternating phases: search and listen phases. Also, there is no need of GO negotiation phase as the devices anonymously become a P2P GO.

- Persistent

  In this method of group formation, they can announce a group as a persistent by employing a flag in the Beacon frames. The group contains the information about the network credentials, the P2P GO and the clients assigned which will be useful in the future connections within the P2P group. Once the services have been discovered, if it is found that it has already created such group with the associated peer in the past, they can quickly re-instantiate the group using a two-way handshake mechanism. The main advantage of this kind of group formation is that the devices do not need to go through the GO negotiation phase. Instead, a two-way handshake procedure is employed. Another advantage is that theere is reduction in the time spent in WPS provisioning as this technique allows the re-usability of previously stored information.

### 3.2.3 Security

Once a group has been formed and GO has been negotiated, WPS provisioning phase starts. All the Wi-Fi Direct devices implement WPS which supports a secure connection without user intervention. Either a PIN is used in the client or WPS push button is used to establish a secure WPS connection. An internal Registrar is implemented by a P2P GO and Enrollee is implemented by the P2P clients in the WPS security. WPS operation is composed of two phases. The network credentials (security keys) are generated and issued to the Enrollee by the Registrar in the first phase. Based on WPA-2 security, WPS employs the Advanced Encryption Standard

(AES) as an encryption technology and Counter Mode with Cipher Block Chaining Message Authentication Code Protocol (CCMP) as a cipher which is highly secure and hard to break. For the mutual authentication, it uses arbitrarily obtained Pre-Shared Key (PSK). In second phase, the Enrollee (P2P client) disconnects and restablishes the connection using its latest authentication credentials. Due to this reason, the first phase of WPS provisioning can be skipped and can directly perform the authentication if two devices are already equipped with the necessary network credentials (like in persistent group formation).

### 3.3 Benefits of Wi-Fi Direct

- Wi-Fi Direct devices are secured with Wi-Fi Protected Setup (WPS) implementation which supports a secure connection with neglegible user interference where it uses a PIN or is required to push a button on both devices.

- Wi-Fi Direct devices use power saving techniques such as Notice of Absence (NOA) protocol and the Opportunistic Power Save (OPS) protocol [10]. In the OPS method, a P2P GO can minimize power consumption when all the joined clients are not awake while in the Notice of Absence method, a P2P GO announces the time interval which is also known as the absence period where the associated clients are prohibited from using the associated channel.

- Wi-Fi Direct can connect anytime, anywhere easily even with the existing traditional Wi-Fi devices. The connection process is so simple that it may even replace Bluetooth in some cases. A user can view a list of devices and the services it has to provide before establishing a connection.

- Unlike conventional Wi-Fi, Wi-Fi Direct is able to employ service lookup at the data link layer before establishing a P2P group. A device may join a group only if a group offers the services it is interested in. It is a huge progress from a conventional Wi-Fi network, where the clients are interested in the internet connectivity.

- There is no need of a wireless router to establish a wireless connection.

- Wi-Fi Direct is much easier to setup as compared to ad-hoc which offers data transfer rate up to 250 Mbps. File sharing applications in smartphones can make use of it to utilize its faster data transfer rate.

# Chapter 4

# System Design

This chapter presents an overview of the proposed framework, followed by the details of main components of the framework.

## 4.1 System Overview

The basic mechanism of WiPedCross is simple: If a pedestrian is walking while viewing his or her phone, and if the pedestrian is geographically close to a crosswalk, the probability of collision is calculated by exchanging messages with surrounding vehicles using WiFi Direct. Based on this safety assessment, the pedestrian, the driver(s), or both are alerted to avoid collision. In order to prevent driver distraction, WiPedCross ensures that a number of alert messages are given to the pedestrian first, and only when those messages are ignored, the driver is alerted. Developing this seemingly simple system, however, poses a number of challenges.

- Alerting the users timely and accurately depends heavily on the positioning accuracy. However, it is not trivial to provide accurate localization with a smartphone especially in urban areas.
- The GPS module of a phone consumes too much power if it is turned on continuously.
- Detecting the user-phone-viewing event is not straightforward as existing image processing-based approaches based on face detection are computationally too

expensive.

- Alert messages should be sent in a timely manner based on accurate collision assessment to not send unwanted messages to prevent driving interruption.

- WiFi Direct does not support n-to-n communication while there may be multiple pedestrians who want to send warning messages to surrounding vehicles.

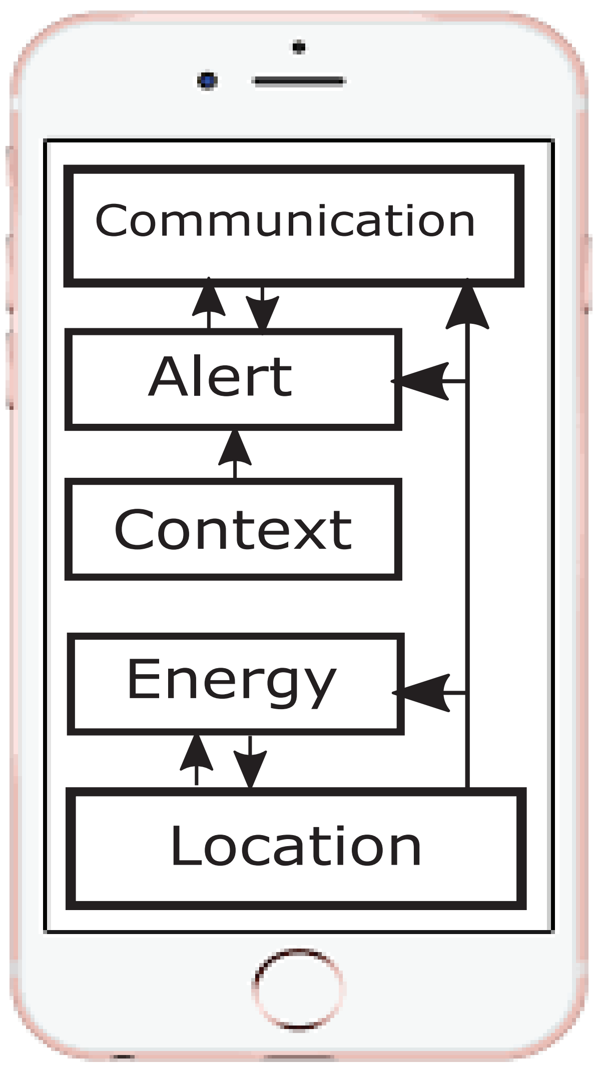


Figure 4.1: System overview.

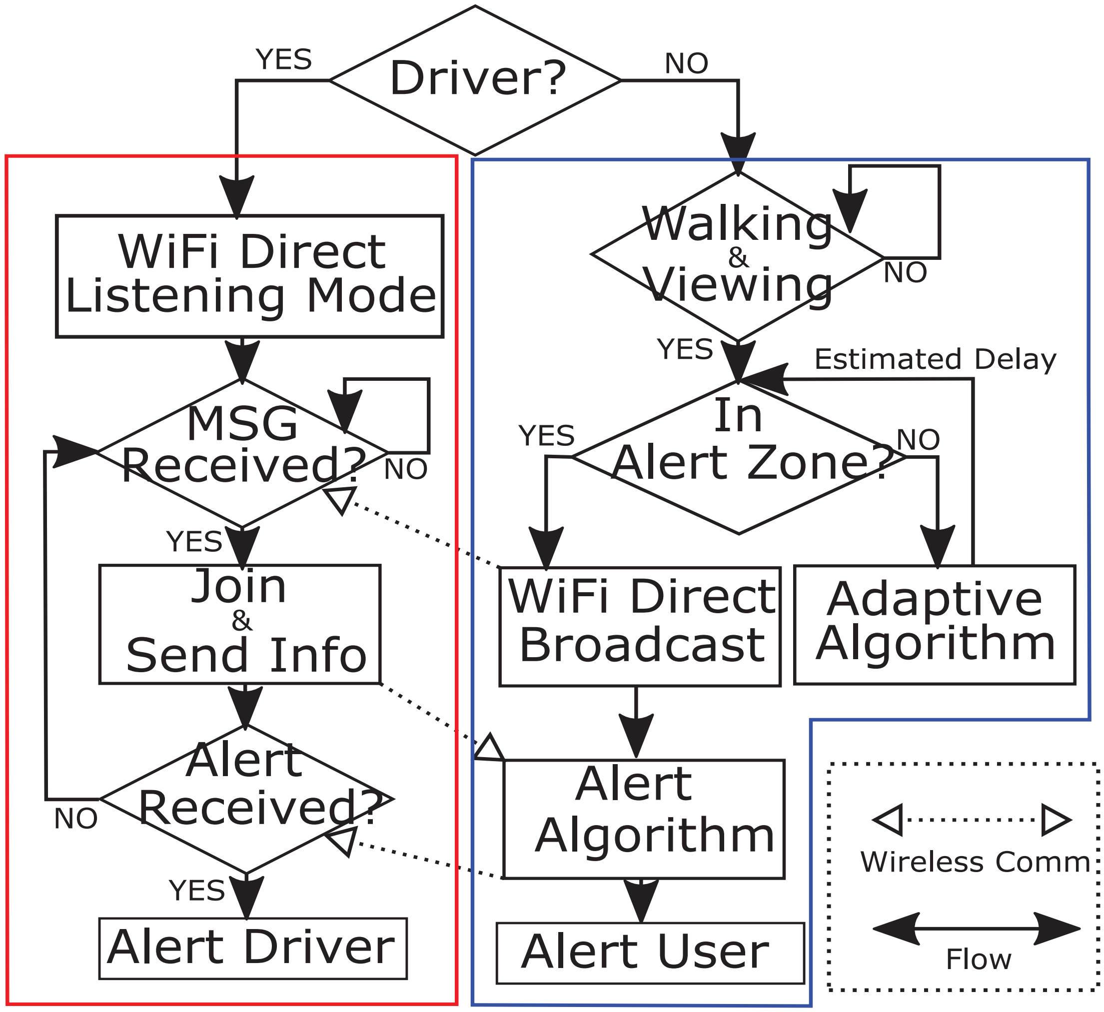


Figure 4.2: Component diagram.

