NAIST-IS-DD1261029

Doctoral Dissertation

Mobile Communication and Sensing for Emergency Management Before and After Disaster Events

Jane Louie Fresco Zamora

September 16, 2015

Department of Information Science
Graduate School of Information Science
Nara Institute of Science and Technology

A Doctoral Dissertation
submitted to Graduate School of Information Science,
Nara Institute of Science and Technology
in partial fulfillment of the requirements for the degree of
Doctor of ENGINEERING

Jane Louie Fresco Zamora

Thesis Committee:

Professor Suguru Yamaguchi	(Supervisor)
Professor Minoru Okada	(Co-supervisor)
Professor Keiichi Yasumoto	(Co-supervisor)
Professor Kazutoshi Fujikawa	(Co-supervisor)
Professor Youki Kadobayashi	(Co-supervisor)

Mobile Communication and Sensing for Emergency Management Before and After Disaster Events*

Jane Louie Fresco Zamora

Abstract

This dissertation presents the use of mobile devices as specific ICT solutions for some natural disasters in order to tackle current issues with existing technologies for mitigating risks before and after disaster events. Several modern ICT applications nowadays adapt to disaster-related issues that enable the society to mitigate the risks brought about by these events. However, with the onset of climate change, more individuals especially those who are considered to be at a disadvantage are increasingly vulnerable to disaster risks. Dynamic improvements in ICT applications, thus, are yet to be continually implemented.

In this research, the motivation is to be able to save lives from the impending risks of natural hazards by providing the opportunity to be informed and communicate despite the limited resource. Technology is speedily growing but not all individuals have similar access rate and adaptation to the growth. Moreover, in the event of a large-scale disaster, all things essential to the function of one system, i.e., communication towers, may be lost. Therefore, in this regard, there is a need for a low-cost and easily implementable technological countermeasures.

In this dissertation, specific implementations for smartphone-based information sharing by communication and environmental observation for rescue and disaster preparation, respectively, are proposed. Particularly, because communication channels can be lost during a large-scale disaster such as in a devastating earthquake, a quick countermeasure for communication to rescue victims

*Doctoral Dissertation, Department of Information Processing, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-DD1261029, September 16, 2015.

for within the critical 72 hours was investigated in this research. In this regard, SOSCast, which is an Android OS-based application is developed to enable victims to exchange emergency messages even when trapped under debris. However, SOSCast is yet to be improved in terms of efficiency as it currently functions limited to battery life. In addition, because of the limited resources in some parts of the world, not all individuals have easy access to weather information which makes them vulnerable to flooding and landslides, for instance, during heavy rains. In this regard, the research focus was to examine easily implementable and low-cost alternatives to weather observation. That is, in addition to the information that can be measured by the built-in sensors in smartphones such as humidity, pressure, temperature, acceleration, etc., we also investigated on received signal level (RSL) on the device. This information can be a supporting data to pinpoint heavy rain events, as such, we conducted several measurement experiments using smartphones to support the idea that pervasive smartphone-based data is helpful in describing heavy rain events.

Keywords:

smartphone, sensing, communication, information sharing, mobility, disaster events, risk mitigation, emergency management, search and rescue

Acknowledgements

To my parents, for raising me a curious child.
To my teachers, who opened doors and led the ways.
To my friends, for the insanity.
To my colleagues, near and far, for the shared experience and knowledge.
To random people I met along the way, who widened my perspectives.
To MEXT and NAIST Foundation, for financially supporting my curiosity.
Thank you.

Contents

List of Figures

List of Tables

1. Introduction

1.1 Motivation

Information and communications technology (ICT) in this century has become more advanced parallel to the rapid growth of new technological developments. Cellular phones beginning the 1970s have particularly evolved from large blocks of plastic-cased basic wireless communication devices to sleek aluminum-cased devices, which recently include sophisticated minuscule sensors [15]. According to ITU's (International Telecommunication Union) latest ICT statistics, the mobile-cellular subscriptions worldwide has reached approximately 7 billion in the year 2015 from 738 million in 2000 [17]. In this regard, the use of mobile devices for ICT applications in education, health, transportation, weather, disaster and risk mitigation etc., has become increasingly endless.

ICT generally enables the user to access and manipulate data as well as transmit and receive it. With the smartphone as a tool for ICT, it can enable the user to manage data anywhere desired. Thus, it is beneficial to take advantage of the smartphone in sensing actual information specifically where pinpoint data is needed. For instance, in environmental monitoring, there are particular remote areas where observational data is required but where conventional sensors are difficult to install. It is in such cases when mobile devices can be very helpful in delivering environmental data.

Some of the modern ICT sensing applications nowadays also adapt to disaster-related issues that enable the society to mitigate the risks brought about by these events. With the continued increase of mobile device use especially in least developed countries, it may be one of the low-cost solutions that could save their lives as they are statistically the most vulnerable to disasters.

1.2 Problem Statement

Recently, disasters caused by natural hazards are becoming frequent globally and is commonly attributed to the impact of climate change by scientists. In the latest IPCC (Intergovernmental Panel on Climate Change) report of 2014, 90% of the population in Asia are projected to be vulnerable to flooding and rainfall-induced landslides due to increase in sea level rise, heavy rain, and tropical cyclones [11]. In this regard, researchers are also developing ICT applications as solutions to mitigate risks that follow these disasters. There are currently several ways of providing ICT solutions for disaster mitigation. In our study, we focus in the application of the smartphone as our ICT tool to communicate, exchange data, and receive timely and actual information. More specifically, we require these solutions to be in the form of low-cost and low-maintenance tools that are also easily implementable as alternatives for search and rescue and risk mitigation before or after a devastating disaster occurs.

This study attempts to answer three underlying issues in the following discussions, for which specific proposed solutions are described for each issue:

1. **When communication channels are limited or inexistent in the event of a devastating earthquake, the probability that victims can be rescued within 72 hours of expected survival duration is low.**

   Normal use of the mobile devices typically requires data to be sent over wireless channels serviced by communication towers such as WiFi or 3G. Phone calls, text messages, or media to be sent can be quickly transmitted across a wide interconnection of these channels and in long distances. In the event that the network infrastructure is damaged, there can only be a few of these towers to handle the transmission traffic. In such cases where victims would naturally ask for help by making phone calls or send messages, the limited channels will not be capable of processing such activities promptly. Thus, this would result to either the rejection of the victims' call or unsent messages.

   Research Goal 1: *Develop a communication scheme using the Bluetooth (BT) function that would enable victims to send emergency messages directly from their mobile device to another*

2. **Scarcity of in-situ weather sensors can result to a lack of ground-based observation data with the consequence of inaccurate forecasts on localized heavy rains.**

Heavy rains are usually detected by a combination of data from satellite, radar, and weather stations. However, in the case satellite and radar for instance, the observation coverage is typically from 100 to 100,000 km which is sometimes incapable of detecting localized heavy rains occurring at less than 10 km. Adding more weather stations on the ground, for example, may not be economical in terms of cost and deployment. In that regard, wireless communication towers have long been existent since the beginning of WiFi and 3G technologies and its transmission information can be helpful in detecting the onset or occurrence of localized heavy rains. Wireless networks are typically affected by water vapor and this is evident in the decrease of transmission power level. If this information can be harnessed to describe rain activities, then it can be a cost-efficient supporting information for describing rain events.

Research Goal 2: *Find an alternative and supporting information for existing systems, that is, in the form of Received Signal Level (RSL) in wireless networks that could describe heavy rain activities.*

3. **Several available sensed weather data via participatory sensing using mobile devices can be unreliable especially on the process of obtaining these information.**

   Considering that it is the aim in this study to mobile devices as a tool for obtaining weather information, there are potentially abundant data that can be used to describe in-situ weather conditions. Nowadays, there is a gaining popularity of mobile device use and harvesting data from these can suffice for the lack of ground weather information. However, in the process of obtaining these information, user activity can greatly affect the quality of the data. As mobile devices, such as in the form of smartphones, are typically used as devices for communicating, device movement and handling can distort the information taken in the background. This in turn can greatly affect the quality of the forecasts as well in large scale.

   Research Goal 3: *Perform a series of calibration processes on several obtained data as affected by user activity and evaluate the reliability of the results.*

1.3 Contributions

It is the main objective of this research to be able to use *smartphones as pervasive sensors for emergency management*, that is, for search and rescue and risk mitigation. To support search and rescue of victims trapped under debris, the primary goal is to develop a tool for victims to easily send emergency messages by using their mobile device even without Internet. And so far, we have developed an approach of a device-to-device communication via BT in the form of a smartphone application. Currently, this tool is in the process of improvement for a more efficient exchange of emergency messages. On the other hand, in the search for alternative ways of obtaining in-situ weather observation data, we have conducted several measurement experiments of using the smartphone as a sensor and information sharing device.

Thus far, we contribute to providing low-cost, low-maintenance, and easily implementable solutions for emergency management. Particularly, we develop Android OS-based smartphone applications to communicate, exchange data, and receive information. In this way, we empower individuals to be part of the disaster mitigation by participating in the process. Moreover, we contribute by supporting individuals, especially victims of disasters, to help themselves to hasten the rescue process. Also, we encourage individuals to participate by sensing using their devices to contribute data that would be very helpful in forecasting natural disasters.

The specific contributions are as follows:

1. Introduce a way to communicate between smartphones even without the conventional Internet-based communication

2. Introduce a novel way of using smartphones as a weather sensor in addition to its communication capabilities

1.4 Organization

The remainder of this dissertation presents the details of each independent experiments that focus on the use of smartphones in mitigating risks before or after a particular disaster has occurred.

Chapter 2 discusses some cases of ICT applications of smartphones to examples of disaster-related situations. Chapter 3 presents a proposed work on using smartphones to communicate during post-disaster situations such as during a devastating earthquake. Chapter 4 presents an improve smartphone-based direct communication approach related to the prior chapter. Chapter 5 presents a proposed work on using smartphones to gather weather information by way of RSL, using the device to observe and measure current conditions. Chapter 6 presents experimentation of measuring in-situ weather conditions using the smartphone in collaboration with other existing commercial sensors. Chapter 7 discusses a calibration method considering the effects of user activity to the observed data. Finally, Chapter 8 summarizes the proposed works and introduces future works for the continuation and improvement of this study.

2. Smartphones for ICT in disaster research

In this study, we present ICT-based solutions with focus on two types of natural hazards, earthquake and heavy rain. This chapter discusses more background on the risks by the aforementioned phenomena as in Sec. 2.1. It is followed by Sec. 2.2 which highlights related works on device-to-device communications as a solution for emergency situations such as during earthquakes. Lastly, Sec. 2.3 discusses similar studies on conventional monitoring and observation of environmental conditions using ICT.

2.1 Natural disasters and risks

In the past decades, industrialization has caused the increased production of carbon dioxide causing more heat to be trapped on earth. In return, this has caused increased warming and abnormal greenhouse effect which is one of the reasons for the on-going debate on climate change. Specifically, the Fifth Assessment Report of IPCC [11] projected that increasing emissions of carbon dioxide into the atmosphere increases surface temperature. This, in effect, will "very likely" cause frequent heat waves and extreme precipitation. Flood events continue to increase in the past three decades according to World Bank [42]. In fact, it was reported in the World Disasters Report of 2013 [15] that Asia has had 38,986 deaths due to flooding based on statistics from 2003 to 2012. This number is the highest in among other continents, which accounts for 68.5% of the total.

However, climate change is believed not only to affect the frequency and intensity of precipitation but can also trigger earthquakes. According to Bill McGuire [28], the melting of the ice and the rise of sea levels destabilizes the plates causing the occurrence of earthquakes. Moreover, the excess water also drives the weathering of the underground volcanic activity which makes volcanoes prone to eruptions. Within a span of ten years, earthquakes and tsunamis has caused much damage in Asia with a death toll of 449,941 (66% of total worldwide). These numbers implies that there is much ICT innovation that needs to be done to save more lives.

IPCC has summarized in their recent report that "disadvantaged" individuals and groups are more likely vulnerable to the risks considering the effects of rising global mean temperatures. In that regard, many ongoing efforts are in place, developing several and sometimes conventional ways of mitigating deaths due to risks from effects of natural phenomena, particularly of earthquakes and heavy rainfall.

2.2 Emergency communications

In this section, we discuss related works on emergency communications during devastating disasters. Specifically, we focus on ICT innovations on communications on events where transmission channels are unavailable. Especially during large-scale disasters, victims of the disaster must be rescued during the 72 critical hours where communication plays a crucial role. It is important that emergency messages must reach designated rescuers in this time duration but because of destroyed infrastructures, the Internet or phone lines are unavailable. It is with this motivation that researchers thought of implementing peer-to-peer communications during emergency to bypass the unavailability of the communication channels. For instance, one study focused on collaborating WiFi, WiMax, and GEO (geostationary orbit) satellite as a hybrid mobile ad-hoc network for enabling VoIP calls and exchange of multimedia data between rescuers in an emergency situation [26]. Their approach for a speedy rescue via efficient communication and information sharing was to develop a hybrid network that replaces the conventional Push-to-Talk (PTT) communications. Their networks extends the communication range between rescue teams and accommodate multimedia data for a richer information content, which can not be done when using PTT. However, this type of system does not address the absence of communication infrastructures in the disaster area. On the other hand, [30] developed an overlay network called the Human-centric Wireless Sensor Network (HWSN) to support the use of conventional VHF/UHF radio communication systems. To make communication more efficient even without the usual infrastructure, firetrucks were utilized to route disconnected networks and stationary communication units were deployed to store information. However, this system may not apply to instances when the disaster area is only accessible by foot and thus, limiting search coverage. Moreover, for both of these studies, the participation of the victims in the rescue operations was not considered although the communication schemes were proven effective.

In both studies aforementioned, the focus was on the rescuers using the system to share disaster information for rescue operations. Note that, according to the World Disaster Reports 2013 [15], people affected in a disaster are to be considered "first responders" more than referring to them as victims. In addition, these victims needs to be involved hands-on with emergency response. By being first responders, disaster-affected people can aid in rescue operations by providing information by their selves. In that regard, *Twimight* [13] was developed to support disaster information acquisition from first responders or the victims of the disasters themselves. It is generally an online social networking service in the form of a smartphone application designed after the Twitter platform having similar basic functions, such as posting short text messages. *Twimight*'s approach to sharing information in a disaster-affected area that is independent of the communication infrastructure was basically developing a "disaster mode", which differentiates it from Twitter. That is, in an event of a disaster, the smartphone application enables the exchange, aggregation, and transmission of short messages directly with other smartphones via BT. However, some people may not be accustomed to Twitter and this manner of asking for help may not be direct. Also, the use of *Twimight* eventually requires Internet connection and thus, not ideal for use in a catastrophic disaster with the communication infrastructures destroyed.

The disaster mode in *Twimight* actually utilizes opportunistic communications via BT, which most smartphones nowadays are capable of. In that regard, references [43] and [23] explored and supported the idea of using smartphones with BT devices to disseminate or aggregate information. Opportunistic communications via BT is especially encouraged for use in situations where communication infrastructures are unavailable due to natural hazards. Even without existing channels, messages can be widely spread using smartphones via epidemic routing, which also ensures that these messages are received by nearby smartphones. Simultaneously, messages can be aggregated and ferried among smartphone devices until it reaches the recipients, such as the rescuers or family members for instance.

However, epidemic routing itself consumes high battery power when not managed efficiently. In the application of such opportunistic communication technique, if the smartphones ran out of power, then the probability of being rescued decreases. Several studies [10, 44], therefore, proposed clustering techniques among smartphones to optimize disaster information gathering and reduce battery consumption at the same time. For example, in [10], a wireless ad hoc network architecture named DistressNet was developed to organize multiple sensors to sense, localize, and communicate in a disaster situation where communication is congested, limited, or non-existent. To efficiently manage energy consumption in devices during disasters for a successful rescue, their approach was to optimize information delivery and situational awareness. The focus is mainly on the system architecture, protocol design, and application development that address the goals of DistressNet. On the other hand, in [44], algorithms for creating and managing clusters to conserve energy consumption among nodes in a mobile ad hoc network is proposed for use in a disaster scenario. Though having similar aims with [10], the difference lies in the solution method for clustering and battery conservation. To investigate if the techniques satisfy the aims, both studies have evaluated their respective systems in a simulation, but with [10] comparing different algorithms and [44] inspecting the performance of the proposed network mechanisms. However, [10] focuses more on the system and requires a hardware they developed to be deployed. In [44], no actual application were done to show the effectiveness of their system and there was no mention of securing the records of the transactions.

2.3 Environmental Sensing and Weather Information Services

In this section we describe research on sensors and sensor networks that focuses on environmental observation. Before digital instruments took over measurements of the real world, there used to be analog sensors where data logging was done by hand and may be at times subjective based on the observer. In the recent years, measurements are automated and better techniques are being proposed nowadays. In that regard, scientists have thought of other ways where they can maximize the use of digital sensors and sometimes develop new ones. For instance, remote areas that are usually difficult to access have very few observation data or sometimes not existent. Researchers working on SensorScope [16] for example, provides solutions for sensor network deployment customized according to different sensing requirements and geographical locations even in remotely difficult areas such as the Swiss alps. In this study, the researchers designed a system of low-cost sensors as alternative to the deployment of conventional sensing stations that are commonly priced at 80,000 USD each. As these may be costly, having only a few of these sensors would limit the spatial coverage of near-surface observation and the availability of weather information. In this study, we aim to use already available devices that can obtain environmental data without the high price. As an alternate solution to this sensor cost issue, researchers explore the potential of commercial wireless communication networks or CWCNs [36] as a way to lower the cost specifically of precipitation measurement and observation. In their study of reconstructing rain maps from RSL measurements of CWCNs, they have found that there is more work to be done in terms of accurate reconstruction and data sufficiency. The concept is favorable with the exception of having to consider more about the acquisition of RSL data especially with private communication companies. Some companies will not easily provide the RSL data as it may jeopardize their business. Also, for the purpose of increasing the accuracy of the rainfall map from RSL, it was mentioned as that their is a need for assimilating RSL data with in-situ rain gauges in sparse areas for instance. Which is why in this study, we take advantage of the growing popularity of mobile phones. As these devices are capable of sensing and data transmission, nowadays, it is a most likely cost-efficient way of obtaining environmental data especially for pinpointing extreme weather such as heavy rains. In a survey on the emerging field of mobile phone sensing or MPS [22], the applications for mobile device based information harvesting range from social network services, environmental monitoring, and personal health improvement. A typical example of an MPS application would be user context recognition systems [12] that aim to deliver better services based on user behaviour, for instance. Context includes user activities and interactions with other users or with the environment that is based on the MPS data. More often, to get a general view of a group of users, the data has be to be sourced from a multiple group of users having similar qualifying criteria. These criteria are based on categorization of MPS data by statistical methods

Table 1: List of smartphone models used and their available sensors

Common Name	Samsung Galaxy Nexus[33]	Samsung S3[35]	Samsung S4 [34]
Model	I9250	GT-I9300	GT-I9500
Android OS version	4.3	4.2.2	4.3
Light (lux)	✓	✓	✓
Proximity (cm)	✓	✓	✓
Gyroscope (rad/s)	✓	✓	✓
Accelerometer (m/s^2)	✓	✓	✓
Magnetometer (μT)	✓	✓	✓
Pressure (hPa)	✓	✓	✓
Temperature (°C)	N/A	N/A	✓
Humidity (%)	N/A	N/A	✓

like Mean, Median, Variance, etc., to analyze patterns in the data and group like patterns accordingly. In addition to context recognition, which frequently makes use of position or motion sensor data, environmental application of MPS as listed in [8] also includes utilizing microphones and cameras to provide audio and image samples of the environment, respectively. Another example of environmental MPS is discussed in [31] where it makes use of the temperature sensor to estimate urban air temperatures. With recent mobile phone models, like in Table 1, it is now possible to observe pressure, ambient temperature, and relative humidity for meteorological applications.

With such capabilities in a mobile device may it be in sensing or communication, in this study, we take advantage of these functions for use in emergency situations or observing weather conditions to mitigate risks. The following sections discusses in details the different mobile-phone-based solutions specific to a scenario, such as during damaging disasters or when the need for pinpoint weather observation in the event of a heavy rain arises.

3. Using smartphones to communicate during post-disaster situations

In a devastating earthquake such as the Great Eastern Japan earthquake of 2011, we find that communication channels are most likely destroyed leaving victims with difficulties in contacting their friends and families. In an event of a disaster, victims need to be rescued for at least 72 hours from when it happened. Survival rate typically decreases after this duration for which rescue operations must be hastily initiated.

3.1 Overview and Requirements

We have developed a smartphone-based communication tool called SOSCast, which enables victims to communicate even with the absence of communication channels. It allows for the victims to exchange SOS messages that contains information of their current physical status, location information retrieved by GPS (Global Positioning Satellite), etc., via the smartphone using Bluetooth (BT) communications. Fig. 1 shows a general picture of how victims are capable of sending emergency messages to rescuers using SOSCast on their smartphones.

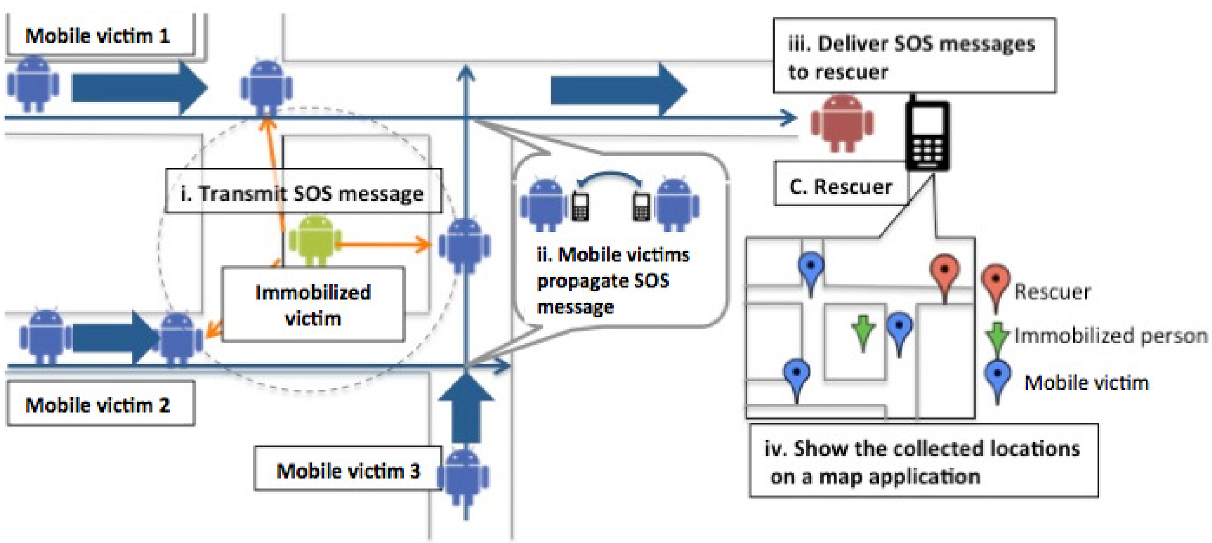


Figure 1: Emergency message sending process using SOSCast

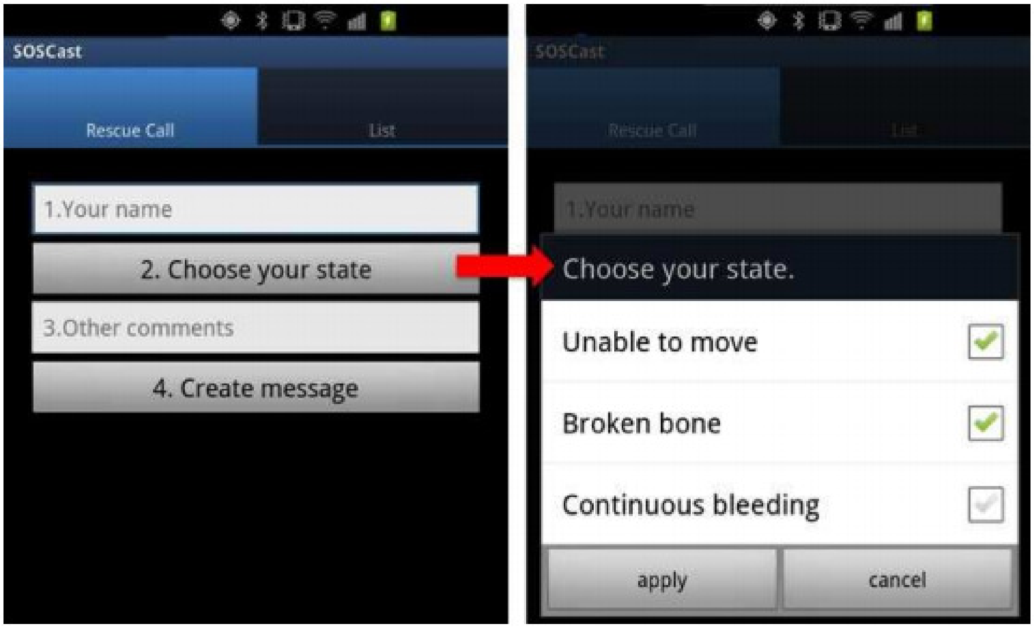


Figure 2: Captured screen images of the SOSCast application on the immobilized victim's device

It is assumed that the application will be utilized in a scenario where the disaster-affected area is cut off from the conventional communication services, i.e., landline and cellular. That is, victims having smartphones with the installed application are able to communicate directly. The process begins by having the victim create his or her SOS message as in Fig. 2.

First, the victim identifies self if immobilized or mobile. If the victim is immobilized, the victim will be asked to create the message that requests the rescue using the smartphone application. We refer to this message in the application as "SOS message". The application makes the SOS message with the needed information

Otherwise, the mobile victim would simply enable smartphone to broadcast a pairing request. After the immobilized victim creates the SOS message, the smartphone must be enabled to broadcast a pairing request as well. While the application is used by a mobile victim, the device continues to listen to pairing requests from other victims using the application. If the mobile victim's device found a pairing request, the device will begin to establish connection. Then, if

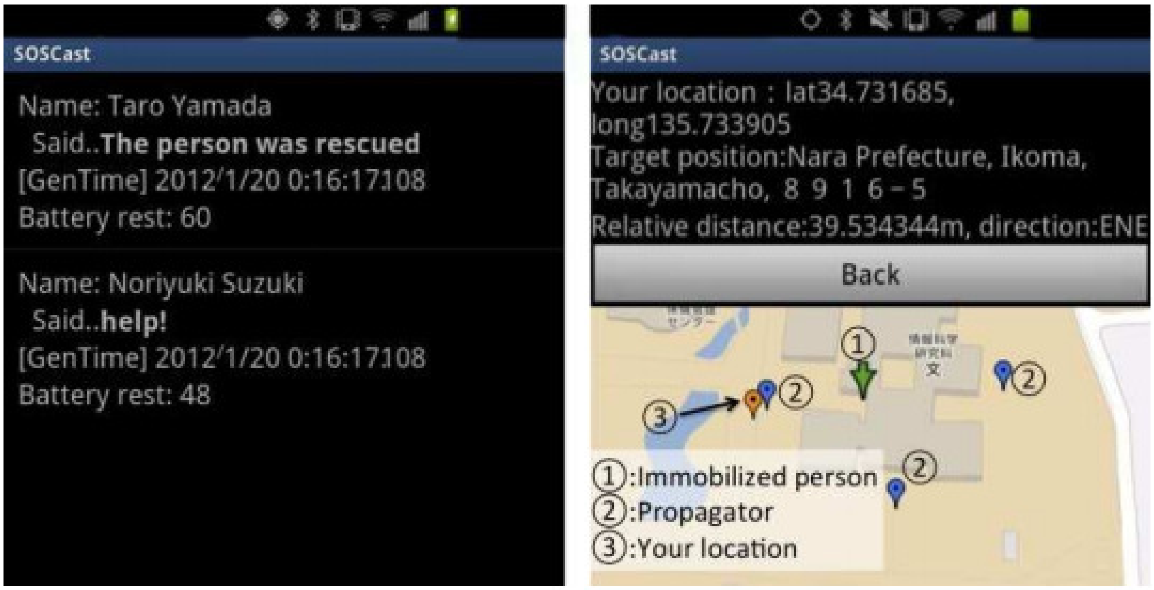


Figure 3: Captured screen images on the SOSCast application on the mobile victim's device

the connection is identified to be with an immobilized victim, the mobile victim's device will wait to receive the SOS message. Simultaneously, the immobilized victim sends the created SOS message to the mobile victim and waits to receive the extended SOS message. At the time when the mobile victim receives the SOS message, the mobile victim will attach personal information to the message, store this on the device, then send back to the immobilized victim the extended message (Fig. 3). Finally, the mobile victim ends the connection with the immobilized victim and broadcast again a new pairing request searching for other victims.

On the other hand, if the mobile victim happen to find a pairing request from another mobile victim or rescuer, the mobile victim on inquiry mode sends a connection request and establish connection with the other party. Then, instead of receiving an SOS message, the mobile victim sends the other mobile victim or rescuer the currently stored SOS messages from immobilized victims with the extended information. Afterwards, the mobile victim waits to receive the stored messages from the other party to update the database in the device. When the exchange of stored messages is done, both parties end the connection and begin to search for other unaccounted victims.

3.2 Message Format

The SOS message formats between immobilized and mobile victims differ in content. The immobilized victim records ID, and remarks on current physical condition. The location information and the time the message was created are also logged. The immobilized victim's ID may include a nickname or the real name of the person. As for the remarks, it is a message field where the immobilized victim selects from a dropdown menu of current statuses with predefined messages. The location field includes the currently identified GPS information by the device. Lastly, the time when the immobilized victim created the message is also recorded. From this, it is possible to estimate the time when the immobilized victim began to ask for help. Moreover, it is useful to sort and delete duplicated messages based on this time field.

The mobile the victim's information includes ID, the current location information, and the time when the mobile victim communicated with the immobilized victim. Similar the to immobilized victim's ID, this field may also include a nickname or real name. The location field logs the current GPS information where the mobile victim has established connection with the immobilized victim. Lastly, the time when the mobile victim has received SOS message from the immobilized victim is logged in the time field.

In general, the number of bytes allocated for each of these information is listed in Table 2. Also, it should be noted that the GPS information for both the immobilized and mobile victims is inaccurate by around 10 meters. The SOSCast application is not capable of identifying the accuracy of the obtained GPS information. However, we rely on this information to at least have an idea where the immobilized victim may be at.

3.3 Propagation Process

Figs. 4 and 5 illustrates how messages are exchanged between an immobilized person and a propagator and between propagators, respectively. While both communication processes exhibit a similar manner of establishing a connection via BT, it is obligatory for propagators to store and share received SOS messages. In SOSCast, by having propagators acquire both of the SOS messages they received

Item	Description	Size (byte)
Immobilized victim ID (IMEI)	Name to identify the victim	15
Message creation time	Time when the SOS message was composed	4
Immobilized victim location	GPS information of the victim	8
Communication time	Time when a connection was established	4
Propagator location	GPS information of the propagator	8
Immobilized victim information	Current status of the immobilized victim	0

Table 2: Description of the SOS message format

and what the other propagators receive, this ensures that all immobilized victims are accounted for when the database is relayed to the rescuers.

Meanwhile, the propagator could possibly encounter multiple BT pairing requests upon traversing the disaster-affected area. In this case, we have thought of a process on how the propagators should be able to accommodate as much as possible all connections. Table 3 shows six types of connection lists, namely, immobilized persons, propagators, and already connected persons classified into either having a strong or weak Received Signal Strength Indication (RSSI). Note that already connected persons refers to persons, including both immobilized persons and propagators, of whom SOSCast has made a connection with. Since a propagator can acquire the RSSI of inquiry responses received, the propagator first classifies them according to RSSI level. For example, if the RSSI of an inquiry response that an immobilized person sent is larger than the predetermined RSSI threshold[1], SOSCast inserts the immobilized persons ID into the list of immobilized persons with strong RSSI. Otherwise, the immobilized persons ID will be kept in the list of immobilized persons with weak RSSI if the RSSI

[1]This threshold was decided by the preliminary experiment whether the connection was stable.