The proposed app consists of five modules that are designed to address the afore-mentioned challenges, namely Location, Energy, Context, Alert, and Communication modules (Figure 4.1). While the Location module is primarily responsible for providing location information of the user to other modules, it is also used to improve the positioning accuracy especially in urban areas. The Energy module, as shown in Figure 4.1 interacts with the Location module to adaptively control the operation of the GPS module to reduce power consumption. The Context module detects efficiently the user-phone-viewing event to activate the proposed system only when the user is viewing his/her phone while walking. The Alert module is the heart of the proposed system that determines when and to which user (driver or pedestrian) an alert message will be sent based on the collision probability analysis. As the figure shows, the Alert module relies on the Communication module that implements reliable wireless communication between a pedestrian and a driver based on WiFi Direct. The communication module is designed such that n-to-n communication is enabled for WiFi Direct.

A flowchart (Figure 4.1) better explains the general usage of the proposed system. It starts with identifying the user type, *i.e.,* whether the user is a driver or a non-driver by running a driver phone detection algorithm which has been researched extensively. Numerous works have been proposed for driver phone detection [31]. If the user type is determined to be a driver, the app opens a network port for WiFi Direct and waits for incoming messages from pedestrians. More specifically, the 'autonomous' mode of WiFi Direct [10] is used to ensure that the pedestrian immediately becomes the P2P Group Owner and surrounding vehicles scan for network to join the group. Once a message is received from a pedestrian, the driver phone joins the group created by the pedestrian and sends necessary information to the pedestrian so that the pedestrian can assess the safety level by performing the collision probability analysis. Consequently, the driver receives an alert message depending on the result of the assessment.

If it is determined that the user is a pedestrian, the proposed system continuously checks for the user-phone-viewing event, *i.e.,* the user is walking while viewing his/her phone. In particular, even if the event is detected, when the user is not geographically close to a crosswalk, the GPS module is put into an inactive mode to save power consumption. On the other hand, if the user is close to a crosswalk and is viewing his or her phone while walking, the pedestrian immediately advertises a message to surrounding vehicles to create a WiFi Direct P2P group with them. As a result, the pedestrian is able to collect necessary information from surrounding vehicles via WiFi Direct. Based on the collected information, the pedestrian phone performs the collision probability analysis to determine whether or not to send an alert message to himself/herself, the driver(s), or both.

### 4.2 Improving Positioning Accuracy

Pedestrian accidents occur frequently in urban areas. The positioning accuracy of GPS is, however, significantly degraded in urban canyons due to multipath and non-line-of-sight effects constraining the use of pedestrian safety apps. Using the GPS module of Samsung Galaxy S6, we collected GPS locations in a metropolitan area, where skyscrapers are concentrated. Figure 4.3 depicts the measured GPS locations and the ground truth trajectory (green arrow). It can be easily noted that the location errors were significant compared with the ground truth trajectory. This section explains the Location module of WiPedCross that is designed to eliminate location outliers, so that unwanted warning messages are not sent to pedestrians and drivers.

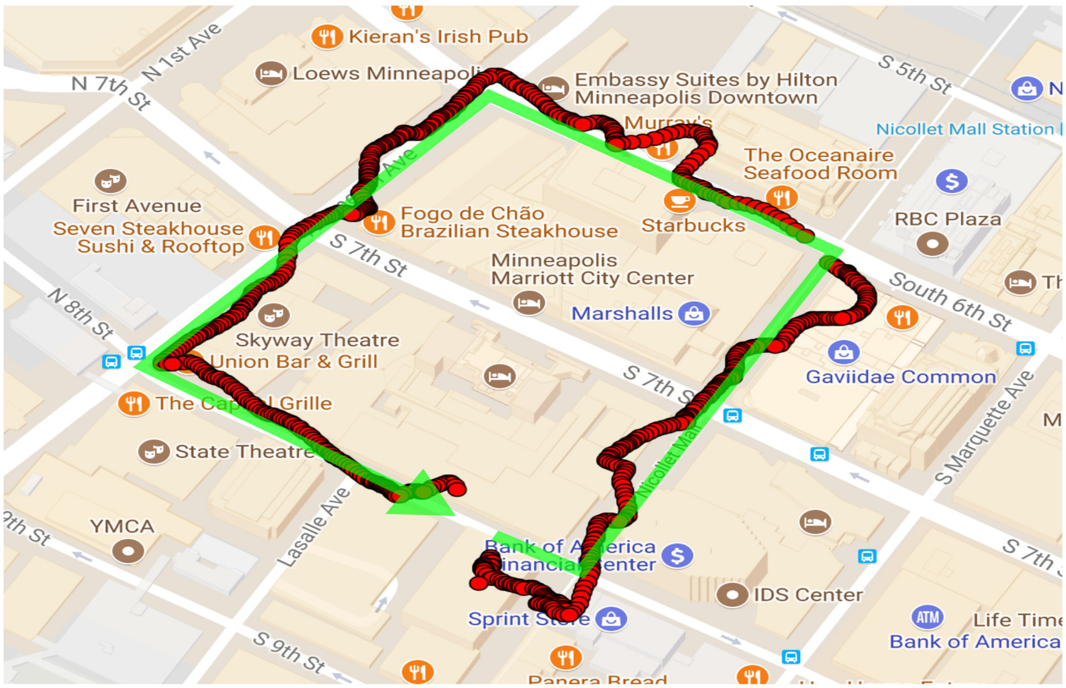


Figure 4.3: Location errors in metropolitan area.

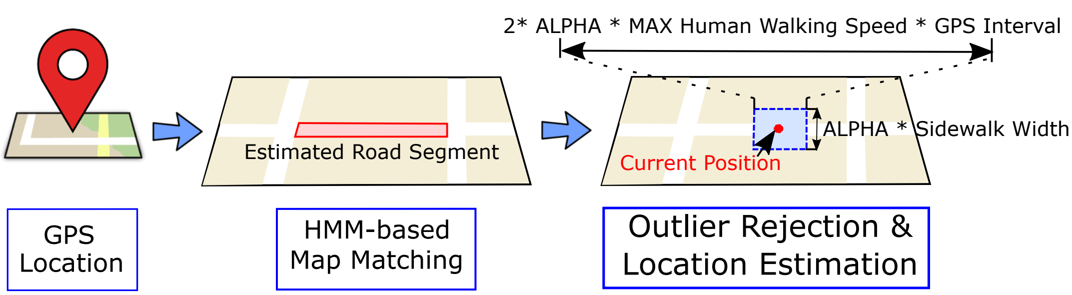


Figure 4.4: Overview of Location module.

The Location module aims to effectively identify location outliers and replace them with estimated user locations. Figure 4.4 shows an overview of the module. The key idea is to leverage our prior knowledge on the user location, *i.e.,* it is confined within a specific segment of a sidewalk. More precisely, given an obtained GPS location, this module identifies the most probable sidewalk segment that would contain the user location by using a Hidden Markov Model-based map matching algorithm. Once the sidewalk segment is estimated, an outlier rejection mechanism is applied to identify location outliers. Such outliers are either removed or replaced with the projected position on the sidewalk segment depending on the degree of location error.

An algorithm that estimates the current sidewalk segment is described. The design of this algorithm is motivated by recent map matching algorithms [23][7]. The proposed algorithm formulates the problem of identifying the most probable sidewalk segment using a Hidden Markov Model (HMM). More specifically, define a set of states $S = \{r_1, r_2, ..., r_N\}$ where each state represents a sidewalk segment, where $N$ is the total number of states. To reduce computational overhead, only the road segments that are in the proximity of the current user location are considered as the states. Now based on HMM, the algorithm finds the most probable segment denoted by $r_i \in S, 1 \leq i \leq N$ given an observation $Z_t$, *i.e.,* a set of latitude/lontitude measurements in a sliding window at time $t$.

More formally, a HMM is modeled as $\lambda = (S, Z_t, A, B, \pi)$. $S$ is the state set, and $Z_t$ is an observation that is represented as a sliding window consisting of GPS measurements of size $\omega$ at time $t$, *i.e.,* $Z_t = \{z_1, ...z_\omega\}$, where $z_j$ is a GPS measurement at time $j$. $A$ is the observation probabilities denoted by $P(Z_t|r_i), 1 \leq i \leq N$ which define the likelihood that the pedestrian is actually on each segment $r_i$. $B$ is the transition probabilities $P(r_j|r_i), i \neq j(i, j = 1...N)$ that represent the likelihood of the pedestrian moving from one segment $r_i$ to another $r_j$. $\pi$ is the initial state probabilities which are defined as $P(Z_1|r_i), i = 1, ..., N$. In contrast to [23][7], the key idea is to use a set of prior GPS measurements $Z_t$ as an observation motivated by our findings that using a single GPS location as an observation incurs significant errors in identifying a sidewalk segment.

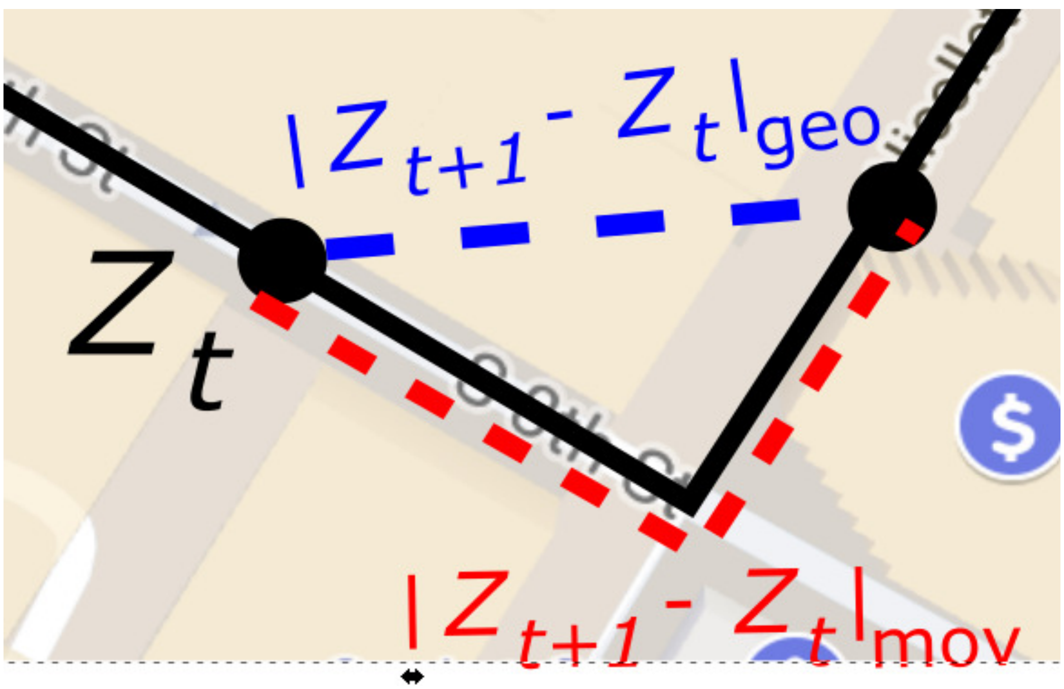


Figure 4.5: Illustration of moving distance and geodetic distance.