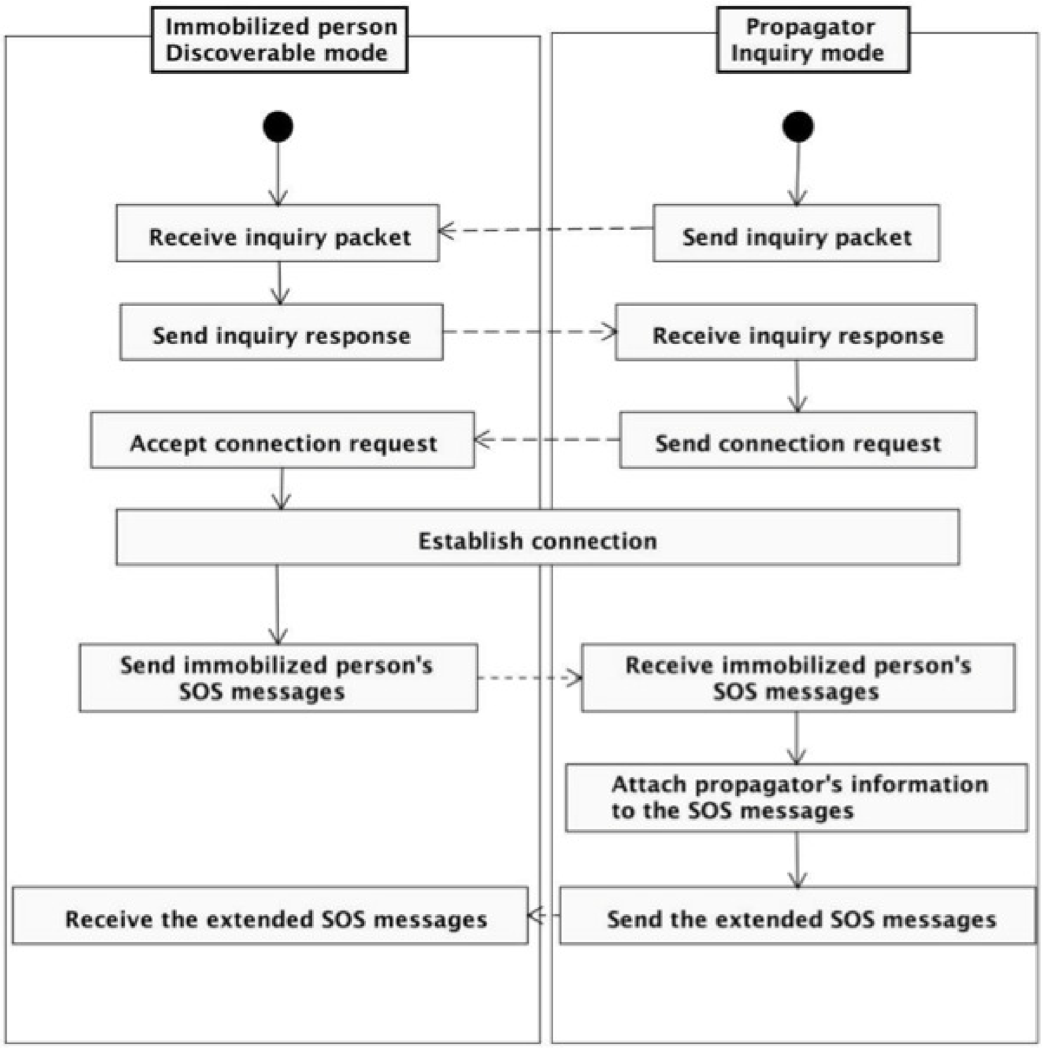


Figure 4: Communication process between an immobilized person and a propagator

is lower than the given threshold. Propagators and already connected persons are similarly classified based on the reported RSSI levels. After making the six lists, SOSCast establishes connection in the following order: (1) immobilized persons with strong RSSI, (2) propagators with strong RSSI, (3) already connected persons with strong RSSI, (4) immobilized persons with weak RSSI, (5) propagators with weak RSSI, and (6) already connected persons with weak RSSI. Each of lists with weak RSSI have low probability of establishing connection, so these lists

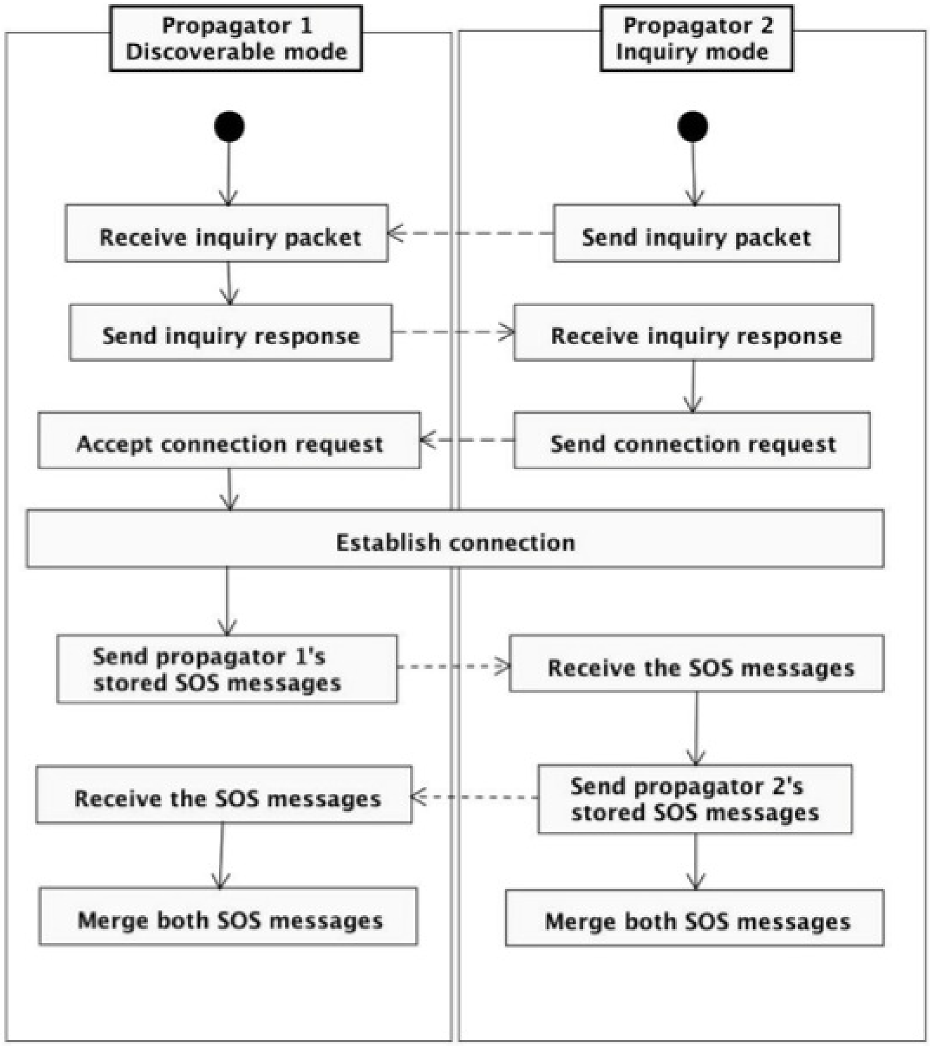


Figure 5: Communication process between propagators

were postponed. Note that the corresponding IDs in each list will be addressed according to the order of the arrival of the SOS message.

3.4 Message Deletion

If an immobilized person were rescued, the SOS messages that the immobilized person distributed will mislead the rescuers as they would continue to search for the immobilized person. To prevent this drawback, unnecessary SOS messages

Priority	Classification	RSSI level
1	Immobilized persons	Strong
2	Propagators	Strong
3	Already connected persons	Strong
4	Immobilized persons	Weak
5	Propagators	Weak
6	Already connected persons	Weak

Table 3: List of SOSCast connection priority

need to be properly removed from the network as soon as possible. Illustrated in Figs. 6 and 7 is the deletion processes of unnecessary SOS messages.

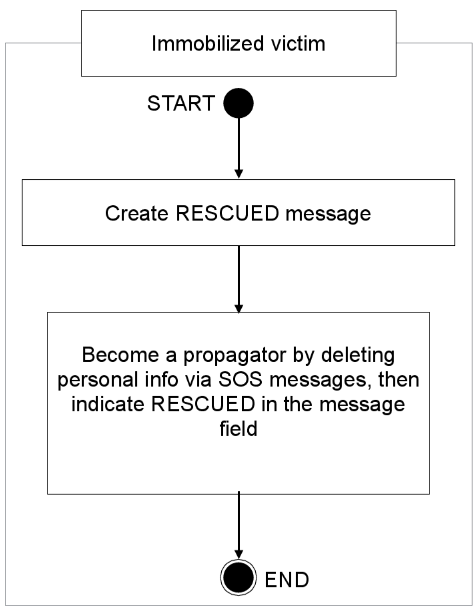


Figure 6: Deletion process in immobilized victim's device

After being rescued, an immobilized person should now become a propagator and continue propagating other immobilized persons SOS messages. During this change in status, the immobilized person first deletes the information relating to its own information from the holding SOS messages and indicate RESCUED

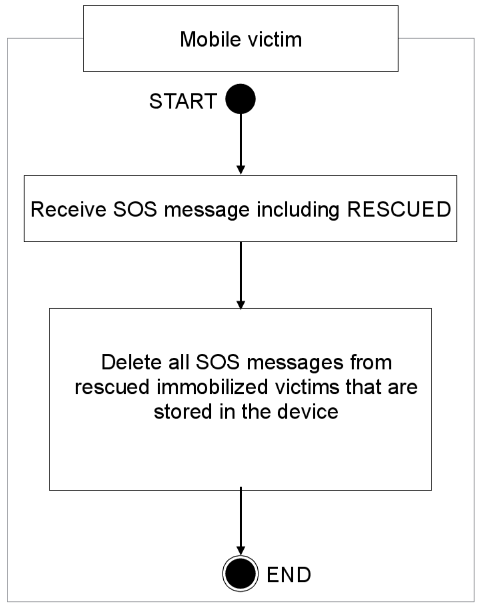


Figure 7: Deletion process in mobile victim's device

in the message field. The new propagator, i.e., the rescued immobilized person, may now be able to exchange information upon meeting another propagator. If the received SOS messages include rescued information, each propagator removes the related information from the holding SOS messages according to the rescued information. However, some people who had sent SOS messages and rescued, may need to call rescue again by some reason like secondary disaster. They can also create SOS messages, and these messages are not deleted by the old rescued message. Note that the rescued information is maintained in the SOS message. Therefore, rescued immobilized persons' information is deleted during propagating process.

3.5 Location Estimation

Based on the logged SOS messages, the rescuers can easily estimate the immobilized victims' location from the recorded GPS location information. Actually, SOS messages will be propagated via mobile victims until these messages reach the rescuers. Note that, although SOSCast provides the same functions to both mobile victims and rescuers, the difference lies in the capability to actually search for immobilized victims. SOSCast can display locations of immobilized persons and propagators on a map from collected SOS messages for the purpose of providing aid to the rescuers. In the display, if the GPS information of the immobilized persons SOS message is incorrect or missing, the information will be useless. To prevent that, the SOSCast also collects propagators' GPS information where having received it. Thus, as the number of propagators' GPS information increases, the location of the immobilized person will be estimated correctly.

3.6 Prototype implementation

To observe how the SOSCast application actually works, we conducted an experiment using Android OS-based smartphones. First, the experimental environment is discussed then the results are explained in the following sections.

Environment

Firstly, based on the above design, we implemented SOSCast as an Android application. Then, we conducted a preliminary experiment in a residential area to confirm the potential of SOSCast. As shown in Fig. 8, an immobilized person is presumably trapped in a house while propagators walk along the street five times in each of the five directions. Note that all participants have installed SOSCast on their smartphones (Samsung Galaxy Tab SC-01C). Finally, the locations of both the immobilized person and propagators will be indicated on the map when a rescuer collects the entire SOS messages from the propagators.

Results

Shown in Fig. 8 are the locations of the immobilized person and the propagators on a map which the rescuer can view on his or her own smartphone. During the experiment, all propagators were able to communicate with the immobilized person for every trial. As seen from the figure, the propagators were able to receive SOS messages from the immobilized person within 10 m in average along the directions 1, 2, and 5. Furthermore, even if the indicated location of the captured information is more than the average as in the case of directions 3 and 4, it is important to note that this implies the existence of an immobilized person within the indicated perimeter. Also, even if an immobilized person has incorrect GPS information or lacks it altogether, rescuers can estimate the location of the immobilized person by utilizing the GPS information of propagators.

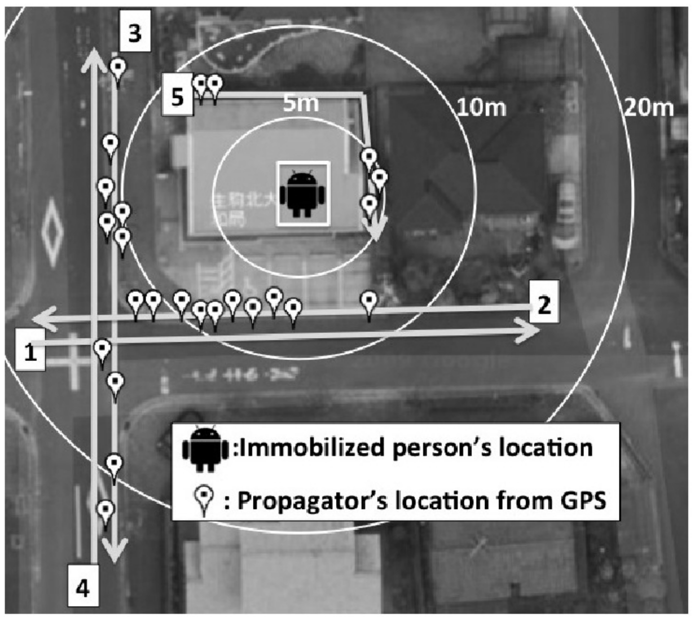


Figure 8: Experimental result on using the SOSCast application

3.7 Issues with SOSCast

Since SOSCast had some problems with it during the first application trial, improvements were accordingly done to make it more efficient. One of the problems we found with SOSCast is that it has a limited search coverage. In the aforementioned experiment scenario, the walls of the building as a physical obstruction have drastically lowered the communication range. We cannot expect a maximum range in a disaster-affected area where there are possibly many obstacles. As the device coverage area may be limited to 10-20 m in radius, it is possible

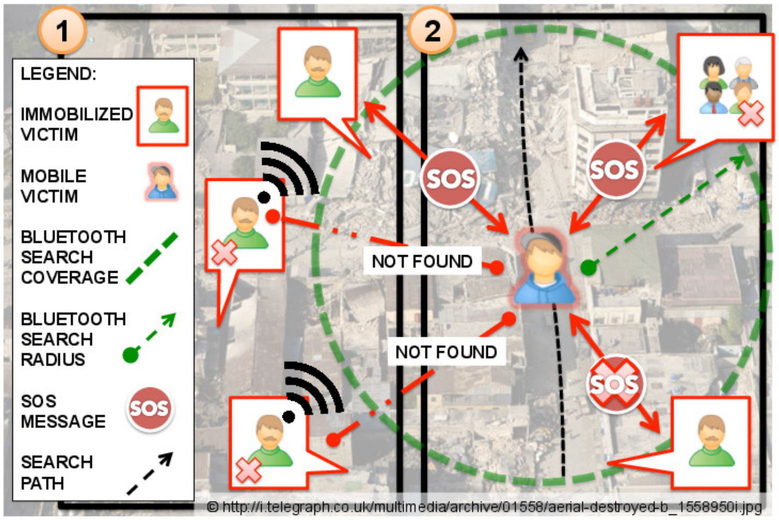


Figure 9: The problems of the original SOSCast

that some messages from the immobilized victims are missed. In Fig. 9 (1), for example, while the mobile victim (or possibly a rescuer) walks through the search path, his or her device may not be able to communicate with the immobilized victim's device (victim marked with "x"). Since the immobilized victim is located outside the BT search coverage of the mobile victim's device, it is not possible to establish a connection between the devices. Thus, when the mobile victim has passed through the search path, he will miss receiving the immobilized victim's SOS message.

In another case (Fig. 9 (2)), the SOS message will be missed when establishing a connection has not been completed like when there are multiple immobilized victims in one area. Since SOSCast requires a one-to-one connection, the mobile victim may not have sufficient time to establish a connection with each of the immobilized victims. Similar to the previous scenario, the immobilized victims will eventually be outside the BT search coverage area when the mobile victim has passed the search path.

SOSCast typically uses BT to communicate with other nodes. In order for mobile victim nodes to receive SOS messages from immobilized victim nodes, a communication has to be established via BT[2]. Also, as the SOSCast automatically

[2]It takes about 1-5 seconds to establish a BT connection before starting communication

collect SOS messages, mobile victims are not involved in collecting them.

As an example, when the mobile victim has left the immobilized victim outside the device coverage, the immobilize victim will continue to broadcast SOS messages until the opportunity of another mobile victim passing nearby. In this case then, the immobilize victim will continue using the device and eventually use up battery power and decrease the probability of being found. In another scenario where there are several immobilized victims present in a similar point location, the mobile victim node may not be able to accommodate all connection requests one at a time. As the mobile victim continues to move farther, some connections may not happen and eventually some immobilize victims may not be accounted for. Therefore, it is important that we have to increase the opportunity that a connection is established between devices of both mobile and immobilized victim. As such, by conserving battery, there is a higher chance that an immobilized victim can be found due to the increased chances of transmitting SOS messages.

3.8 Summary

The basic implementation and design of sending SOS messages by victims of a very damaging disaster, namely the SOSCast application is discussed in this section. By mapping estimated locations, we have shown that SOSCast has the potential to locate immobilized persons based on this information. In SOSCast, victims collect and propagate these messages which rescuers can use to estimate locations of the immobilized persons by using the smartphone. In a disaster area where conventional communication services are impaired, SOSCast can potentially aid the work of rescuers as it can enable direct communication between smartphones.

between nodes and exchange data.

4. iSOSCast: Enhanced SOS message collection in disaster-affected area

Improved SOSCast (iSOSCast) was developed to address the problems experienced from the actual experimentation of the original SOSCast. In this section, we present the construction of an information cluster of immobilized victim nodes as a solution. The goal, thus, is to increase the opportunity of locating victims in a disaster area by virtually extending the search coverage while saving battery consumption.

To address the problems mentioned in Sec. 3.7, we propose the construction of an information-sharing cluster (see Fig. 10) among immobilized victim nodes. Making a cluster presents two contributions. Firstly, SOS messages of all immobilized victim nodes in a cluster can be sent to a mobile victim node all at once. Having this feature reduces the number of communications. The second contribution is that the designation of representative nodes, which is explained later on, virtually extends the communicable area of immobilized victim nodes. Thus, such a clustering method improves the opportunities to connect with mobile victim nodes and simultaneously extends the search coverage area virtually while prolonging battery life. Moreover, this method ensures that the collected messages withstand transmission to authorized rescue teams.

4.1 Selecting representative nodes

To efficiently make a cluster while lowering battery consumption, we begin by employing representative nodes among immobilized victim nodes to collect SOS messages within one hop. Each node promotes an immobilized victim node with the highest remaining battery within one hop as a representative node. Each node first sets a new BT device name including its own node ID, remaining battery level, and blank representative node ID. After conducting a BT device search, each node identifies a node with highest battery among the group, including itself, as a *representative*. Each node then records the node ID of the promoted node in the representative node ID. For instance, in Fig. 10, nodes b, c, and f are representative nodes. Then, nodes a and d send their SOS messages to c, which sends its SOS message to b. Nodes e and g then send their SOS messages to f.

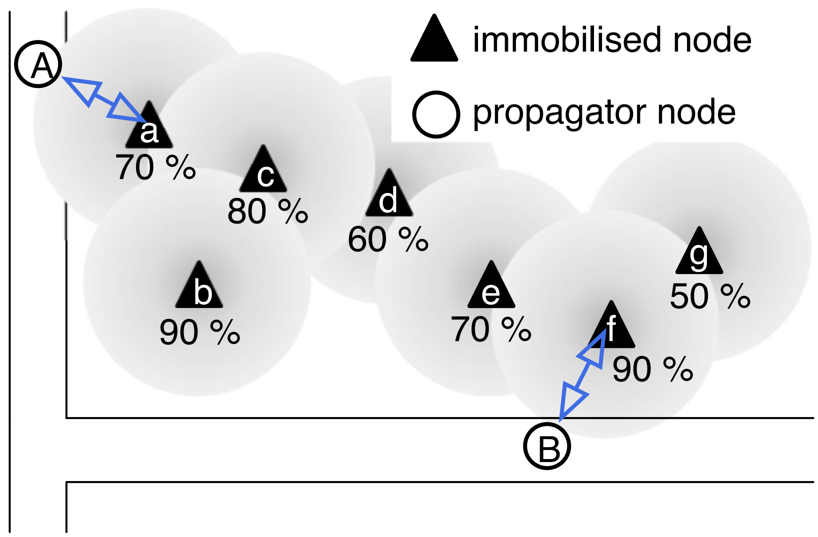


Figure 10: Information clustering process

Each representative node then stores the received SOS messages. Note that, as nodes b and f are nodes with highest battery levels within their one hop neighbor, they do not send their SOS messages. After selecting a representative node, a normal node, i.e., non-representative node, directly sends its SOS message to its representative node without the BT device search.

Considering only the above step, the SOS messages of all nodes may be divided among multiple representative nodes because a representative node is passively selected. Moreover, each immobilized victim node regularly monitors their respective battery levels and constantly compares it with one-hop neighbor nodes. If, for instance, the battery level of the current representative node goes below a predetermined threshold, the immobilized victim nodes in the cluster select a new representative node following the same process. In our case, we have chosen 15% as the threshold since Android-OS based phones typically prompt the user to charge the device upon reaching 15%. If the remaining battery of all nodes is below that threshold, the selection process is not executed. In the experiments, we measure the lifetime of all the immobilized victim nodes until battery power runs out.

Table 4: Distribution of node roles

NODE	**a**	**b**	**c**	**d**	**e**	**f**	**g**
Representative node	c	b	b	c	f	f	f
One-hop neighbor nodes	c	c	a, b, d	c, e	d, f	e, g	f
Two-hop neighbor nodes		(c, a), (c, d)	(d, e)			(e, d)	
Neighbor representative nodes		c	b, (e, f)			(d, c)	

4.2 Message sharing among representative nodes

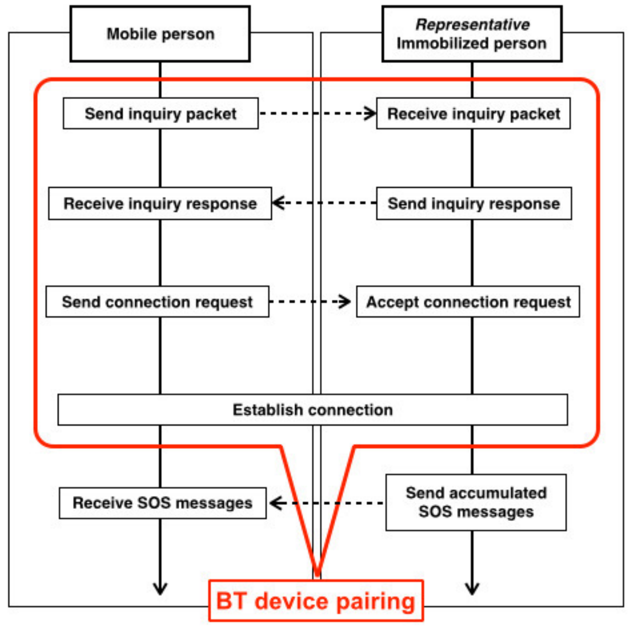
When SOSCast is used, each representative node is assumed to hold all SOS messages collected within a particular area. In the following discussion, we explain the process of sharing stored SOS messages among representative nodes. When collecting SOS messages from one-hop neighbor nodes, a representative node can also obtain the information of two-hop neighbor nodes because the SOS messages include the information of one-hop neighbor nodes, as shown in Table 4.

Note that, the notation of (c, a) shows that node c connects with node a. In this case, node b recognizes nodes a and d within two hops, c knows e, and f knows d. Then, since node b needs to know the representative node for nodes a and b to share the SOS messages with neighboring representative nodes, node b sends the stored SOS messages to a and d via c. However, as node c is the representative node for a and d, node c updates the stored SOS message based on received SOS messages and then sends the updated SOS messages back to node b. By receiving the updated SOS messages from c, node b knows that c is the representative node for nodes a and d. In the case of node f, it sends the stored SOS messages to d via node e. When receiving the SOS messages, node d forwards them to node c as it is the representative node for node d. After receiving them, node c sends the updated SOS messages to nodes b and f and, consequently, all representative nodes have shared all the SOS messages. Note that, when the stored SOS messages are being updated, a representative node can send the updated SOS messages to neighbor representative nodes. By constructing an information-sharing cluster, every normal node can obtain all SOS messages from their representative node within one hop. Moreover, the battery consumption of each node decreases due to reduction of the number of communications. Lastly, the communicable area is extended virtually.

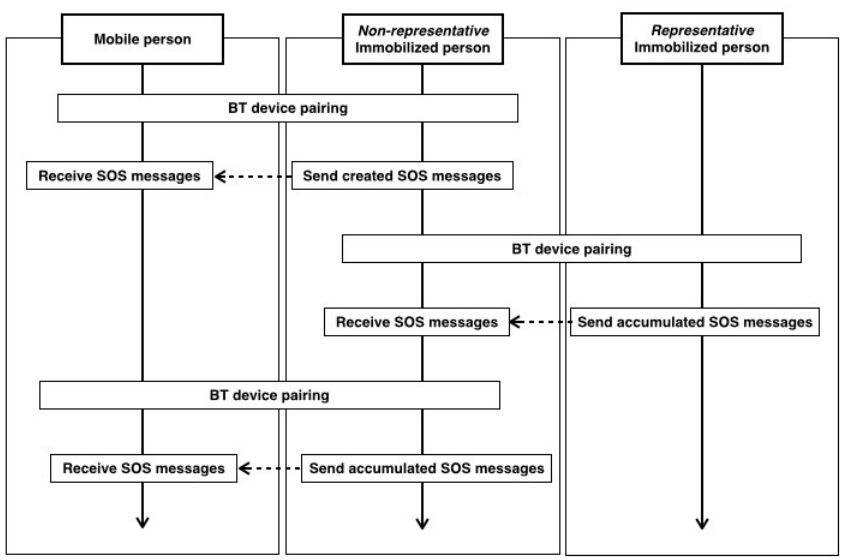
From the process, a mobile victim node can obtain the SOS messages from a representative node all at once (Fig. 11a). Even if a mobile victim node cannot connect with a representative node directly, it can obtain them from a non-representative node in a short amount of time because the non-representative node can forward the data from the representative node within one hop.

4.3 Communication between mobile node and non-representative node

Figure 11b illustrates the process of how a mobile node can indirectly obtain the accumulated SOS messages via a non-representative node. Generally as in Fig. 11a, the immobilized victim activates the BT device into discoverable mode and broadcasts BT pairing while the mobile node activates the BT pairing request. When the immobilize node receives an inquiry packet, it will send an inquiry response to the mobile node that requested for BT pairing. The connection begins when the immobilized node approves the connection request from the mobile node. The representative immobilize node then sends the accumulated SOS messages to the mobile node. As for BT usage between mobile node and non-representative node, the same process occurs with the difference in an added connection with the representative immobilized node. Similarly with the communication process between a mobile node and a representative node, the non-representative immobilized node first sends its own SOS messages to the mobile node. Then, it requests for a connection with the nearest representative immobilized node within one hop. Instead of the representative node directly sending the accumulated SOS messages to the mobile node, it sends these first to the non-representative node. When the non-representative immobilized node has received the SOS messages, it will request a connection with the mobile node. Finally, the mobile node receives all SOS messages from the cluster.



(a) Communication process between mobile person and representative immobilized person



(b) Communication process between mobile person and representative immobilized person via a non-representative node

Figure 11: Communication processes among different nodes

4.4 Simulation Results and Analysis

Evaluation via prototype systems

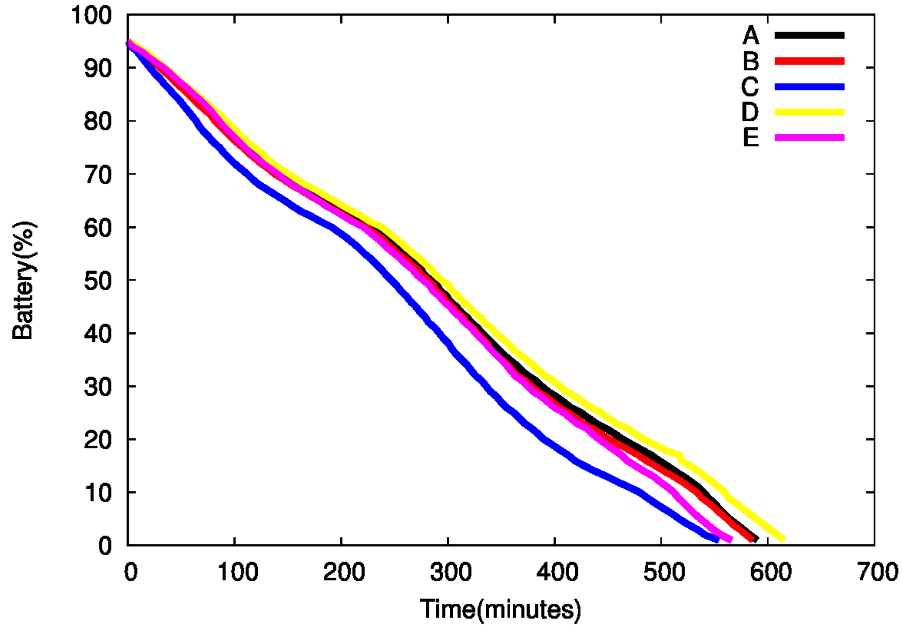
In a disaster area, immobilized victim nodes need to save battery as long as possible in order to increase the probability of being rescued. To investigate the lifetime of the smartphone using SOSCast, the implementation of the previous method was compared with the proposed information-sharing cluster method. In this experiment, six devices were set up in one location where one of the devices was assigned as the mobile victim node and the rest as the immobilized victim nodes The five immobilized victim nodes initially have full battery and each device communicate to each other via BT.

In the experiment with the previous method, the mobile victim node communicates to each immobilized victim node every minute in order to obtain an SOS message. With the proposed method, on the other hand, the immobilized victim nodes first elect a representative node among them. Then, the mobile victim node directly obtains the SOS messages of all immobilized victim nodes from the representative node once every minute.

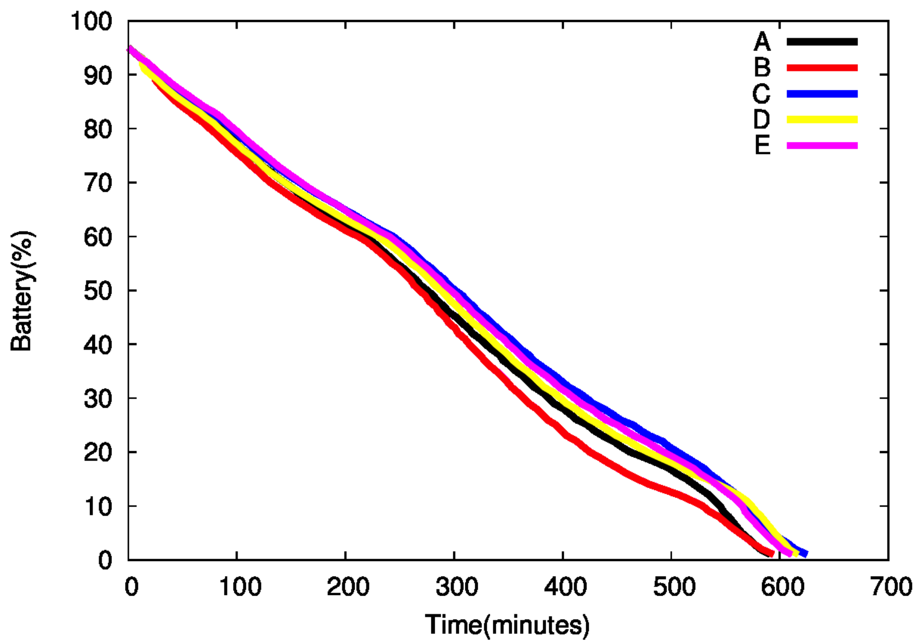
Figures 12a and 12b show the running time of the nodes implementing the previous and the proposed methods, respectively. Whereas, Table 5 gives the running time of each node in both figures. From the table, we can see that the running time of some immobilized victim nodes in the proposed method is extended. In the previous method, as each immobilized victim node does not share their SOS messages, an immobilized victim node stops sending its SOS message when it runs out of battery. At this time, the SOS message of the node is not distributed anymore.

Table 5: Comparison of running times of each node between the previous and proposed methods

	Node A	Node B	Node C	Node D	Node E
Previous (minutes)	590	586	554	615	586
Proposed (minutes)	591	594	625	617	611



(a) Using the *previous* method



(b) Using the *proposed* method

Figure 12: Node running time results in both methods

In the results, SOS message distribution by Node C is terminated after 554 minutes, while that of Node D is 615 minutes. On the other hand, since each immobilize victim node shares SOS messages among them as in the proposed method, their SOS messages are distributed until Node C runs out of battery after 625 minutes. Also, since the mobile node does not need to connect with all nodes, it contributes to conserving the battery of mobile victim nodes by at least 2-8 % in the experiment.