To determine the most probable state (*i.e.,* a sidewalk segment), we need to model $A$, $B$, and $\pi$. More specifically, in modeling $A$, we leverage the observation that a measured location $z_t$ that is far from the true road segment $r_i$ is less likely [23].

Thus, the probability distribution of the geodetic distance between $z_t$ and $z_{t,i}$ denoted by $|z_t - z_{t,i}|_{geo}$ where $z_{t,i}$ is the closest point on $r_i$ from $z_t$, is used to represent $P(z_t|r_i)$. The geodetic distance $|z_t - z_{t,i}|_{geo}$ specifies the GPS positioning error, which can be modeled as zero-mean Gaussian as follows.

$$P(z_t|r_i) = \frac{1}{\sqrt{2\pi}\sigma_z} e^{-0.5(\frac{|z_t - z_{t,i}|_{geo}}{\sigma_z})^2}, \tag{4.1}$$

where $\sigma_z$ is the standard deviation of GPS measurements, which can be obtained empirically. Now considering a set of GPS measurements in a sliding window, the observation probabilities $P(Z_t|r_i)$ is defined as follows.

$$P(Z_t|r_i) = \frac{\sum_{t=1}^{\omega} P(z_t|r_i)}{\omega}. \tag{4.2}$$

Next the transition probabilities $B$ given two observations $Z_t$ and $Z_{t+1}$ are defined as follows. Note that there is a new GPS point $z_{t+1} \in Z_{t+1}$ compared with $Z_t$. Given two GPS points $z_t$ and $z_{t+1}$, let us define the 'moving distance' as the distance between the two points along the shortest sidewalk trajectory. Figure 4.5 shows the geodetic distance $|z_{t+1} - z_t|_{geo}$ and the moving distance $|z_{t+1} - z_t|_{mov}$ between $z_t$ and $z_{t+1}$. Newson and Krumm found that a transition between road segments would occur less likely when the moving distance is close to the geodetic distance [23]. However, we note that this does not apply appropriately to the pedestrian walking scenario where the pedestrian moves much shorter distance than a car between two GPS measurements. Thus, rather than using the two immediately subsequent GPS points to calculate the moving distance and the geodetic distance, two points $z_t$ and $z_{t+\epsilon}$ with some interval $\epsilon$ are selected. More specifically, given the two points, the difference between the moving distance and the geodetic distance is defined as $\delta = ||z_{t+\epsilon} - z_t|_{mov} - |z_{t+\epsilon} - z_t|_{geo}|$. It was shown that $\delta$ follows an exponential distribution [23]. Thus,

$$p(\delta) = \frac{1}{\beta} e^{\frac{-\delta}{\beta}}. \tag{4.3}$$

Here $\beta$ is a system parameter that a larger value represents more tolerance to non-direct paths. Consequently, we can define $B$ as the following.

$$P(r_j|r_i) \approx p(\delta). \tag{4.4}$$

We also note that using Eqs. 4.1 and 4.2, $\pi = P(Z_1|r_i)$ immediately follows.

Once the most probable sidewalk segment is determined, given that segment, previous GPS measurements, and the maximum brisk walking speed [1], we calculate a rectangular region (shown as a dotted blue area in Figure 4.4). The width of the region is defined as '$\alpha\cdot$ (max walking speed) $\cdot$ (GPS interval)', where $\alpha$ is a parameter that adjusts tolerance to location error, and the height of it is '$\alpha\cdot$ (width of sidewalk)'. Consequently, measured GPS locations that are outside the region are rejected and replaced by the estimated location, which is defined as the projected location on the identified sidewalk segment'. Furthermore, if the measured GPS point is far from the region greater than a threshold, that GPS point is not considered. For our experiments, we used $\alpha = 2$, and the threshold was 15m. The threshold of 15m was selected to include enough GPS locations for Map Matching algorithm while maintaining the location accuracy. Selection of lower threshold value results in the exclusion of significant GPS locations which does not include all the necessary points resulting in inaccurate location and hence the inaccurate sidewalk segment.

### 4.3 Improving Energy Efficiency

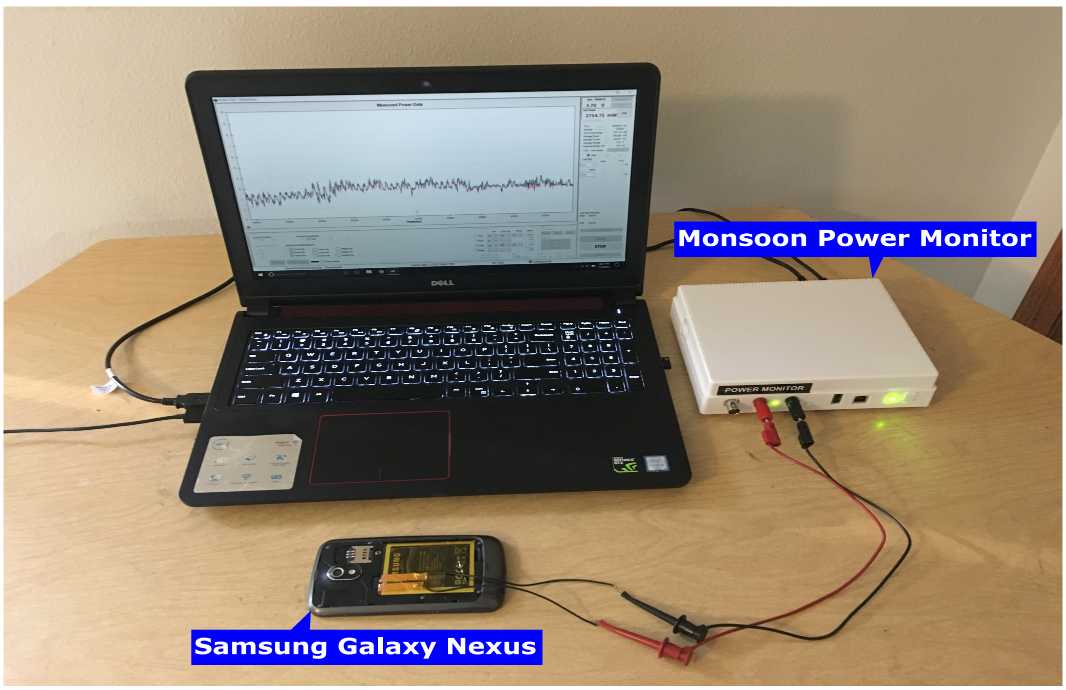


Figure 4.6: Experimental setup for power consumption measurement.

The GPS module of a smartphone is one of the major power consumers. Keeping the module on continuously will drain the battery very quickly. Experiments were performed to characterize the power consumption of the GPS module. More specifically, we used the Monsoon power monitor and Samsung Galaxy Nexus to

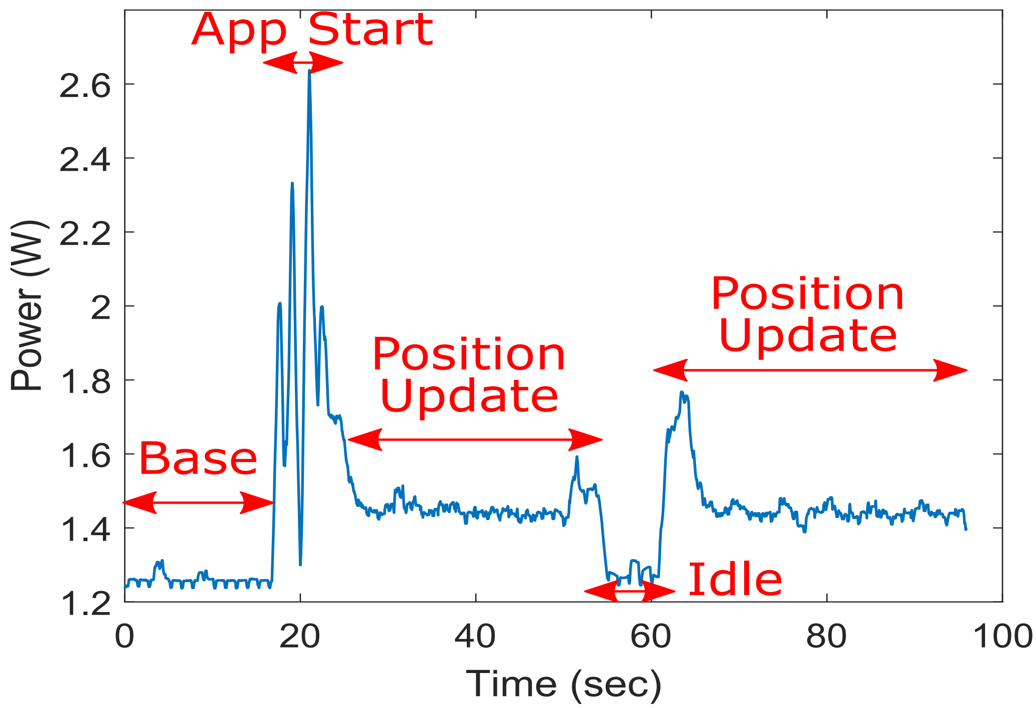


Figure 4.7: Power consumption of GPS module of smartphone.

measure power consumption. The experimental setup is depicted in Figure 4.6.

Figure 4.7 displays the power consumption of the smartphone GPS module. As it is shown, when WiPedCross was started, a large amount of power was drawn to load, process, and display the app on the screen. It was also observed that the periodic position update of the GPS module used a lot of energy compared with when it was in the idle mode. These results indicate that if we put the GPS module into the sleep mode when it is not needed, significant energy savings can be achieved.

To explain the Energy module, a term *alert zone* must be defined. The alert zone is a set of GPS points for which the geodetic distance to the nearest crosswalk is smaller than a certain value, which is a system parameter that can be adjusted based on the degree of GPS positioning error. Now the key idea of the Energy module is simply to put the GPS module adaptively into sleep mode if the user is not located within an alert zone. More precisely, the Energy module estimates the time for the user to arrive at the nearest alert zone, *i.e.,* the estimated time is calculated as $\frac{d}{v_{max}}$ where $d$ is the geodetic distance between the current user position and the nearest alert zone, and $v_{max}$ is the maximum brisk human walking speed [1]. The GPS module is then put into sleep mode until this estimated time is expired. This design decision of calculating the distance to the nearest alert zone is to ensure maximum pedestrian safety as we do not know which crosswalk the user will use. However, it should be noted that even if the user does not use the determined crosswalk (*e.g.,* walking away from the determined crosswalk), as soon as the estimated time is expired, the Energy module will quickly recalculate the distance to a nearest alert zone and put the GPS module into sleep mode, thus not much affecting the energy efficiency.

A notable observation is, however, that using the maximum human walking speed in estimating the user arrival time might cause the GPS module to turn on too early. The Energy module is thus designed such that the user can use his/her actual walking speed that is estimated based on the histogram of the actual user walking speed obtained from the Location module. More specifically, the user can select the walking speed at different percentile values from the histogram to adjust the balance between the accuracy of the estimated user arrival time and energy efficiency.

## 4.4 Detecting Pedestrian Phone Use

It has been reported that 56% of pedestrian phone use is telephone communication [4], which can be easily detected by using the proximity sensor of a smartphone. However, other phone use involving a phone-viewing activity, *e.g.,* texting, watching video, or reading email messages, which takes 21% of the pedestrian phone use is very tricky to capture especially when such activity does not incur user interactions like tapping on the phone. The Context module is designed to detect the user phone-viewing event efficiently to avoid sending false alert messages to drivers and pedestrians. A straightforward method to detect the phone-viewing activity is to utilize the phone camera to recognize the user face. This approach, however, raises privacy issues and consumes a lot of energy for running heavy-weighted image processing algorithms.

The key idea to detect the phone-viewing activity without using the phone camera is based on the observation that when users view their phones while walking, they tend to try to minimize the shaking of the phone to better read the screen, type a message for texting, or watch videos. So the Context module examines the variance of the acceleration magnitude of the phone to quantify the shaking of the phone and use it as an indicator to determine whether the user is viewing the phone while walking. More specifically, given accelerometer data $(a_x, a_y, a_z)$ of a phone in $x$, $y$, and $z$ directions, random noise is removed, and we obtain filtered accelerometer data denoted by $\hat{a_x}, \hat{a_y}, \hat{a_z}$. The magnitude of the acceleration vector $m$ is then calculated as follows.

$$m = \sqrt{\hat{a_x}^2 + \hat{a_y}^2 + \hat{a_z}^2}. \tag{4.5}$$

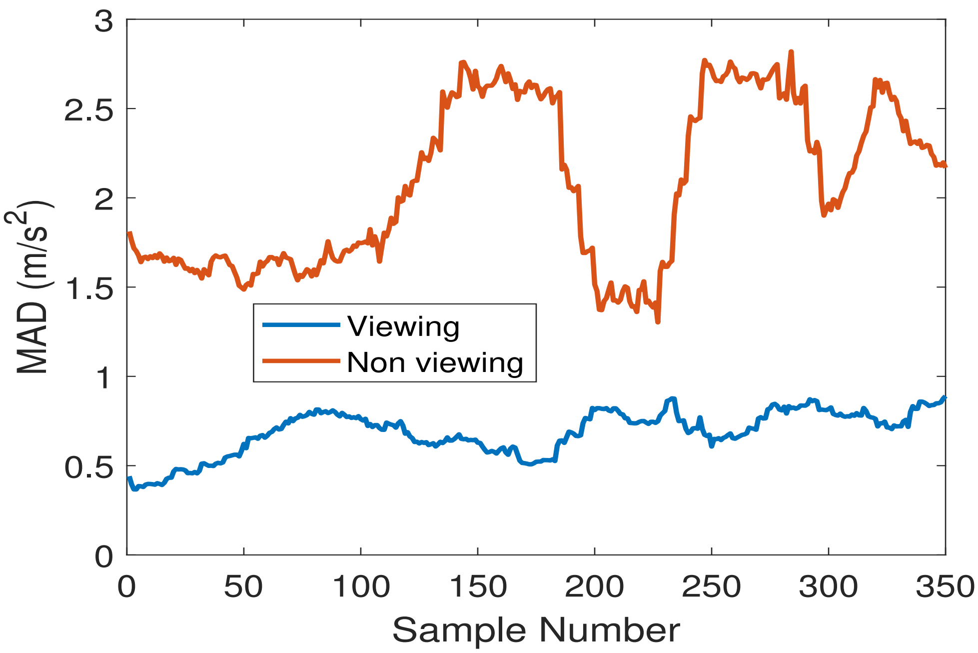


Figure 4.8: Sliding window of 10 sec.

We define a sliding window $W$ of size $\phi$ that saves a set of acceleration magnitude values, *i.e.*, $W = \{m_1, m_2, ...m_\phi\}$. For each sliding window, we calculate mean absolute deviation (MAD) that represents the degree of shaking as follows.

$$\frac{1}{\omega}\sum_{i=1}^{\omega} |x_i - \mathrm{mean}(X)|. \tag{4.6}$$

Proof-of-concept experiments were conducted to understand the feasibility of this approach. Five human subjects walked with and without viewing their phones. Figures 4.8 and 4.9 show the results for different sliding window sizes of 10sec and

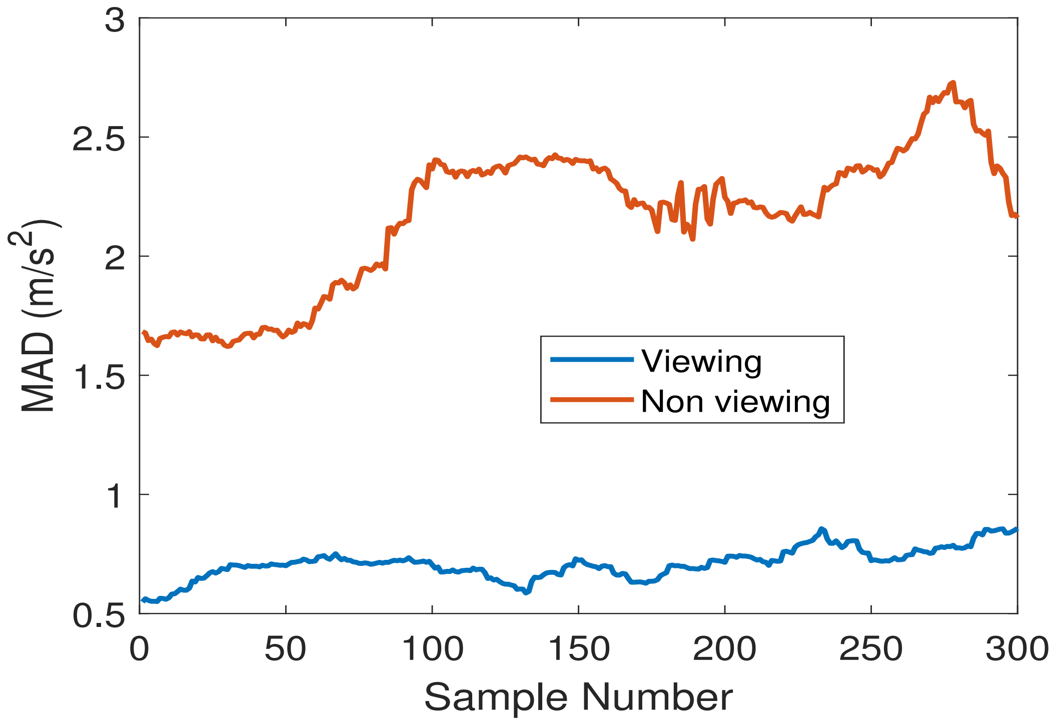


Figure 4.9: Sliding window of 20 sec.

20sec, respectively. As it can be noted, the variance of MAD for the phone-viewing scenario was significantly smaller than when the user is not viewing his or her phone while walking. The results also indicate that the accuracy of event detection depends on the window size.

### 4.5 Alerting the User

Sending an alert message to an appropriate user in a timely manner is crucial to not disturb the user phone use and to prevent driver distraction. The Alert module is designed to calculate the collision probability, based on which to determine when and to whom to send an alert message.

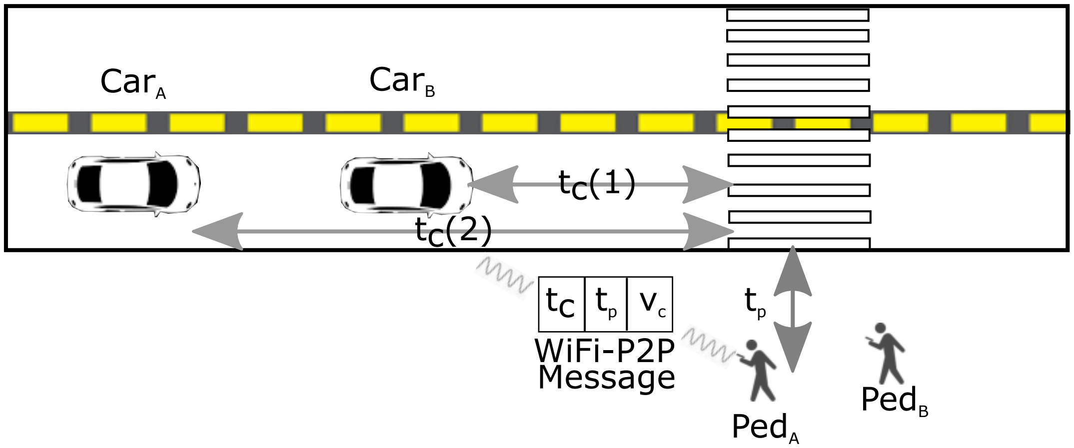


Figure 4.10: Overview of Alert module.

The Alert module is activated when the user enters an alert zone, and computes the collision probability to determine when to send an alert message to the pedestrian and/or the drivers of surrounding vehicles. More specifically, as soon as the position of the pedestrian is within an alert zone, the pedestrian sends a message via WiFi Direct to surrounding vehicles to obtain necessary information for calculating the collision probability. The message consists of $t_c$, $t_p$, $v_c$ (Figure 4.10), where $t_c$ is the time for the vehicle to reach the crossing; $t_p$ is the time for the pedestrian to reach the crossing, and $v_c$ is the speed of the vehicle. The message sent by the pedestrian contains only $t_p$; surrounding cars receiving this message then calculate their $t_c$ and $v_c$ values, add them to the message, and send it back to the pedestrian. Given the information contained in the returned message, the pedestrian calculates the collision probability to determine whether or not to send an alert message, and to whom to send it. These message exchanges occur continuously (with retransmission if necessary, *e.g.,* due to packet loss) while the pedestrian is inside an alert zone for computation of an up-to-date collision probability to ensure real-time safety.