Evaluation via simulation experiments

In the prototype system, we cannot exactly evaluate the scalability of the method. As an alternative, however, we evaluate the performance by having a large number of nodes and provide the results though a simulation experiment with Scenargie simulator. The simulation is needed to observe the difference of proposed SOSCast with the original SOSCast. To illustrate this comparison, we show results of the number of messages sent per device and percentage of battery consumption. We focus on these parameters so we can compare the amount of load between the original SOSCast and the proposed SOSCast, in a group of smartphones in general. Evaluating the lifetime of several smartphones is insignificant in this case as it typically varies on how the devices are used in real situations such as the type of smartphone, current position on the user, etc. In addition, the communication device was set to use the abstract model with varying number of terminals from 16 to 2,500 units for every run. Each simulation lasts for a period of 36,000 sec when all messages are completely accounted for. Furthermore, the message size of the SOS message sent by each device was about 40 bytes. This value includes 15 bytes of device ID, 4 for the message, 8 for device GPS information, 4 for the time communication was established, 8 for the GPS information of the other device, and an optional 1 byte for the information on the main rescuer. Each terminal is approximately located 10 m away from each other with a potential one-hop connection with neighboring nodes as in the model in Fig. 13.

Fig. 14 shows the comparison of the proposed SOSCast method and original SOSCast routing in terms of the number of messages sent. For example, for approximately 1000 nodes, the average number of messages sent by each device is less than 100 when using the proposed SOSCast method. Whereas, when

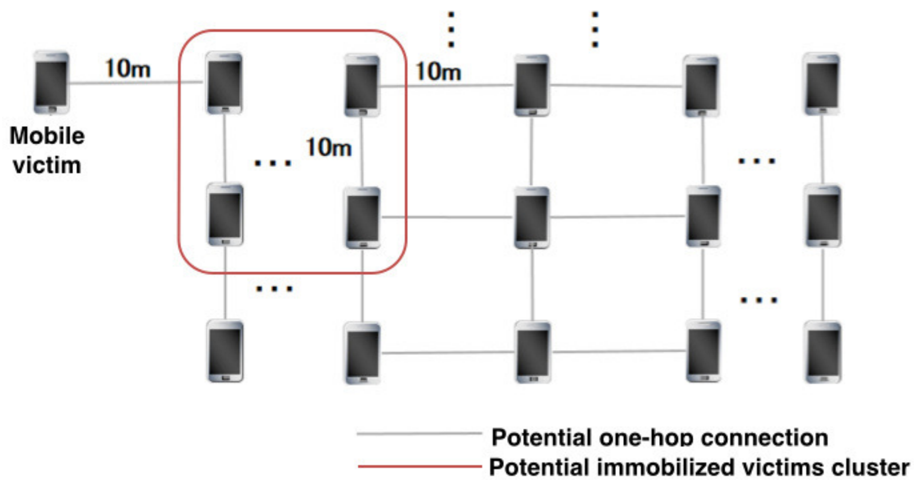


Figure 13: Simulation model

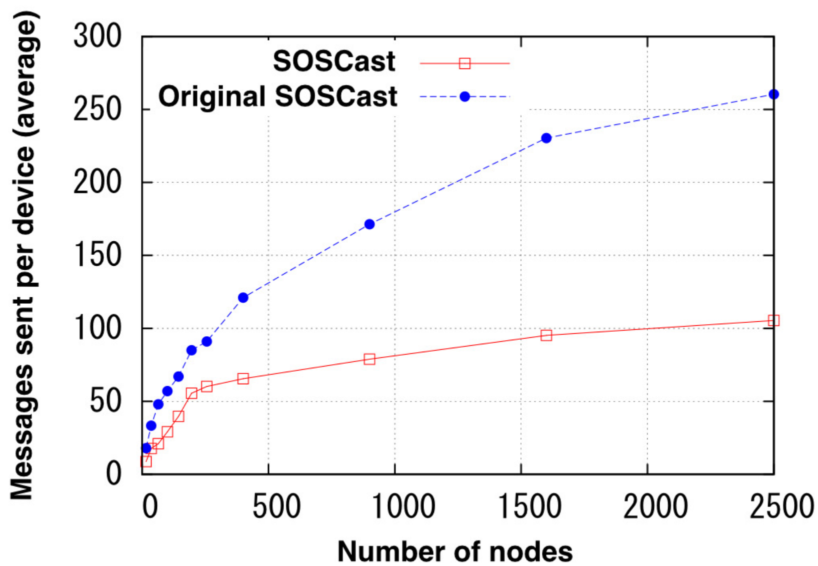


Figure 14: Average number of messages sent per device for a corresponding number of nodes

using the original SOSCast, the average number is more than 150 messages. The graph in general shows that SOSCast with the information-sharing cluster method requires less messages to send to account for all message from all nodes.

In another simulation result as in Fig. 15, the percentage of battery consumption is compared between using the proposed SOSCast method and original SOSCast. The results of device battery consumption by representative nodes and non-representative nodes (or others as in the figure) is further graphed in detail. With the proposed SOSCast, the representative nodes clearly consumes more battery than the nodes using the original SOSCast and the non-representative nodes or others. As we employed representative nodes as a new factor, we found the need to further investigate this factor in contrast with non-representative nodes. That is, considering that representative nodes need to communicate with other representative nodes aside from the non-representative nodes, then it will obviously have to consume more power. Note that non-representative nodes only need to with the respective representative nodes by one-hop, which results to lower batter consumption. With the combination of both representative and non-representative nodes, the overall average battery consumption is reduced compared with the original SOSCast.

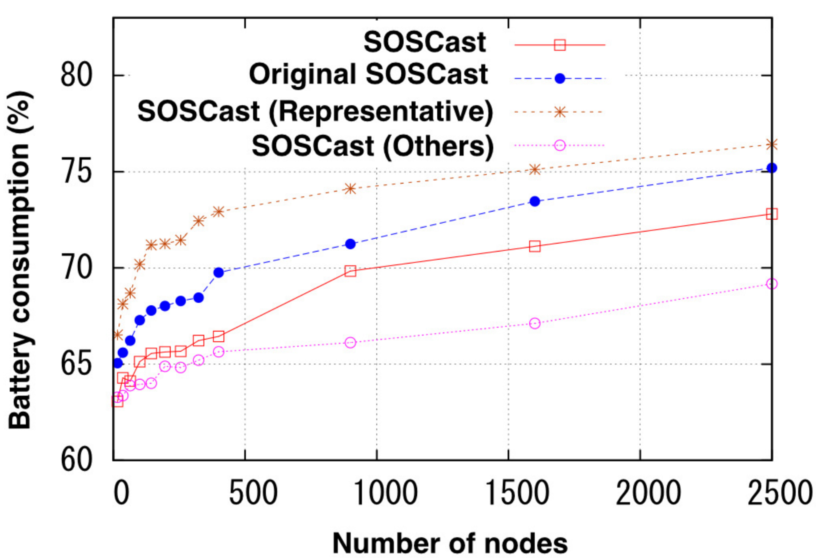


Figure 15: Percentage of battery consumption

4.5 Summary

This section discussed an information-sharing cluster technique with reduced battery consumption in SOSCast. The aim is to effectively collect SOS messages from immobilized victim nodes in a disaster area with collapsed communication service. The SOSCast application was developed to support rescue operations for finding immobilized victims. However, upon extended research on SOSCast, we found that it is necessary to improve the search coverage for immobilized victim nodes and find a way to collect SOS messages from immobilized victim nodes while saving battery. Also, it is important that the SOS messages are kept until all nodes disappear. Thus, as a solution, we propose the implementation of an information-sharing cluster in SOSCast as an improved feature. In this way, we ensure that all existing SOS messages are accounted for, and that these collected messages until it reaches the authorized rescue teams even when all nodes run out of battery. Evaluation by actual measurements was performed by comparing the battery consumption of the SOSCast with information-sharing cluster and the prior SOSCast. Moreover, performance evaluation was also done via simulation where the battery consumption of the proposed SOSCast is compared with the original SOSCast. Overall, the number of messages sent per device was reduced and the battery consumption was less when using SOSCast with the information sharing cluster compared with the original SOSCast. By extending the battery life, the survival of these SOS messages is also increased.

5. Using smartphones to gain weather information for pre-disaster preparation

As mentioned previously, mobile communication devices or smartphones these days have sensing and communication capabilities. With these functions, we are able to observe current weather conditions at point locations whilst expecting extreme events and anticipating the risks, for instance, in an event of an extremely heavy rain.

First, when we say heavy rain, we refer to rain events that have rainfall rates ranging from 30-50 mm/hr for up to 200 mm/hr in extreme cases. These type of rain events are typically highly localized, approximately within 1-5km area, due to regional heating as a byproduct of greenhouse effect especially in highly urbanized places. When it suddenly rains heavily, which commonly happens for only within an hour, there are risks of flash flooding or land mass movements (landslide) [38]. With insufficient lead time for preparation, the risk of inconvenience and getting caught up in the disaster is highly likely.

Moreover, current sophisticated systems that monitor these potentially dangerous rainfall systems are not effecient for most of the time. One reason is that they sometimes can be very costly. Moreover, these systems usually cover thousands of kilometers and are unable to pinpoint the heavy rain events occuring in a few hundred meters. These system are also deployed in a few kilometers from each other, unable to cover all areas that needs monitoring for these localized heavy rains. Take for example the AMeDAS environmental monitoring system in Japan where a network of 1,300 weather stations are deployed all over the country with each station averagely distanced 17km from each other [19]. Some developing countries like the Philippines have sparse, if not available, weather stations [6]. While radars and satellites are helpful in describing cloud presence and density, it may still be insufficient in determining highly localized rain systems such as squalls. This is especially dangerous in ubranized areas as with the climate trend, considering that population density is very high putting civilians vulnerable to the risk of city flooding. In the following discussions, Section 5.1 describes how smartphones were used to investigate the influence of localized heavy rain on the device. Section 5.2 explains how the radio signals were processed to find the re-

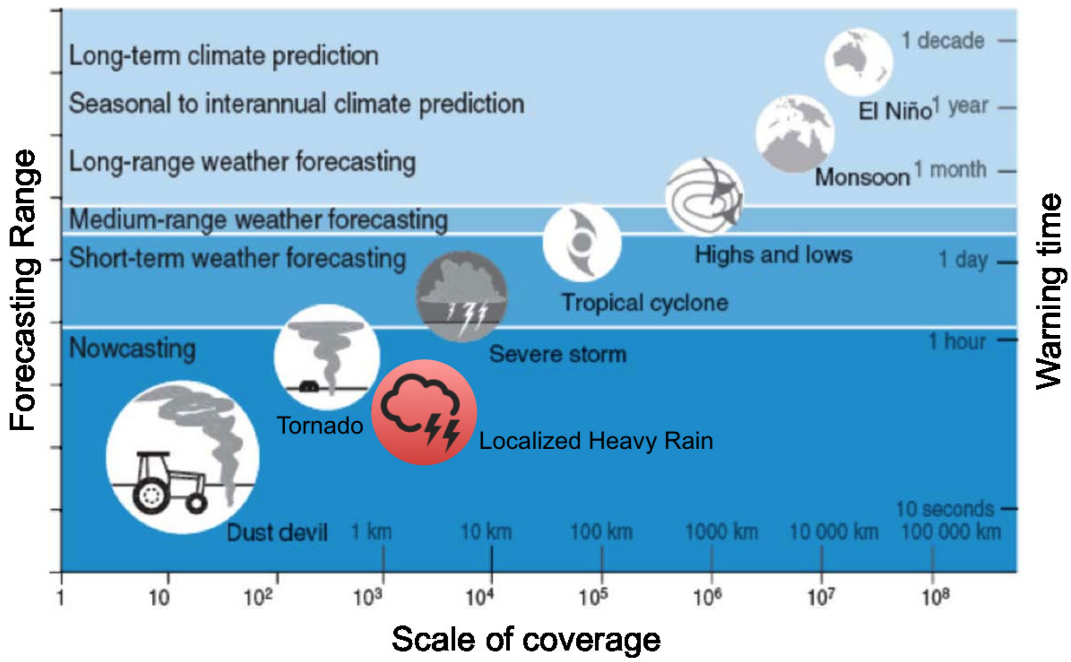


Figure 16: Different weather types according to lead time, duration, coverage area

lationship between smartphone-based measurements and heavy rain. Section **??** describes how smartphones can be used to support existing commercial sensors. Lastly, Section **??** discusses one of the ways to ensure that smartphone-based measurements are calibrated accordingly as a supporting information.

5.1 Measure the influence of localized heavy rain by using smartphones

Despite the limited equipment and information resource, as with the case of Philippines for example, there may be an alternative information resource that be harnessed. Due to increasing demand for mobile communication, service providers began putting up more and more microwave links to provide channels to users even in the remotest of locations. These towers typically service to WiFi and 3G communications which are sensitive to rainfall at very high frequencies. If the received signal level (RSL) is measured, there is a noticeable decrease in the distribution in an event of a heavy rain passing over the link [24]. In this regard,

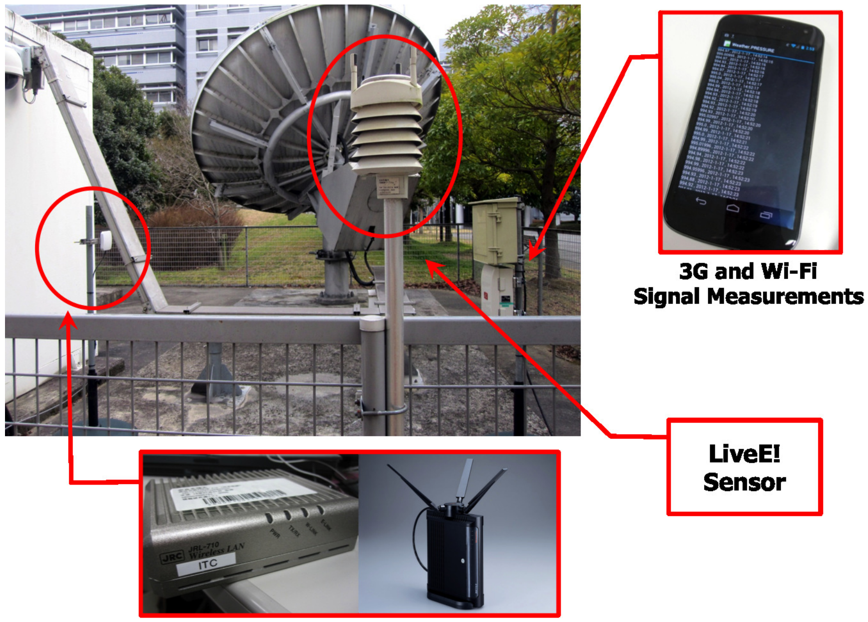


Figure 17: Preliminary measurement setup

the investigation of RSL measurements of Wi-Fi and 3G as observed by the smartphone was conducted inside the campus. *Generally the goal was to be able to design a warning system for localized heavy rains that is easily implementable and low cost, i.e., by use of the smartphone. We particularly want to implement such warning system for developing or least developed countries with little or no access to sophisticated weather observation equipments.* To find out if RSL in smartphones are influenced by localized heavy rain, we set up the following experiment as in Fig. 17.

The radio signals were measured using an Android-based smartphone[3] located a few meters from the Wi-Fi access point (AP)[4] . The set-up is deployed nearby a Live-E! sensor[5] to obtain in-situ weather information on the location. The

[3] Samsung P1000 Galaxy Tab with Android OS v2.2 (Froyo) and Samsung Galaxy Nexus with Android OS v4 (Ice cream sandwich)

[4]For the 2.4 GHz link, we used a JRL710 AP2 802.11 b/g Wireless LAN AP while that of the 5 GHz link was a Buffalo 802.11 a/b/g/n Wireless LAN Broadband router (WZR-AMPG300NH)

[5]This particular Live E! sensor was a Vaisala Weather Transmitter WXT520

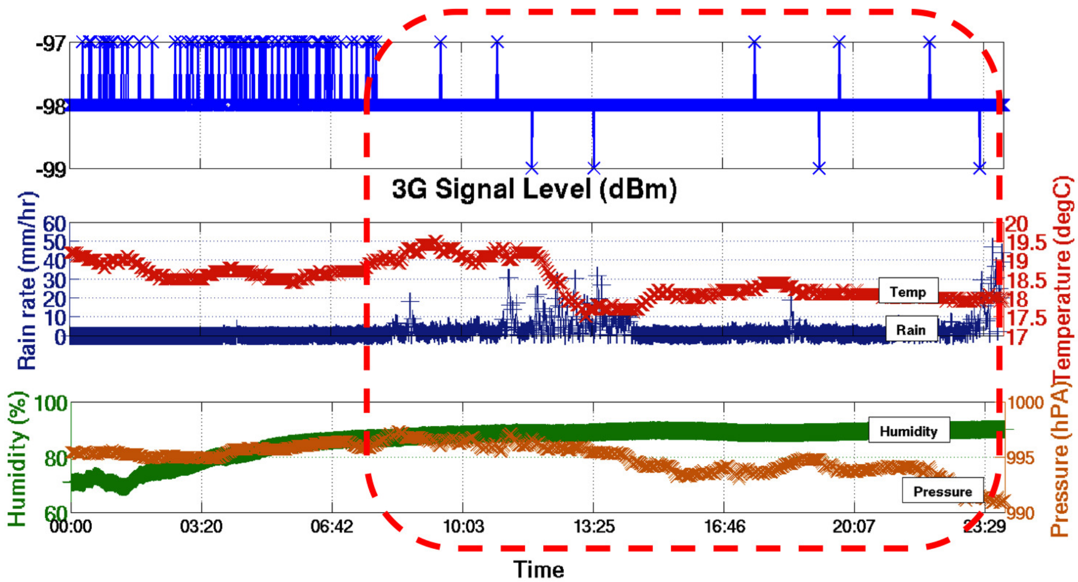


Figure 18: 3G RSL measurement results

radio signals were measured by an application developed on the smartphone to constantly observe and log the data locally. To obtain the 3G signal strength ([45],[18]), we obtain the Arbitrary Strength Unit (ASU) and derive the UMTS signal value in dBm by the following conversion: dBm = ASU – 166. ASU is a constant number that describes the mobile phone signal strength and for UMTS networks, it ranges from -5 to 99. As for the 2.4 GHz, the smartphone was measuring signal strengths of the beacon signals from the access point. The application directly provides the real signal strength in dBm Wi-Fi signals and the obtained information is stored locally as well. These measurements were done every minute on a daily basis independently as in the setup.

Note that in this experiment, we were observing three kinds of communication channels mainly 3G, 5GHz and 2.4GHz WiFi. In general, 3G measurements showed limited effect on signal attenuation under rain condition as shown in Fig 18.

The 3G-network provider under investigation was of NTT Docomo FOMA, W-CDMA/HSDPA/HSUPA with 2100 MHz frequency. One of the results for example on June 21, 2012 rain event, reveal fluctuations on the signal between -97 and -99 dBm when it was raining at 30 mm/hr and. Compared with fluctuations

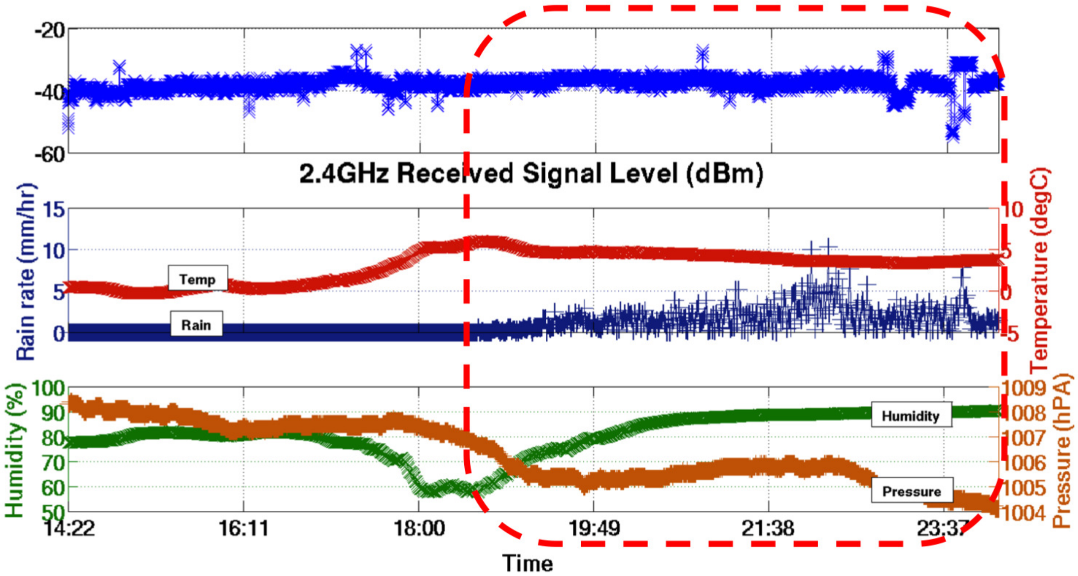


Figure 19: 2.4G Wifi RSL measurement results

from -98 to -97 dBm when there was no rain, these effects are currently unclear as to whether it was mainly affected by rain or not.

In the rain event of January 19, 2012, rainfall rates peaked for up to 10mm/hr (Fig. 19).

However, the intensity was insufficient to be considered heavy rain as to significantly affect the RSL of the 2.4 GHz WiFi. Thus far, we are unable to determine clearly what could have been the effect for this frequency in an event of 30mm/hr or more rain event.

Meanwhile, 5 GHz RSL measurements showed obvious effects of heavy rain as in Fig. 20. At the maximum rain rate of 64 mm/hr, a 4 dBm attenuation was observed. Overall, however, there is a need for more investigation with multiple links and frequencies considering the few chosen results.

5.2 Data processing by sample measurements of 5GHz WiFi

While 5GHz WiFi has proven to be useful among other communication channels, we used the data to understand further how RSL degradation can be linked to localized heavy rainfall. This section discusses methods of preprocessing raw RSL

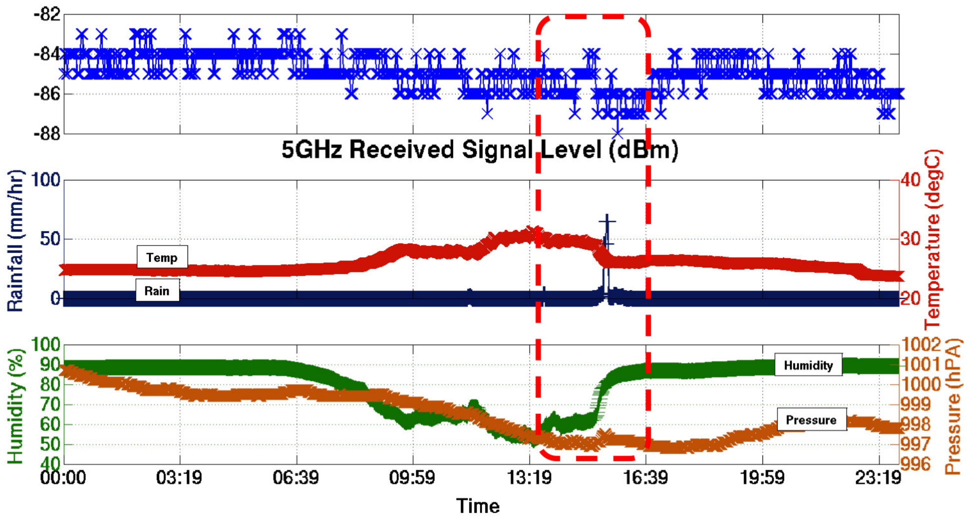


Figure 20: 5GHz Wifi rsl measurement results

data in preparation for exploratory data analysis and cross correlation function to find out its relationship with rainfall rate.

Sliding Window

The original measured 5 GHz WiFi data are RSLs observed every minute from beacon signals received from the AP. On a normal basis, the RSL in 5 GHz would usually be observed between -100 to -80dBm. However, in other data, the values reached up to -200dBm as in Fig.21. This is unusual for radio signals especially that of Wi-Fi since the lowest receiver sensitivity [19] of the WLAN receiver (Broadcom BCM4329) is -90 is dBm. We found that in Android development, the instance of such a value in fact indicated a dropped signal. This value is in fact pre-assigned and hardcoded in the program when it detects no signal from the source. As such, the moving median window method had to be applied to eliminate this phenomenon. This method, as illustrated in Fig. 21, is done such that the median is determined for every 5 for up to 10 samples per window (window size, WS).

The window is then moved by a certain number of units (overlap size, OS) until it has gone through all the samples.

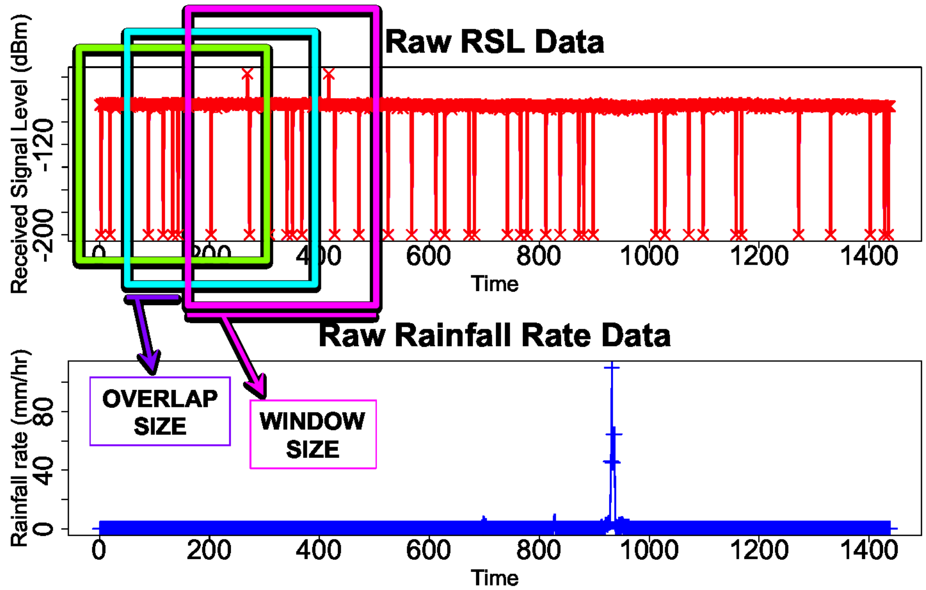


Figure 21: Details when processing raw RSL data

Similarly applied to RSL, the weather parameters were also given the same treatment to preserve consistency in analysis and sample quantity. To determine which WS and OS would significantly retain the original RSL information, we investigated with different combinations to see which combination would result to samples nearly similar to the original. Also, by choosing the appropriate WS and OS, we retain significant values from the original data set. Results showed (Fig. 22) that by increasing both values simultaneously or one at a time would decrease the number of produced samples.

For this case, the minimum value of 5 units for WS with WS-1 as the OS was considered for the instantaneous change in weather conditions for guerilla rain.rain event with WS=5 and WS=10 with both having OS=WS-1.

Exploratory Data Analysis (EDA)

The primary approach to compare and contrast smartphone data with weather-related parameters is to perform several methods of EDA. Firstly, a scatterplot of the RSL in comparison with temperature, humidity, pressure, and wind is plotted in Fig. 23.

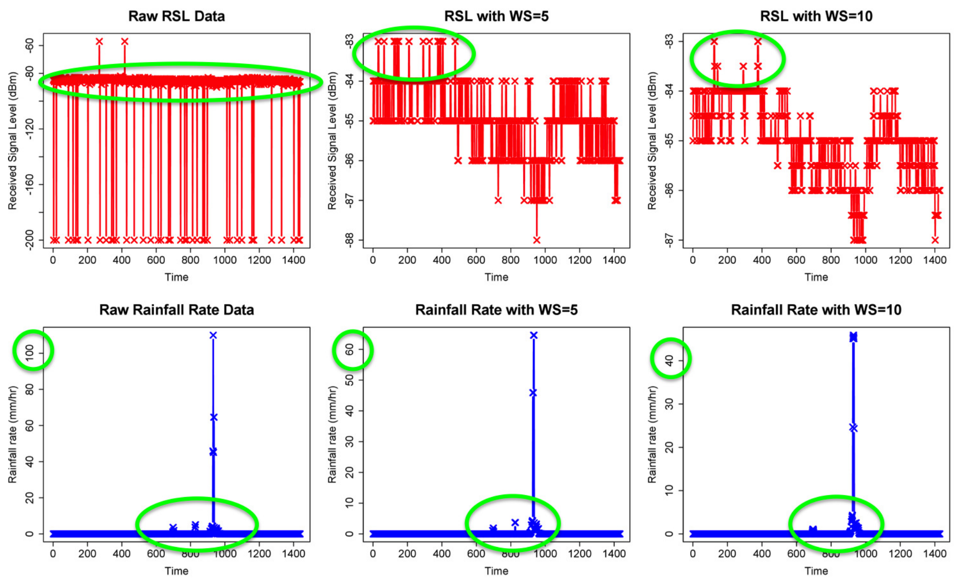


Figure 22: Preliminary processing of raw RSL data

When RSL was plotted with rain, no significant trend and/or threshold was observed even for other rain events with maximum rainfall rates of 50mm/h. For humidity, however, it can be observed that it there is a dense concentration of points at the beginning and at the end of the negative skew of the LOESS (locally weighted scatterplot smoothing) as in the solid curve line (Fig. 24).

We assume then that RSL may be related to humidity, such that, in the next section we discuss the results of cross-correlating humidity and RSL.

Cross-correlation Function (CCF)

With the assumption that RSL is related to humidity, we applied CCF to determine the relationship. First, we examine the entire data in a day (Fig. 25) then investigate further only when there is rain actually happening (see enclosed part).

Note that for the RSL in Fig. 25, the LOESS was calculated before applying the CCF. Even with the processed RSL data, noise is still evident in the resulting graph as in Fig. 27. LOESS, as previously mentioned, is one technique to easily eliminate noise in the data by estimating weights based on the samples and produce values that are almost similar to the original.

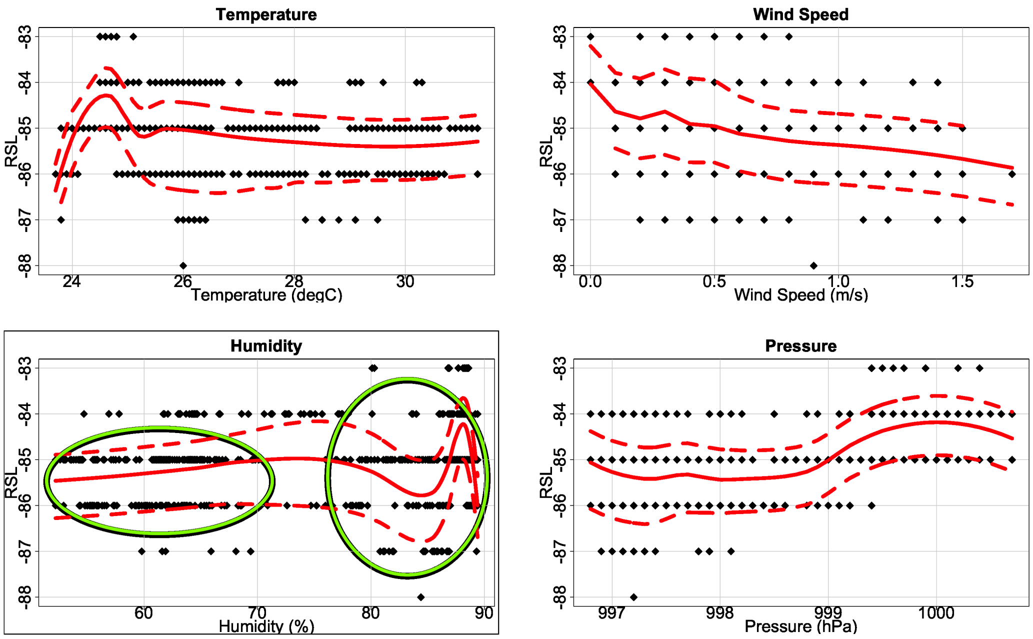


Figure 23: Scatterplots of RSL values with temperature, wind speed, humidity, and pressure.

The CCF is defined by the following equation for continuous data sets:

$$(f \star g)(t) = \int_{-\infty}^{\infty} f^{*}(\mathrm{t})g(\mathrm{t}+\tau)\mathrm{d}\tau \tag{1}$$

where $f^*(t)$ is compared to $g(t)$ for some lag τ, which in this case is the maximum number of samples minus one. Note that, f^* is the notation for the complex conjugate of f. The reason for choosing such lag is to maximize the comparison between two time series to get the highest resolution especially for very short-term analysis of guerilla rains. For a guerilla rain event on Aug 30, 2012, a maximum of -0.6 cross-correlation can be observed. The reason for calculating the CCF is not only to provide evidence for the potential relationship of 5 GHz attenuation signal to humidity, but also to investigate the probability of deriving humidity levels from RSL when there is no sensor available for it in the smartphone.