In particular, $t_p$ is calculated differently depending on whether the pedestrian is walking or running. The Android context API is used to detect the user activity, *i.e.,* walking or running, and the pedestrian speed $v_p$ is determined accordingly. To ensure greatest safety, the maximum user walking speed as well as the maximum user running speed were used [1]. Now given the pedestrian speed $v_p$ and the distance between the pedestrian and the crossing within the alert zone denoted by $d_p$, $t_p$ can be calculated as $\frac{d_p}{v_p}$. Note that $t_c$ is calculated similarly by the driver based on the vehicle speed, and the distance to the crossing ahead obtained from the Location module.

Now if $t_p \gg \max(t_c(i))$ for all surrounding vehicles $i$, which means that all surrounding vehicles will pass before the pedestrian arrives at the crossing, no alert message is generated. Note, however, that any vehicle appearing in the range of WiFi Direct may invalidate this condition as the collision assessment is continuously performed. On the other hand, if the condition is not true, there is a chance of collision. We then define the 'user warning time' denoted by $t_{warning}$ as $\min(t_c(i)) - t_p$ such that $t_{warning} > 0$, *i.e.,*, if there is at least one car $i$ that has not passed the crossing when the pedestrian arrived at the crossing. Now given the user warning time, the following conditions must be satisfied to send an alert message to the pedestrian and/or driver:

- The pedestrian is walking/running while viewing the phone.
- The probability of collision is greater than a threshold.

The first condition ensures that if the pedestrian is stopped, or not viewing his or her phone, then there is no need to alert him. The second condition indicates that the probability of collision must be greater than a threshold. Now we calculate the probability of collision given $t_{warning}$. The probability of collision is defined as:

$$P(t_{delay} + t_{react} + t_{skid} > t_{warning}), \tag{4.7}$$

where $t_{delay}$ is the round-trip message delay for a single-hop 802.11 link; $t_{react}$ is the driver reaction delay; and $t_{skid}$ is the amount of time from the point when the driver applies brakes until the car completely stops. If the sum of these time components is greater than $t_{warning}$, the likelihood of collision is deemed high. In particular, we disregard the WiFi Direct connection establishment time since the connection has been already established before the first message is sent from the pedestrian.

The probability of collision is explained in more detail. $t_{delay}$ is empirically obtained as the pedestrian continuously exchanges messages with the vehicles. More specifically, we use the average of measured round-trip message delays as $t_{delay}$. In calculating $t_{react}$, it is known that the log-normal probability model fits the driver reaction time well [29]. Thus, $t_{react}$ is defined as follows:

$$f(x|\mu, \sigma) = \frac{1}{x\sigma\sqrt{2\pi}} e^{\frac{-(\ln x - \mu)^2}{2\sigma^2}}, \tag{4.8}$$

where we select $\mu = 1.14$ and $\sigma = 0.32$ [15].

To calculate $t_{skid}$, we first derive the distance $d_{skid}$ that a vehicle moves before complete stop when full brakes are applied as follows.

$$d_{skid} = \frac{mv^2}{2f}, \tag{4.9}$$

where $m$ is the vehicle mass, $v$ is the vehicle speed, and $f$ is the resistance force, which is given as follows [17].

$$f = \mu_k mg + \frac{\rho A C_d v_r^2}{2} + f_0, \tag{4.10}$$

where $\rho$ is the density of air, $A$ is the cross-sectional area of the vehicle, $C_d$ is the drag coefficient, $v_r$ is the speed of the vehicle relative to the air, and $f_0$ is the other resistance force, *e.g.*, due to the tire condition, and performance of the braking system. In our experiments, performed on a sunny day on a good conditioned road with Volkswagen Passat 2013, we used the parameter: $m = 1400kg$, $\mu_k = 0.8$, $A = 2.7m^2$, $C_d = 0.25$, $\rho = 1.23kg/m^3$ according to [9][24][18]. $v_r$ was approximated as the current vehicle speed $v$ due to the slow wind speed. Thus,

$$t_{skid} = \frac{d_{skid}}{v}. \tag{4.11}$$

Now since $t_{delay}$, $t_{skid}$, and $t_{warning}$ are known, the collision probability can be written as:

$$P(t_{react} > t_{warning} - t_{delay} - t_{skid}). \tag{4.12}$$

which can be calculated as $t_{react}$ follows the log-normal distribution specified in Eq. 4.8.

If all the conditions are satisfied, the pedestrian receives a warning message and is asked to respond to the message. An alert message is sent to the driver only if the pedestrian ignores the warning message a number of times to minimize driver distraction. The intuition is that due to the alert message, the user is expected to stop using his or her phone, and look up for safety. However, if the user keeps using it by ignoring the alert message $N$ times, an alert message is sent to the driver. This parameter $N$ can be adjusted to determine the tradeoff between the user safety and driver distraction. For our experiments, $N = 3$ was used.

### 4.6 Communication Engine

WiFi Direct supports only 1-to-1 or 1-to-many communication patterns. For example, CarA and CarB are connected to the WiFi Direct group owner PedA (Figure 4.10). However, the problem occurs when there is another pedestrian, say PedB who wants to send messages to CarA and CarB. But these cars would not respond to this request since they have formed a group with PedA already. To address this challenge, the following protocol is proposed.

The key idea is simple. A pedestrian, say PedB, overhears the network for a very brief period of time before it forms a group with the vehicles. If there is any pedestrian, say PedA, who has already formed a group with the vehicles, PedB stops broadcasting messages since it cannot form a group with the vehicles; instead PedB joins the group formed by PedA as a client. After that, any message exchanges between PedB and cars are done via PedA, basically implementing the communication between PedB and the vehicles without establishing a new group.

# Chapter 5

## Experimental Results

In this chapter, we evaluate the performance of WiPedCross specifically focusing on the following questions.

- Does the Location module provide sufficiently high positioning accuracy in both rural and metropolitan environments?
- Does the Energy module activate the GPS module in a timely manner? What is the effect of human walking speed?
- Does the Context module accurately detect the user-phone-viewing activity? What is the optimal system parameter for this module?
- Consequently, putting all modules together, are alert messages sent to the pedestrian and drivers correctly, reliably, and in a timely manner?

The proposed system was implemented on Samsung Galaxy S6 with 1.5GHz octa-core processor, and 3GB RAM running on Android 5.0. Since it was challenging to build a fully controlled environment on real-world roads, we used a spacious department parking lot as a test site. The dimensions of this test site are shown in Figure 5.1. To ensure safety, all experiments were conducted when there was no car in the parking lot.

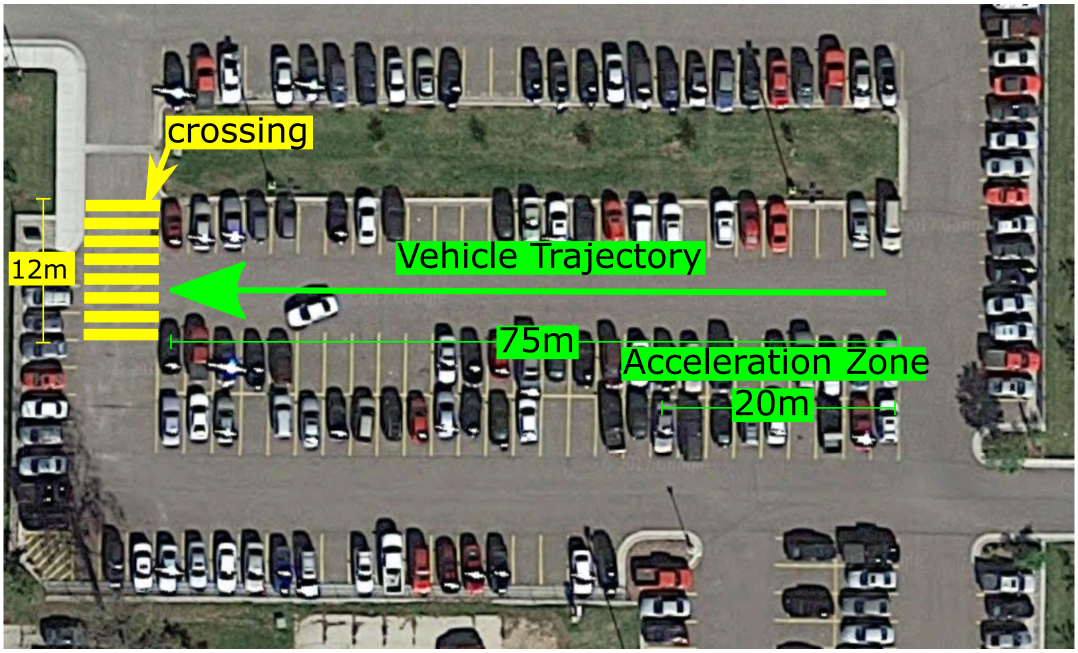


Figure 5.1: A test site for proof-of-concept experiments.

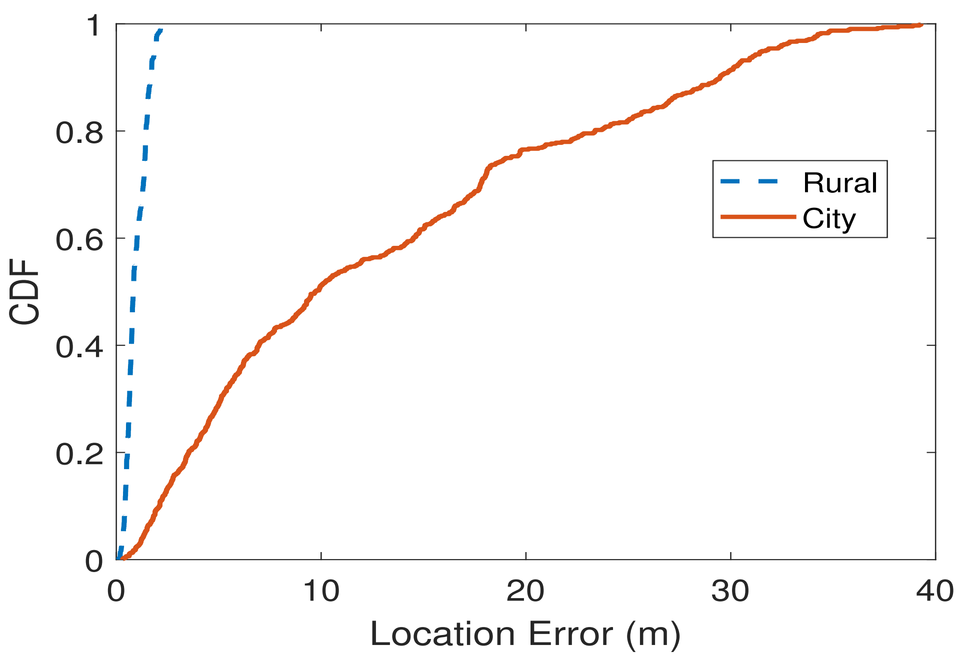


Figure 5.2: CDF of location error.

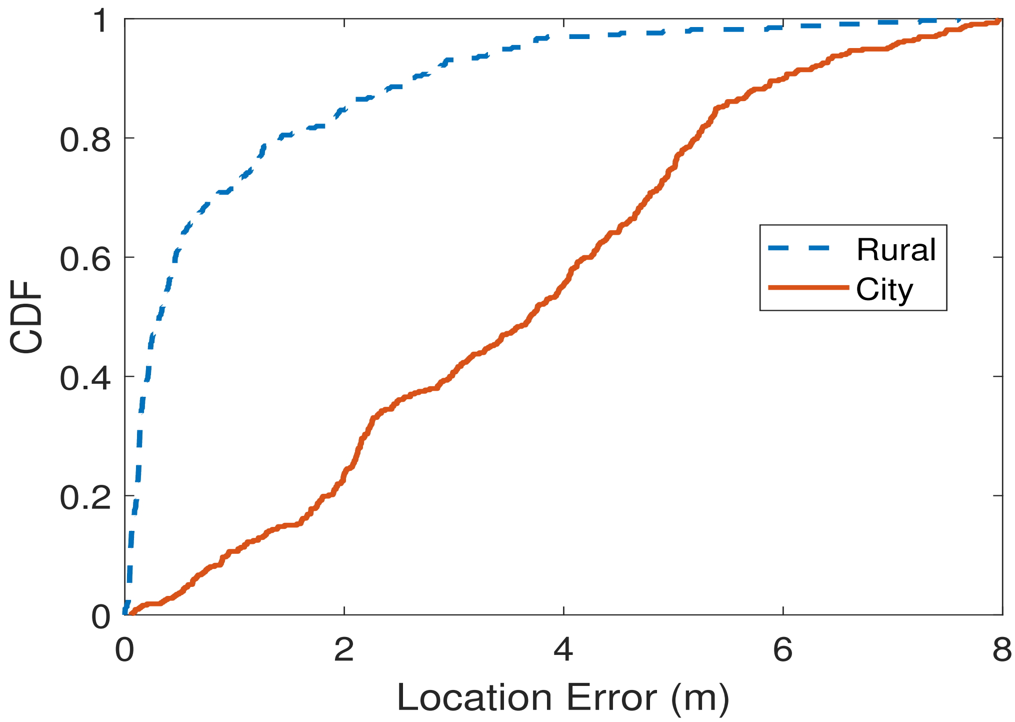


Figure 5.3: CDF of location error after applying Location module.

## 5.1 Positioning Accuracy

GPS locations were measured in both rural and metropolitan areas with concentrated skyscrapers. Five different walking trajectories were selected in each environment. Given the ground truth trajectory that is represented as a sequence of line segments on a sidewalk denoted by $\ell_i, 0 \leq i \leq n$, the location error for a GPS position $p$ was estimated as the shortest geodetic distance to the line segments as follows.

$$\min_{0 \leq i \leq n} \{\text{dist}(p, \ell_i)\}, \tag{5.1}$$

where $dist(a, b)$ is the geodetic distance between a GPS point $a$ and a line segment $b$.

Figure 5.2 depicts the location errors for the rural and city areas. The mean location error for the rural area was 0.97m while that for the city area was 12.99m. The error was significantly higher in the city area which made it challenging to use raw GPS data for WiPedCross.

We then applied the Location module to reduce the location error especially in the city area. Figure 5.3 displays that the average location error for the rural and the city areas was 0.88m, and 3.57m, respectively. A notable result is that the location error for the city area was significantly decreased by 72%. Although an average error of 3.57m is not completely negligible, this huge error reduction by the Location module allows us to easily compensate for this error, *e.g.,* by extending the range of an alert zone.

## 5.2 Energy Efficiency

The longer the GPS module is put into sleep mode, the more energy can be saved. On the other hand, it is important to reactivate the GPS module in a timely manner to ensure that an alert message is sent to the users at the right moment. This set of experiments thus are performed to confirm that the Energy module activates/deactivates the GPS module appropriately to achieve the balance between energy efficiency and timely generation of an alert message. As a performance metric, we used the geodetic distance between the alert zone and the pedestrian location measured when the GPS module was reactivated.

To understand the effect of real-world human walking speed on the performance

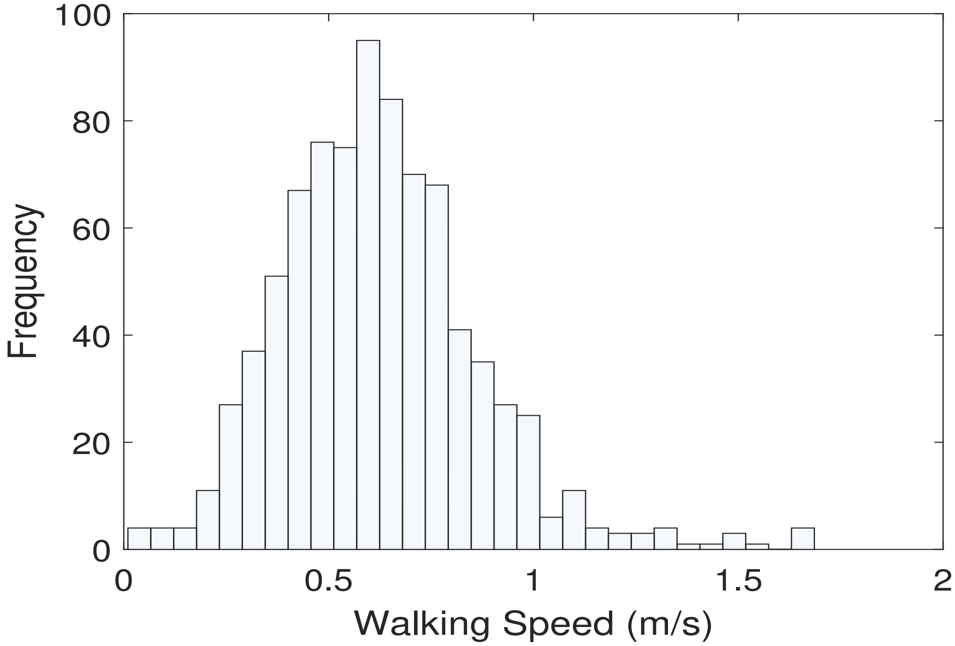


Figure 5.4: Histogram of measured walking speed.

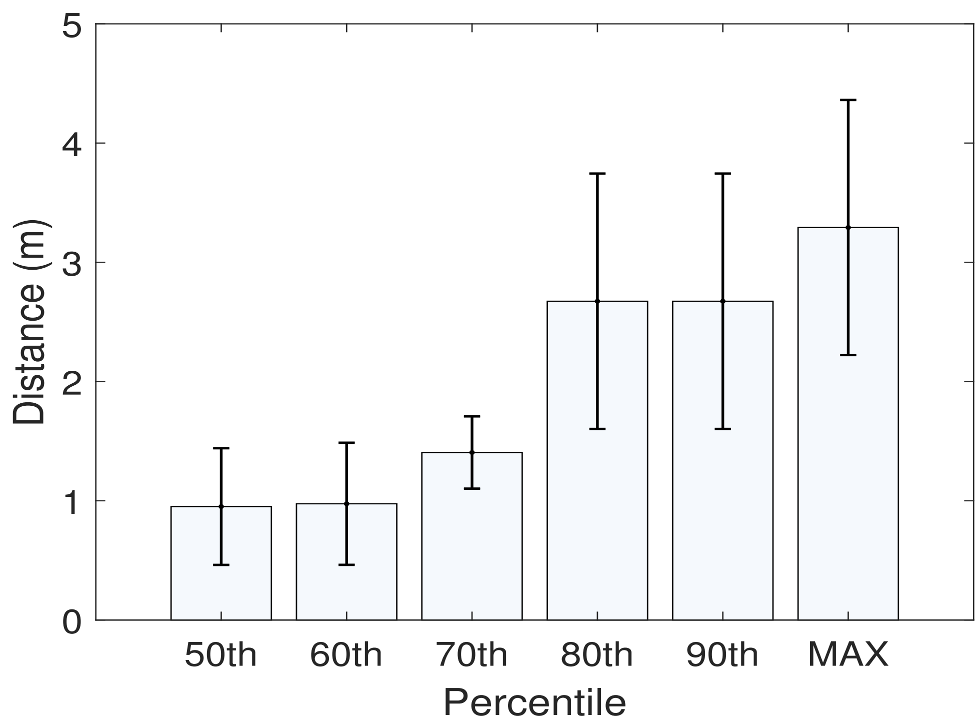


Figure 5.5: Performance of Energy module with varying walking speed.

of the Energy module, we collected walking speed for a duration of 5mins, from which we derived a histogram of walking speed (Figure 5.4). And then, based on the histogram, varying human walking speed was used in estimating the time that the user arrives at the nearest alert zone. The human subject started walking 30m away from an alert zone, and we measured the distance between the user location and the alert zone when the GPS module was reactivated by varying the human walking speed. Results are depicted in Figure 5.5. The results indicate that the GPS module was activated nearly on time, *i.e.,* about 1m away from the alert zone when the average user walking speed (at the 50th percentile) was used, and up to 5m when the user walking speed was maximum. Although the distance increased for faster walking speed, considering the location error and to provide utmost safety, we determined to use fast walking speed for our experiments (*i.e.,* the maximum human brisk walking speed [1]).

## 5.3 Context Detection

The Context module is used to detect the user-phone-viewing event. To evaluate the effectiveness of the module, we determined the appropriate size $\phi$ of a sliding window that consists of acceleration magnitude values. We then found the threshold MAD value denoted by $\Gamma$ such that the user-phone-viewing and non-viewing events are effectively differentiated. Consequently, based on $\phi$ and $\Gamma$, the performance of the Context module was evaluated.