This section discussed methods of analyzing raw radio signal data, mainly of the 5 GHz WiFi, and the challenges of finding relationships of it with weather

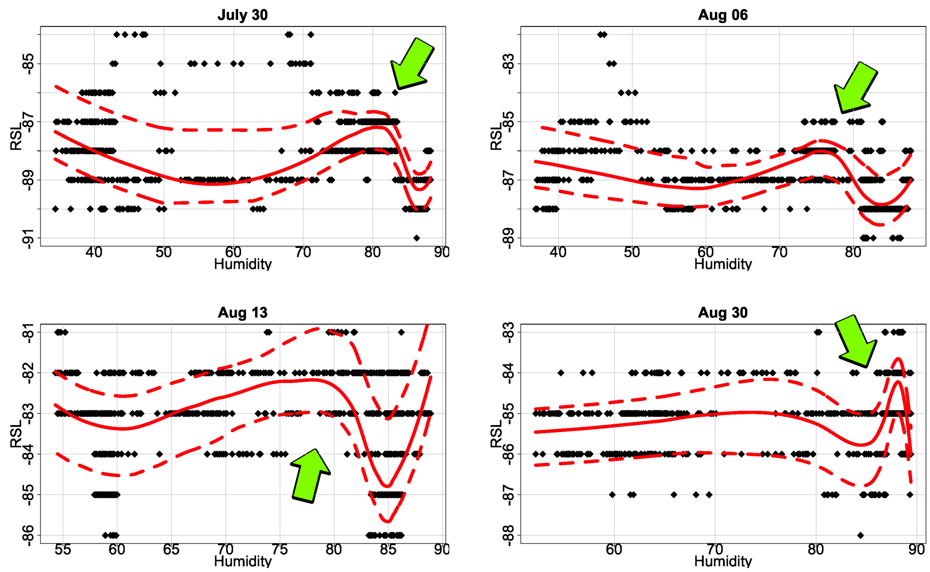


Figure 24: Scatterplot of RSL vs. humidity for summer 2012 having averagely 50mm/hr of rainfall rate. The solid line represents the LOESS while the dotted lines represents the spread of LOESS.

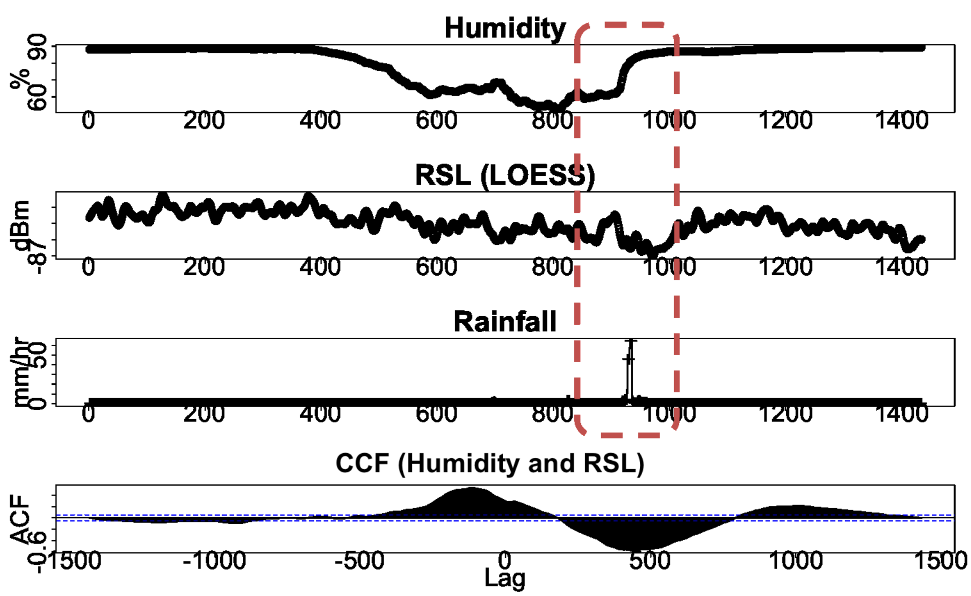


Figure 25: CCF results from an entire day measurement

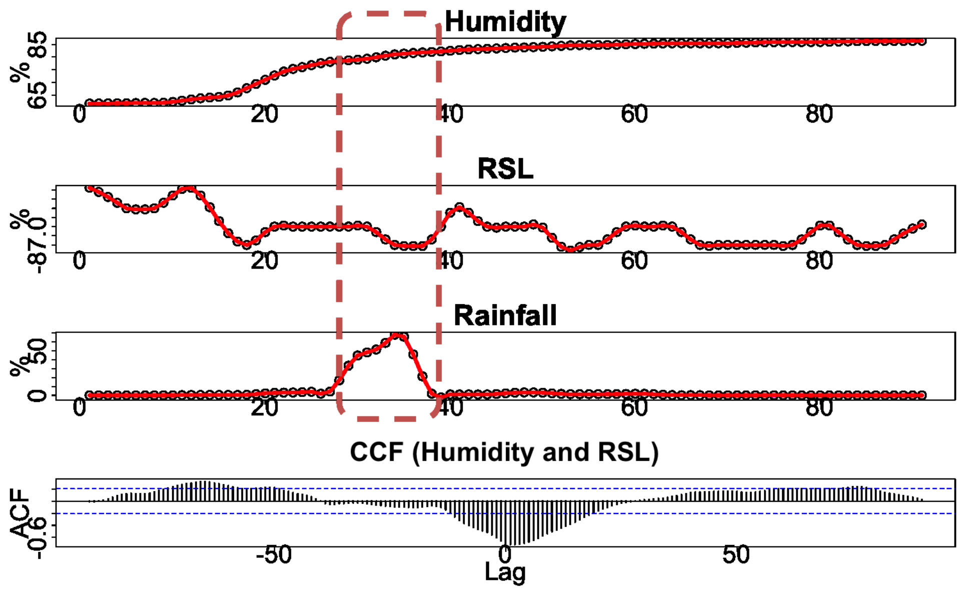


Figure 26: CCF results of the actual rain event duration

parameters. It was not clear whether the underlying results presented potential relations between humidity and RSL. In this current measurement system, therefore, several limitations were observed. Firstly, the location at which the system was deployed did not experience as much strong rains as desired. Therefore, only a few data sets were found significant for processing. Secondly, this measurement system is the only one link being observed. It would have been helpful to be able to observe multiple links, such that a trend can be formalized even for a few guerilla-like rain events. In addition, the system currently relies on a continuous power supply, such that no significant data can be recorded like what happened in the extreme rain event on August 14, 2012 [20]. Lastly, retrieval of the reference Live-E! data was occasionally inconsistent when rain occurs. The problem may be due to the network delay and load as the data is obtained by sending request to the sensor. Also, the sensor has limited sensitivity to precipitation and therefore another reference for on-site weather data is needed.

5.3 Summary

This section discussed methods of analyzing raw radio signal data, particularly of the 3G and WiFi of both 2.4 GHz and 5 GHz, as a preliminary investigation to detect localized heavy rain events for our design of a warning system. We found that there is consistency in the values of humidity for any rain event and it may be a key to understand the behavior of radio signals in describing localized heavy rain events. In addition, EDA results of only the 5 GHz WiFi signal measurements showed significance that is potential for detecting localized heavy rain from its formation to dissipation. It showed consistency in trends more than the findings from 3G and 2.4 GHz radio signal measurements. Furthermore, finding relationships of the measured radio signals with weather-related parameters presented challenges of limited data sets and measurement points.

6. Use smartphone measurements in support of existing commercial sensors

To observe ground weather conditions with increased spatial resolution, we focus on near-surface measurements by consumer handheld sensors and commercial weather stations. In this chapter, we present our conceptual design of a near-surface weather observation.

6.1 Measurement experiments using different sensors

We conducted two kinds of experiments to examine the difference in measurement values using different sensor types. We focused on temperature as the sensor is most common to several of the devices used [4, 35, 37, 40, 41]. In both setups, we used digital and analog handheld sensors as well as weather stations and recorded temperature values every minute.

In the first setup, to see how much difference in measurement values each sensor produce, all devices were simply deployed near each other where a short-term observation was done on a fair day with low-wind condition (Fig. 27).

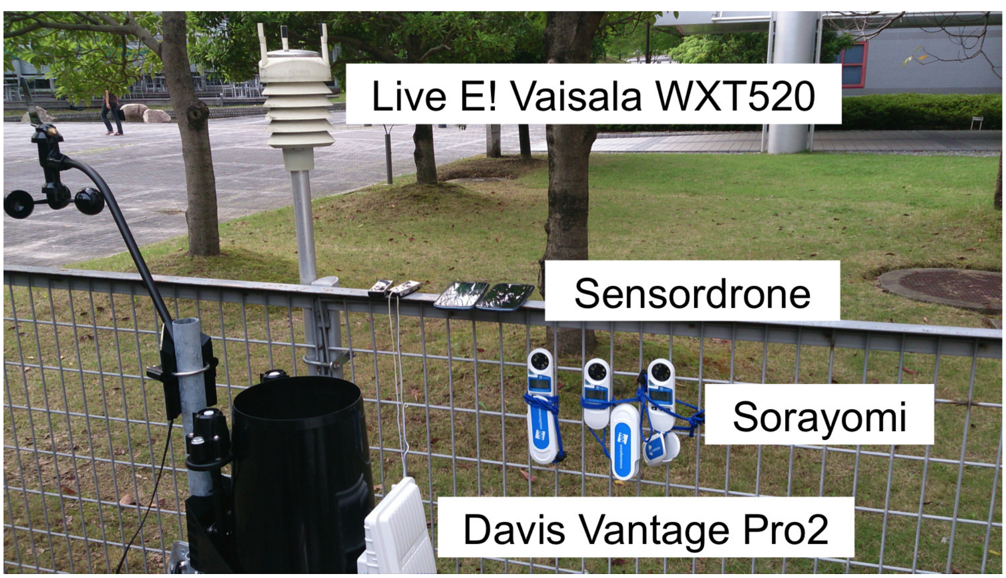


Figure 27: First measurement setup using different sensors

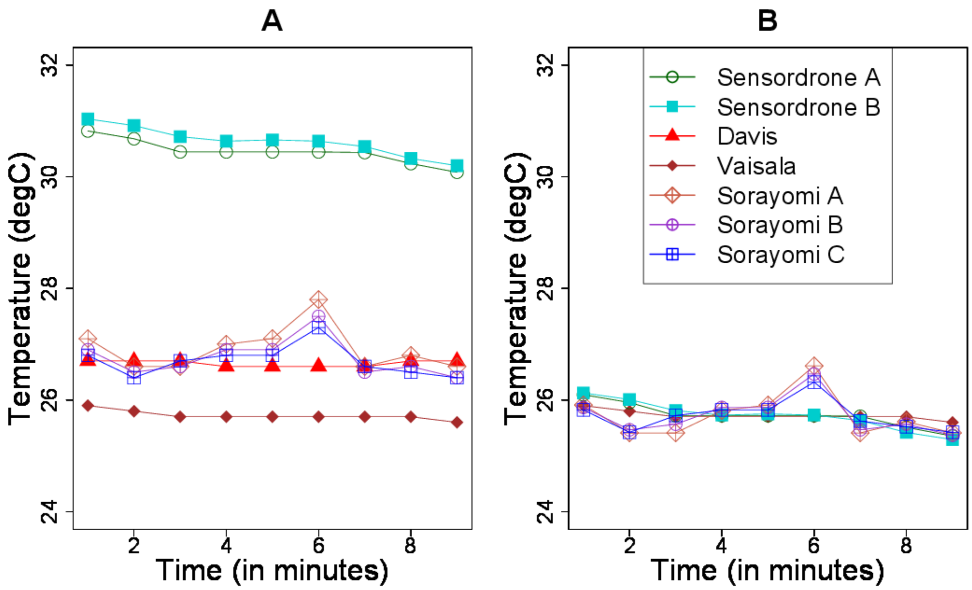


Figure 28: Measurement results of the (A) raw and (B) calibrated temperature values observed by different sensors

According to the raw temperature measurement results in Fig. 28(A), the Sensordrone exhibits temperature values that are averagely higher than the Vaisala by 4.5 °C while the others differ by 1 °C.

We found that the absolute value of each sensor measurement is different but with similar trends. To deal with erroneous values from multiple devices measuring the same event, these were calibrated based on a relatively reliable reference device. Based in the first setup, the values measured by Vaisala was chosen as reference being a fixed standard commercial weather sensor. In this case, calibration was performed by taking the absolute difference of each time-series value from Sorayomi, Sensordrone, and Davis from the corresponding values by Vaisala and solving for the average difference for each sensor. For example, as in Fig. 28(A), the value of 31 °C by Sensordrone A in the first minute is subtracted from 26 °C by Vaisala in the same minute and thus, the absolute value is 5 °C. This will be done for the next 2 to 9 minutes and the calculated average is 4.5 °C, for instance. This value will then be subtracted from the values by Sensordrone A, which produces 26.5 °C as in Fig. 28(B).

In another experiment, ground temperatures were sampled every minute for

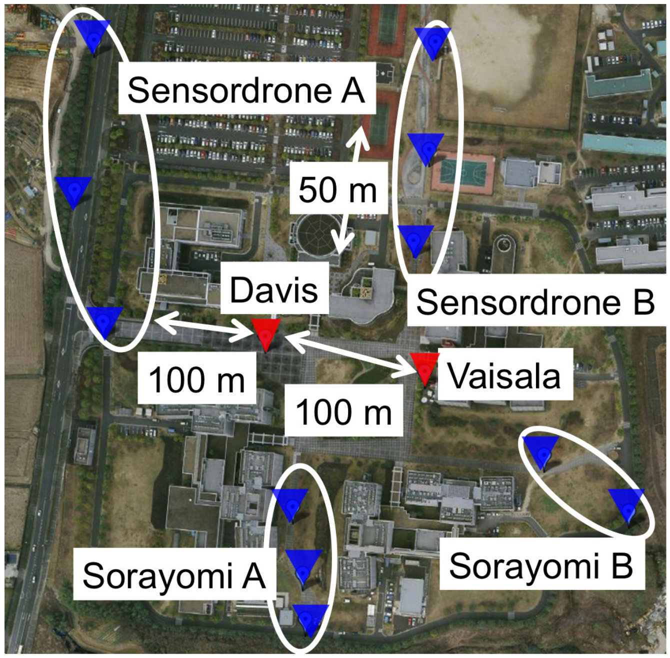


Figure 29: Second measurement setup using different sensors

up to approximately 20 minutes in designated locations as indicated in Fig. 29.

The reason for having this type of setup is to assess the variation in the observed values in terms of distance and time. To find out the difference in measurement values of each sensor, if any, by how far they are located to each other and of the differences in deployed locations, the distances were varied from a minimum of 100 m to 200 m maximum distance depending on the field area limit under observation. Except for the Davis and Vaisala instruments, only the locations of the handheld sensors were varied by 50 m which were recording temperature values for every 20 minutes in each location. The arbitrary minimum sampling time was set to 20 minutes so that the ratio of the variation in measurements may be determined. Moreover, the minimum distance was set to 100 m in the beginning of the measurement to establish it as standard spatial resolution for this study. Finally, the distances were varied by 50 m to determine if the spatial resolution can be extended up to 50 m if significant differences in measurements is observed.

In the second experiment, the same calibration process was applied within the

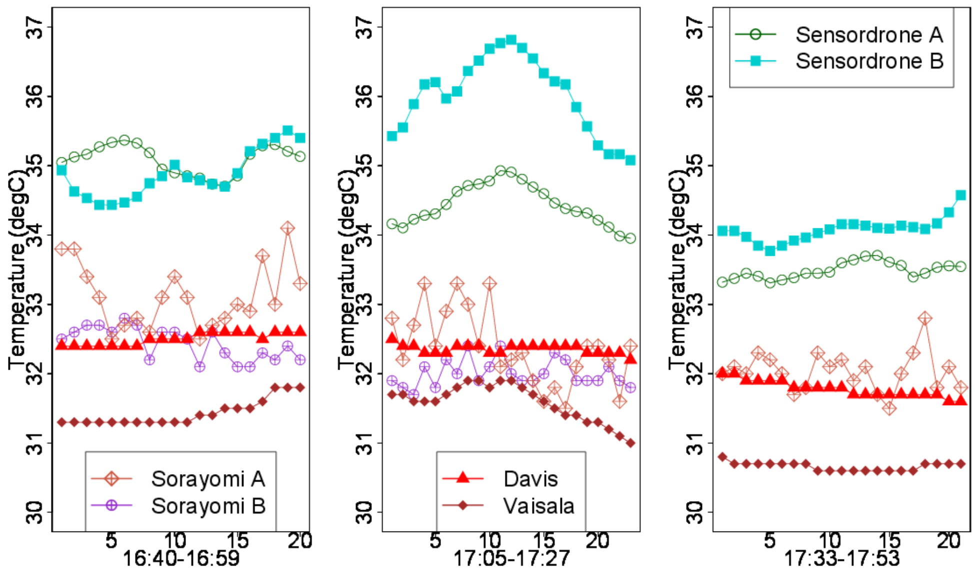


Figure 30: Result of measurement values from all sensors measuring at varying distances at different time durations

similar sets of measured data. The results[6] in Fig. 30 shows that temperature values were significantly lower later in the day.