As the first step, we performed experiments to determine $\phi$. It must be selected such that it clearly distinguishes between the user-phone-viewing and non-viewing events. Let $X = \{x_1, x_2, ..., x_\phi\}$ denote MAD values for the phone-viewing event, and $Y = \{y_1, y_2, ..., y_\phi\}$ be MAD values for the non-viewing event. A metric $\Delta$ is defined to quantify how well the two events are distinguished as follows.

$$\Delta = \frac{\sum_{i=1..\phi}(|y_i - x_i|)}{\phi}. \tag{5.2}$$

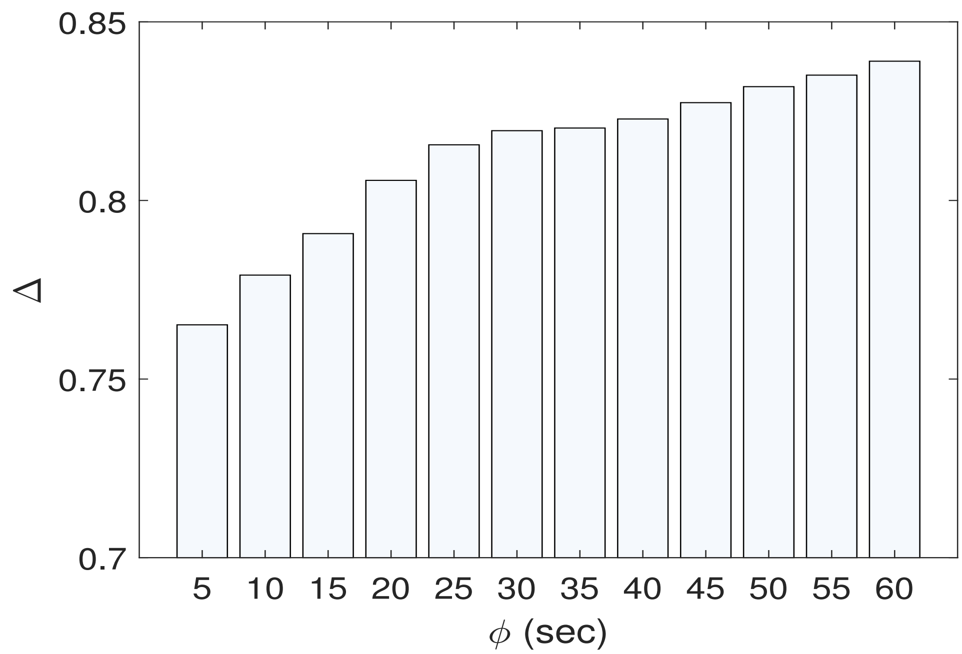


Figure 5.6: Effect of window size.

A greater $\Delta$ value indicates clearer distinction between the two events. We measured $\Delta$ by varying $\phi$. Seven volunteers participated in this set of experiments. Results are depicted in Figure 5.6. As shown, as $\phi$ increased, greater $\Delta$ values were obtained. These results demonstrate that more acceleration magnitude samples in a larger sliding window leads to better separation between the two events. We also note, however, that a larger $\phi$ causes delay in initializing a sliding window with acceleration


Figure 5.7: Detection accuracy of Context module.

magnitude samples. Considering the time to start the app and the time for a human subject to reach the crosswalk in our experimental setting, we decided that the window size $\phi$ of 25sec was appropriate.

Given the MAD values of the two events that are separated sufficiently by selecting an appropriate window size $\phi$, the threshold MAD value $\Gamma$ was determined. More specifically, we used the average of the mid points between a pair of MAD values (*i.e.,* one for the phone-viewing event, and the other one for the non-viewing event) as the threshold $\Gamma$:

$$\Gamma = \frac{\sum_{i=1..\phi}\left(\frac{y_i + x_i}{2}\right)}{\phi}. \tag{5.3}$$

Now based on the threshold $\Gamma$ and the window size $\phi$, we measured the accuracy of the Context module for each participant. Seven participants walked for a duration of 5 mins while performing phone-viewing and non-viewing actions. Figure 5.7 depicts the results, which demonstrate that varying detection accuracy was obtained for different participants potentially due to different phone-using styles. However, the overall average accuracy was sufficiently high as 92%.

## 5.4 Putting It together

We have verified the effectiveness of each module of WiPedCross, and determined appropriate system parameters. Now we are ready to incorporate all modules and conduct experiments and in-depth analysis on the overall reliability, and efficiency of WiPedCross specifically concentrating on whether alert messages are sent to pedestrians and drivers correctly, reliably, and in a timely manner.

We obtained time $t_p$ when the pedestrian sent an alert message to the driver, and recorded time $t_c$ when the vehicle received the message. The user warning time $|t_c - t_p|$ was then calculated that represents the amount of time allowed for the driver to avoid collision after discovering the pedestrian, *i.e.,* after receiving the alert message. Figure 5.8 depicts the user warning time with varying vehicle speed. As shown, the driver is given less than 2 seconds when he was driving at the speed of 20mph. On the other hand, he had a relatively sufficient amount of time of greater than 10 seconds when he was driving at 5mph.

Considering the measured $t_{delay}$ and $t_{skid}$, the collision probability was

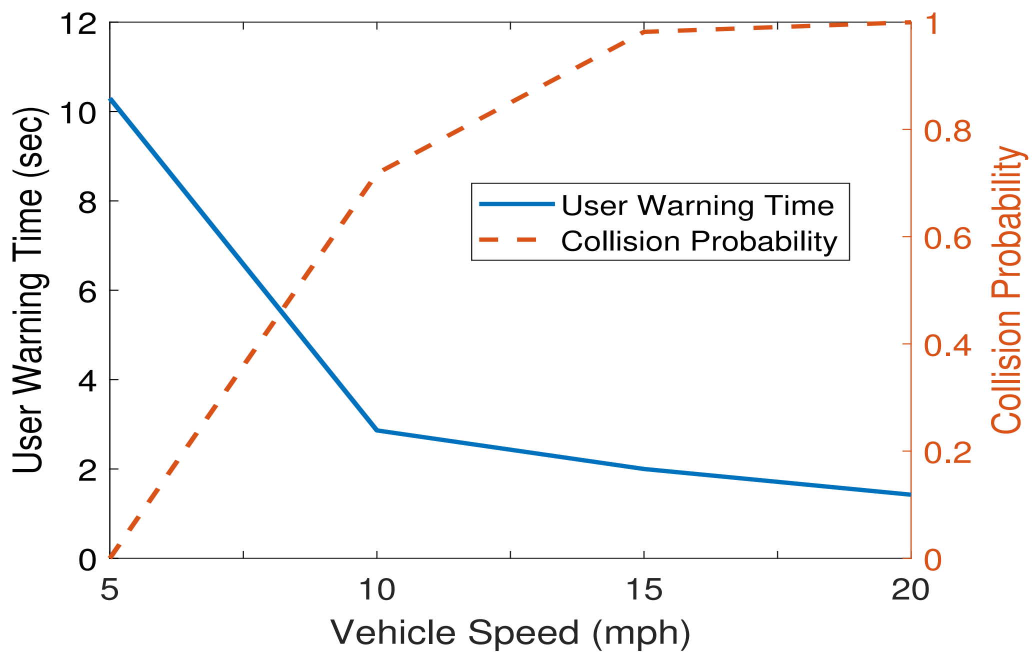


Figure 5.8: User warning time and collision probability.

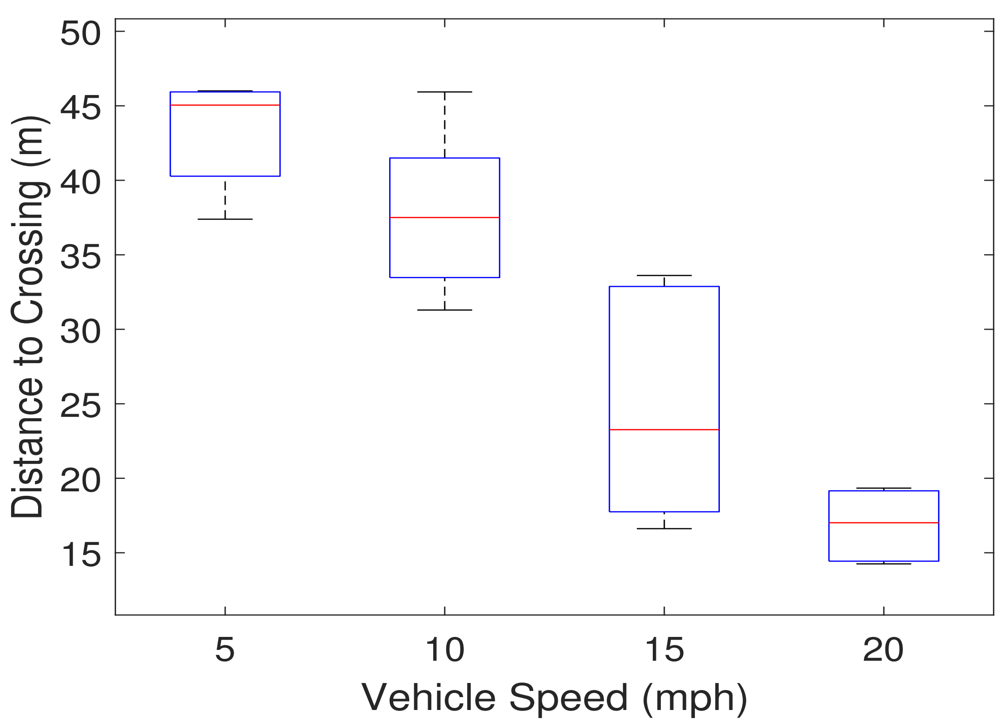


Figure 5.9: Driver warning per vehicle speed.

calculated when the pedestrian sent an alert message to the driver. The results are shown in Figure 5.8. The results indicate that after subtracting $t_{delay}$ and $t_{skid}$ from the user warning time, the driver is left with only less than 1 second to avoid the accident, resulting in a very high collision probability of nearly 100% when the vehicle speed was 20mph. On the other hand, WiPedCross determined that the collision probability for the vehicle speed of 5mph was almost 0% as long as the driver received the alert message and applied braking.

Figure 5.9 shows the distance from the driver to the crossing when the driver received the alert message. To calculate the collision probability and to ensure that the driver received the alert message, we set the threshold for the collision probability to 0, *i.e.,* the alert message was sent when the pedestrian entered into the alert zone, which was set to 7m away from the crossing. Note that this threshold was used for experimental purposes only, and it must be used to adjust the tradeoff between the pedestrian safety, and driver distraction.

# Chapter 6

# Conclusion

## 6.1 Summary

We have presented the design, implementation, and a pilot study of WiPedCross, a pedestrian safety framework based on the WiFi Direct technology that effectively senses, assesses a risk, and provides a warning to the user in a timely and energy efficient manner to prevent traffic accidents. All the challenges identified and addressed one by one. Careful considerations were made to ensure maximum pedestrian safety by selecting appropriate parameters. For example, the choice of brisk human walking speed while testing ensures maximum safety for pedestrians walking at an average speed. Each system components of WiPedCross are tested extensively. The system components of WiPedCross that address numerous practical challenges related to improving positioning accuracy, energy efficiency, and risk assessment will be useful resources for the development of other solutions not only for pedestrian safety but also for general transportation apps. The main focus of this pilot study was to test the feasibility, reliability, and the validity of each module of the framework and the proposed framework. All the tests were performed in the departmental parking lot by defining the crossing zone, the acceleration zone and the sidewalk. This pilot study was challenging to perform in the real road situations because of several moving vehicles. The full implementation of the system was beyond the scope of our thesis. The prototype Android applications were build to test each module. Our pilot study suggests that the system can be fully deployed in the rural areas in the Line-of-Sight (LOS) conditions. Additionally, our system in this phase does not guarantee ultimate pedestrian safety because it is very challenging to model the user behavior who is either driving or walking. Different users have different behavior while walking or driving. Some might slow down their speed at the crossings while other might just move on with the same speed. Some pedestrians walk straight while other might walk in a zigzag way. However, it helps the pedestrian safety app developers to include all the necessary modules and choose the value of system parameters as per their need. All the transportation-related app makes use of GPS to locate the user. The Energy module suggests the developers to put the GPS in sleep mode when it is not in use. Any general purpose app developers can make use of any of the modules and integrate into their app. For example, one may build an app that utilizes context detection for apps which needs to be turned off when a user is not viewing. The benefit of this pilot study is that the developers do not need to perform the proof of concept experiments again when building their apps. Our study provides all the necessary evaluation of each module and the overall system.

## 6.2 Future works

As a future work, the effect of driver compliance rates will be analyzed, and experiments involving a large number of vehicles and pedestrians will be performed to evaluate the potential impact of network congestion. The experiments were performed in the Line-of-sight(LOS) condition where there are no obstructions between the drivers and the pedestrians. In the future, experiments will be performed in the Non-line-of-sight(NLOS) condition where large buildings near the crossing zone are present, obstructing the signals between drivers and the pedestrians. The accuracy of GPS decreases as far as the urban areas are concerned. Further research will be done to improve the GPS accuracy. Extensive experiments will be performed in urban areas and the behavior of both the pedestrians and the drivers at the crossing zone will be modeled and integrated within our framework to increase the accuracy of the collision probability. Experiments were performed module by module assuming other helper modules were present while testing a module. Experiments will be performed by integrating all the modules together, building a complete app.

## References

[1] Brisk Walking Speed.
https://www.verywell.com/how-fast-is-brisk-walking-3436887. Accessed: 2017-5-1.

[2] CBS News. http://www.cbsnews.com/news/1-in-3-use-phones-text-while-crossing-the-road/. Accessed: 2017-5-1.

[3] Health Line. http://www.healthline.com/health-news/tech-texting-while-walking-causes-accidents-031014#1. Accessed: 2017-5-1.

[4] Look Up. https://intersections.devpost.com/submissions/26518-look-up. Accessed: 2017-5-1.

[5] Pedestrians.
https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812375. Accessed: 2017-5-1.

[6] Wi-Fi Direct.
https://www.slideshare.net/shivam_kedia/wifi-direct-26864237. Accessed: 2018-6-19.

[7] Heba Aly and Moustafa Youssef. semmatch: Road semantics-based accurate map matching for challenging positioning data. In *Proc. of SIGSPATIAL*, 2015.

[8] José Javier Anaya, Pierre Merdrignac, Oyunchimeg Shagdar, Fawzi Nashashibi, and José E Naranjo. Vehicle to pedestrian communications for protection of vulnerable road users. In *Proc. of Intelligent Vehicles Symposium*, 2014.

[9] Dennis R Ankrum. Ivhs-smart vehicles, smart roads. *Traffic Safety (Chicago)*, 92(3), 1992.

[10] Daniel Camps-Mur, Andres Garcia-Saavedra, and Pablo Serrano. Device-to-device communications with wi-fi direct: overview and experimentation. *IEEE wireless communications*, 20(3):96–104, 2013.

[11] Appala Chekuri and Myounggyu Won. Automating wifi fingerprinting based on nano-scale unmanned aerial vehicles. In *Vehicular Technology Conference (VTC Spring), 2017 IEEE 85th*, pages 1–5. IEEE, 2017.

[12] Kaustubh Dhondge, Sejun Song, Baek-Young Choi, and Hyungbae Park. Wifihonk: smartphone-based beacon stuffed wifi car2x-communication system for vulnerable road user safety. In *Proc. of VTC*, 2014.

[13] Andreas Fackelmeier, Christian Morhart, and Erwin Biebl. Dual frequency methods for identifying hidden targets in road traffic. In *Advanced Microsystems for Automotive Applications 2008*, pages 11–20. Springer, 2008.

[14] Alexander Flach and Klaus David. Combining radio transmission with filters for pedestrian safety: Experiments and simulations. In *Proc. of VTC*, 2010.

[15] Denos Gazis, Robert Herman, and Alexei Maradudin. The problem of the amber signal light in traffic flow. *Operations Research*, 8(1):112–132, 1960.

[16] Ping-Fan Ho and Jyh-Cheng Chen. Wisafe: Wi-fi pedestrian collision avoidance system. *IEEE Transactions on Vehicular Technology*, 2016.

[17] Ping-Fan Ho and Jyh-Cheng Chen. Wisafe: Wi-fi pedestrian collision avoidance system. *IEEE Transactions on Vehicular Technology*, 2016.

[18] Wolfgang Hugemann. Driver reaction times in road traffic. In *Proc. of XI EVU*, 2002.

[19] Shubham Jain, Carlo Borgiattino, Yanzhi Ren, Marco Gruteser, Yingying Chen, and Carla Fabiana Chiasserini. Lookup: Enabling pedestrian safety services via shoe sensing. In *Proc. of MobiSys*, 2015.

[20] Yuan-Heng Sun Jin-Shyan Lee, Ming-Feng Dong. A comparative study of wireless protocols: Bluetooth, uwb, zigbee, and wi-fi. *IECON 2007 - 33rd Annual Conference of the IEEE Industrial Electronics Society*, 2007.

[21] Chi-Han Lin, Yi-Ting Chen, Jyun-Jie Chen, Wen-Chan Shih, and Wen-Tsuen Chen. psafety: A collision prevention system for pedestrians using smartphone. In *Proc. of VTC*, 2016.

[22] Zhenyu Liu, Lin Pu, Zhen Meng, Xinyang Yang, Konglin Zhu, and Lin Zhang. Pofs: A novel pedestrian-oriented forewarning system for vulnerable pedestrian safety. In *Proc. of ICCVE*, 2015.

[23] Paul Newson and John Krumm. Hidden markov map matching through noise and sparseness. In *Proc. of SIGSPATIAL*, 2009.

[24] Robert W Rivers. *Evidence in traffic crash investigation and reconstruction: identification, interpretation and analysis of evidence, and the traffic crash investigation and reconstruction process.* Charles C Thomas Publisher, 2006.

[25] Ali Rostami, Bin Cheng, Hongsheng Lu, John B Kenney, and Marco Gruteser. Performance and channel load evaluation for contextual pedestrian-to-vehicle transmissions. In *Proceedings of the First ACM International Workshop on Smart, Autonomous, and Connected Vehicular Systems and Services*, 2016.

[26] Florian Seeliger, Galia Weidl, D Petrich, Frederik Naujoks, Gabi Breuel, Alexandra Neukum, and Klaus Dietmayer. Advisory warnings based on cooperative perception. In *Proc. of Intelligent Vehicles Symposium*, 2014.

[27] Aawesh Shrestha and Myounggyu Won. Deepwalking: Enabling smartphone-based walking speed estimation using deep learning. *arXiv preprint arXiv:1805.03368*, 2018.

[28] Chika Sugimoto, Yasuhisa Nakamura, and Takuya Hashimoto. Prototype of pedestrian-to-vehicle communication system for the prevention of pedestrian accidents using both 3g wireless and wlan communication. In *Wireless Pervasive Computing, 2008. ISWPC 2008. 3rd International Symposium on*, pages 764–767. IEEE, 2008.

[29] George T Taoka. Brake reaction times of unalerted drivers. *ITE journal*, 59(3):19–21, 1989.

[30] Tianyu Wang, Giuseppe Cardone, Antonio Corradi, Lorenzo Torresani, and Andrew T Campbell. Walksafe: a pedestrian safety app for mobile phone users who walk and talk while crossing roads. In *Proc. of HotMobile*, 2012.

[31] Yan Wang, Jie Yang, Hongbo Liu, Yingying Chen, Marco Gruteser, and Richard P Martin. Sensing vehicle dynamics for determining driver phone use. In *Proc. of MASS*, 2013.

[32] Myounggyu Won. A review on v2v communication for traffic jam management. *Vehicle-to-Vehicle and Vehicle-to-Infrastructure Communications: A Technical Approach*, page 1, 2018.

[33] Myounggyu Won, Haitham Alsaadan, and Yongsoon Eun. Adaptive audio classification for smartphone in noisy car environment. In *Proceedings of the 2017 ACM on Multimedia Conference*, pages 1672–1679. ACM, 2017.

[34] Myounggyu Won, Ashutosh Mishra, and Sang H Son. Hybridbaro: Mining driving routes using barometer sensor of smartphone. *IEEE Sensors Journal*, 17(19):6397–6408, 2017.

[35] Myounggyu Won, Taejoon Park, and Sang H Son. Toward mitigating phantom jam using vehicle-to-vehicle communication. *IEEE transactions on intelligent transportation systems*, 18(5):1313–1324, 2017.

[36] Myounggyu Won, Shaohu Zhang, and Sang H Son. Witraffic: low-cost and non-intrusive traffic monitoring system using wifi. In *Computer Communication and Networks (ICCCN), 2017 26th International Conference on*, pages 1–9. IEEE, 2017.

[37] Myounggyu Won, Yunfan Zhang, XiaoZhu Jin, and Yongsoon Eun. Wiparkfind: Finding empty parking slots using wifi. In *2018 IEEE International Conference on Communications (ICC)*, pages 1–6. IEEE, 2018.

[38] Xinzhou Wu, Radovan Miucic, Sichao Yang, Samir Al-Stouhi, James Misener, Sue Bai, and Wai-hoi Chan. Cars talk to phones: A dsrc based vehicle-pedestrian safety system. In *Proc. of VTC*, 2014.

[39] Shaohu Zhang, Myounggyu Won, and Sang H Son. Witraffic: Non-intrusive vehicle classification using wifi. In *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems CD-ROM*, pages 358–359. ACM, 2016.