In this case, however, there needs to be a process of choosing the reference device as the distances were varied. In general, we determine the reference as the device having a fixed location and is only 100 m away, with 100 m considered to be the highest spatial resolution. For example, in the case of the first measurement set of Sensordrone A, the reference device was Davis and all of its measurements were calibrated to it. In the case that a reference device cannot be determined, the particular sensor has to be calibrated with a neighboring sensor that has been calibrated to a reference within the 100 m^2 grid. Generally, therefore, all sensors must be calibrated within the grid area with at least one device calibrated to a reference; or to the nearest device in another grid that has been calibrated to a reference.

[6]Sorayomi B did not perform further measurements during 17:33-17:53 due to a limited area.

6.2 Issues to address based on experimentation

The most obvious issue experienced in the experiments was the large difference in the measured temperature values for a small area. To deal with it as with big data, designating a reference point value within a particular grid area and calibrate other measurements to it by our simple method can be a solution. Following that, however, we may have to evaluate which reference device to select. A straightforward process would be to assign a device in the relative center of a grid, one which is not changing locations as we also deal with mobile sensors.

More importantly, not all weather instruments can have the same quantity of sensors. We propose to utilize the BT device in smartphones to compensate for parameters not being to measure for the lack of it in the device. We can deal with insufficient weather-related information at a certain point location by querying for such information from nearby nodes or sensors. We do this via BT wherein the smartphone broadcasts a pairing request for BT-enabled devices within a maximum of 100 m perimeter and request, for instance, wind speed data at a certain point in time.

Lastly, a larger issue would be the resolution of a reliable representative value and a way to aggregate the data. With enough trending weather information of at most 45 data, we can possibly determine the distribution of the change in weather measurements and directly infer on a representative value at certain grid points. This also includes the RSL measurements that we are proposing to integrate with the required weather information. We can look at the distribution of RSL in among several devices in support of humidity measurements, for instance. To gather all information, we rely on the smartphone to directly send the measured information to our central database.

6.3 Summary

This section discussed a conceptual design of a near-surface observation and measurement system using multiple distributed devices for weather observation. Based on our partial implementation using weather stations and handheld sensors, we found the data quality differs among devices in a very small area of 100 m grid.

The advantage of our system design is the ubiquitous acquisition of near-surface weather observation measurements using smartphones that can either have embedded sensors for weather or have external sensors integrated with it. We contribute to finding new data reliability solutions for multi-device weather sensor integration and big data analytics with our on-going implementations.

7. Calibrating smartphone-based weather measurements via Pairwise Gossip

Similar to how weather instruments are calibrated and maintained, however, smartphone-based data should undergo similar data correction processes to yield reliable synoptic weather data. Therefore, in this chapter, we investigate how to correct or adjust smartphone-based environmental data even when the device is normally used as a smartphone. Using commercial weather instruments and built-in smartphone sensors, we adjust the device-based measurement with a heuristic-based pairwise gossip algorithm. In our general setup, fixed sensors like the commercial weather stations are assumed to produce proper information and therefore, the majority of the adjustments are to be performed on the smartphone-based data. To do so, we consider the basic context-based information such as acceleration to observe user activity and adjust the measured information accordingly.

Quantitative estimation of the current environmental conditions, such as temperature, humidity, and pressure allows general forecasting services to have an outlook of the weather in the next few minutes partly based on ground surface information. If the estimate is far from actual conditions, this could remarkably affect the calibration with other weather instruments and eventually, the forecasting model outputs. Therefore, if we could accordingly adjust the values from the source in reference to a relative ground truth, then we may be able to mitigate forecasting errors from the lowest level of computation. The *formulation of a heuristic-based pairwise gossip algorithm that will adjust pressure values as measured by the embedded sensors in the smartphone based on a normal usage* is a contribution of this work. Adjusting the smartphone-based value when the user is stationary or moving requires the reference measurements of a weather station as our established ground truth. In this way, it does not require complex formulations to easily calibrate the pressure sensor on the smartphone.

7.1 Related Work

"Crowdsourced" environmental data is largely affected most especially by how individuals are using their devices when data is taken. Several surveys like [14], [20], and [1], for instance, reveal that individuals would typically keep the devices inside their shirt or trousers pockets or inside shoulder bags or backpacks. Although these survey results were mainly used for activity recognition, we expect that measurements performed in such instances may offset the ideal measured value and notably affect the accuracy and reliability of environmental analysis. Thus, in [29], it has been emphasized that calibration is important in sensor networks to avoid unreliable measurements. This is typical for environmental sensors which weather forecasts rely on and significant for crowdsourced data affected by several human factors. Furthermore, calibration allows for the identification of errors in the system that may be attributed to *offset faults*, *gain faults*, and *drift faults*. While calibration is often a difficult task, it can be typically implemented in sensor networks before they are deployed or while they are on deployment. An example of calibration that is performed on sensors in-situ is the work on target detection using low-cost sensors as in [39]. The study proposed a calibration algorithm based on feedback control theory and a combination of data fusion and Bayesian detection models to properly identify a target exposed under the sensors. Results from small-scale testbed and simulation using real vehicle detection data has proven that target detection using their algorithm achieved optimal performance. Meanwhile, another approach to calibrating sensor networks is based on a gossip protocol [32]. The goal was to estimate a *signal signature* based on the collective sensor node values while calibrating the values at the same time. Distributed processing techniques were applied to uncalibrated sensors in the network to correct them and determine the signal pattern. Based on their system model and the gossip-based distributed algorithm, the Distributed Signature Learning and Node Calibration (D-SLANC) algorithm was derived. This algorithm enables local calibration among sensor nodes and addresses the global estimation problem.

Similarly, we would like to address faults with the embedded smartphone sensors with high-end commercial weather stations as our relative ground truth. In this way, we may be able to utilize the smartphones as a network of weather instruments that can provide general forecasting services with sufficient synoptic ground weather information for their forecasts. Thus, using analytical techniques of context recognition like in [9] and [27] and gossip-based concepts, we would like to investigate the effect of placing the device inside a shoulder bag with some user activity in our aim to correct embedded smartphone sensor measurement at the device level.

7.2 Investigation of smartphone sensor data as affected by user activity

Based on the survey results of mobile phone usage provided in [20], 35% of the respondents put their devices inside a bag. This is greater than both the 30% of respondents who put it inside their trouser pocket and 13% inside the chest pocket. Therefore, considering these statistics, we chose to observe first the effect of placing the device inside a bag using two experiments. Using several models of Samsung smartphones listed in Table 1, we investigate on the pressure data having the sensor common to all devices in the list, while the device is measuring from inside a shoulder bag. Surface pressure is an indication of the changes in atmospheric forces that is helpful for meteorologists to predict what kind of weather we will be experiencing. For now, we simply focus on the pressure readings for the weather measurement since the environmental sensor is most common to some smartphone models which are currently being manufactured. Each device is then installed with an Android application that we developed, which logs the available sensor data for every second. The application, in general, samples the instantaneous measurement of the embedded smartphone sensors at every second then continuously logs these values as a CSV file in the internal device storage.

We used two units of the Samsung S3 models and one unit each of the Samsung Galaxy Nexus and S4 models, subjecting four devices overall in both experimental setups. Both experimental setups used the same shoulder bag by the same user to perform the measurements. Motion sensors, such as the gyroscope, accelerometer, and magnetometer were observed in three dimensions subjective to the device orientation. Meanwhile, environmental sensors like light, proximity, pressure, temperature, and humidity were recorded as is. In both experiments, we refer our readings to high-end commercial weather stations, such as the Vaisala WXT520 [40] to provide us with measurements for our estimation reference. It is important to note that, we are using several devices, each of which has a particular margin of error. As for the Vaisala, it has an acceptable error of ±0.5 hPa considering that it has been calibrated according to standard. The pressure sensor in the smartphones has a maximum absolute error of 4 hPa by the specifications according to [25]. Considering these errors, we can not directly compare the accuracy of smartphone devices to the Vaisala since each instrument was developed for different purposes. However, we find that the pressure sensors in the smartphones can be proven useful if calibrated accordingly and if there is potentially enough data. By sufficient data, this could mean having at least one available smartphone device in a 100 meter unit area.

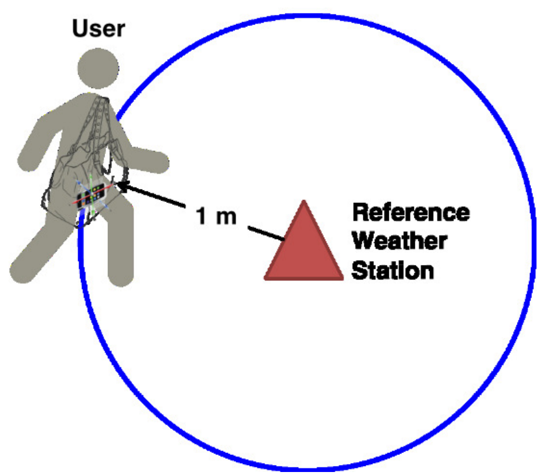


Figure 31: Setup of the stationary user experiment

The first experiment was conducted to investigate the precision of the barometric readings by smartphones compared with that of the weather station. A bigger picture of this scenario is when a user is idling nearby a fixed weather station and with the device measuring on the background. In the setup shown in Fig. 31, the user was required to stand one meter from the reference weather station while carrying the shoulder bag with the devices inside it. A meter away from the weather station minimizes the influence on the instrument. The measurements were performed for three separate afternoons while sampling sensor data for 10 minutes in each event. Studies on context recognition would typically sample for one to a few minutes to get enough data set. For similar experiments with ours, the duration may vary depending on the desired sample size. In our case, we decided on 10 minutes to get enough samples of both environmental and motion sensors. Also, the reference weather station measures every minute and 10 samples is sufficient to describe the surface pressure. Before determining how precise the smartphone-based pressure readings with that of Vaisala, we first pre-processed the data. As the observations were logged in seconds, we wanted to match the per minute resolution of the weather stations. To do so, we sampled an overlapping window on the same minute (60 units) and determined the median. We use these values of median per minute and implemented them in the following uncertainty range equation α as in Eq. 2:

$$\alpha = (\overline{X(t)} - x_{min}(t)) + (x_{max}(t)) - \overline{X(t)}) \tag{2}$$

where $\overline{X(t)}$ is equal to $\sum_{i=1}^{n} x_i(t) + x_{ref}(t)$ divided by N for $n = 4$ smartphone devices used and $N = n + 1 = 5$, which includes the reference weather station having a measurement value of $x_{ref}(t)$. Put simply, it is the the average of the barometric pressure values of both smartphone devices and weather station at time t. Then, we determined $x_{min}(t)$ and $x_{max}(t)$ by comparing pressure readings from among the 4 devices while excluding the weather station since it is a reference. After comparing the smartphone-based pressure readings, we determine the highest pressure value as $x_{max}(t)$ and the lowest as $x_{min}(t)$. In general, determining α can give us a quick and general idea on how much the pressure readings in the smartphone differ with the Vaisala WXT520. Moreover, it is also helpful in knowing how close are the pressure readings among different device models. A sample calculation result can be found at Table 6 based on the sample data in Table 7 where the average uncertainty of smartphone-based sensors for 10 minutes of observed barometric pressure was 2.13 hPa. Therefore, in our actual measurements of a stationary user with the devices in the shoulder bag, we can express that the pressure may be approximately ± 2 hPa precise with Vaisala in reference to the sample calculations.

The second experiment, as in Fig. 32, was performed to observed the effects of user motion on the pressure readings on the smartphone.

Table 6: Uncertainty calculation results based on the Vaisala WXT520

Time	$\overline{X(t)}$	$x_{min}(t)$	$x_{max}(t)$	Uncertainty
14:00	990.47	989.20	991.28	2.0800
14:01	990.44	989.15	991.29	2.1400
14:02	990.48	989.19	991.31	2.1200
14:03	990.47	989.17	991.34	2.1670
14:04	990.41	989.14	991.29	2.1500
14:05	990.41	989.15	991.27	2.1200
14:06	990.34	989.12	991.24	2.1200
14:07	990.35	989.14	991.24	2.1000
14:08	990.35	989.12	991.31	2.1900
14:09	990.34	989.14	991.24	2.1000

Table 7: Sample pressure data for uncertainty calculation

Time	Vaisala	$S3(1)_{\widetilde{x}}$	$S3(2)_{\widetilde{x}}$	$GalaxyNexus_{\widetilde{x}}$	$S4_{\widetilde{x}}$
14:00	991.10	991.28	991.14	989.61	989.20
14:01	991.10	991.29	991.13	989.55	989.15
14:02	991.10	991.31	991.20	989.62	989.19
14:03	991.10	991.34	991.13	989.59	989.17
14:04	991.00	991.29	991.10	989.53	989.14
14:05	991.10	991.27	990.98	989.54	989.15
14:06	991.00	991.24	990.84	989.49	989.12
14:07	991.00	991.24	990.79	989.56	989.14
14:08	991.00	991.31	990.78	989.54	989.12
14:09	991.00	991.24	990.79	989.51	989.14

As a basic scenario for our proposed system, we imagine a user passing by a reference sensor, which is the kind of user motion that we would like to investigate with this experiment. With the same setup, the user at this time was asked to move around the weather station by walking in a leisurely manner. Each set was composed of 10 rounds, to obtain sufficient sample, that was about 7 minutes long while pausing for 2 minutes in between sets. The movement pattern was

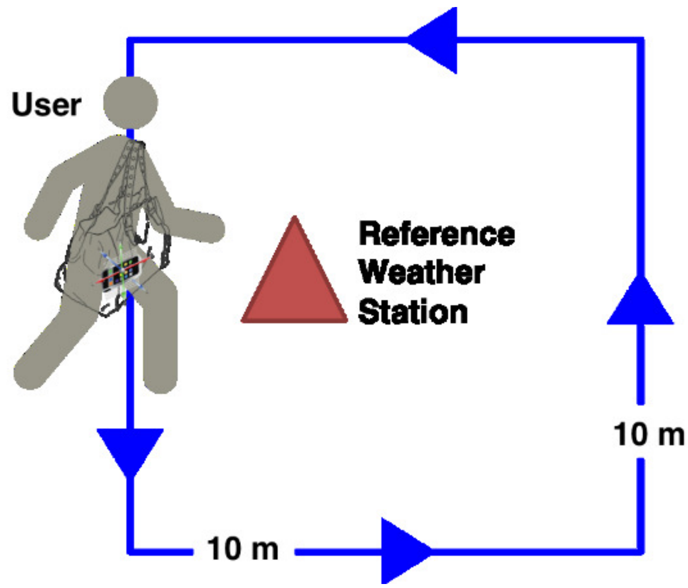


Figure 32: Setup of the moving user experiment

designed to estimate the duration of the rounds so we can replicate the same duration, which was 7 minutes for each round. The pause was done to compare moving and stationary events and serve as a marker between sets. Figure 33 illustrates the raw readings of pressure and accelerometer data taken from the Samsung S4 device as an example. In the graph, the stages of walking and pauses can be easily distinguished by the instability and stability of the accelerometer readings, respectively.

To closely examine the difference between pressure measurements during the presence or absence of movement, we first divided the raw pressure data into partitions of the corresponding stable and unstable measurements of acceleration. This division is shown in Fig. 33, where we have three sets of user movement which correspond to an unstable acceleration and three sets in which the user is not moving which correspond to a stable acceleration. Then, we calculated the variance for each partition and the results are shown in Table 8.

Examining the variance of pressure measured between movement and inactivity can indicate the ability of the sensor to stabilize its readings even when subjected to physical disturbance. Weather stations generally follow a standard

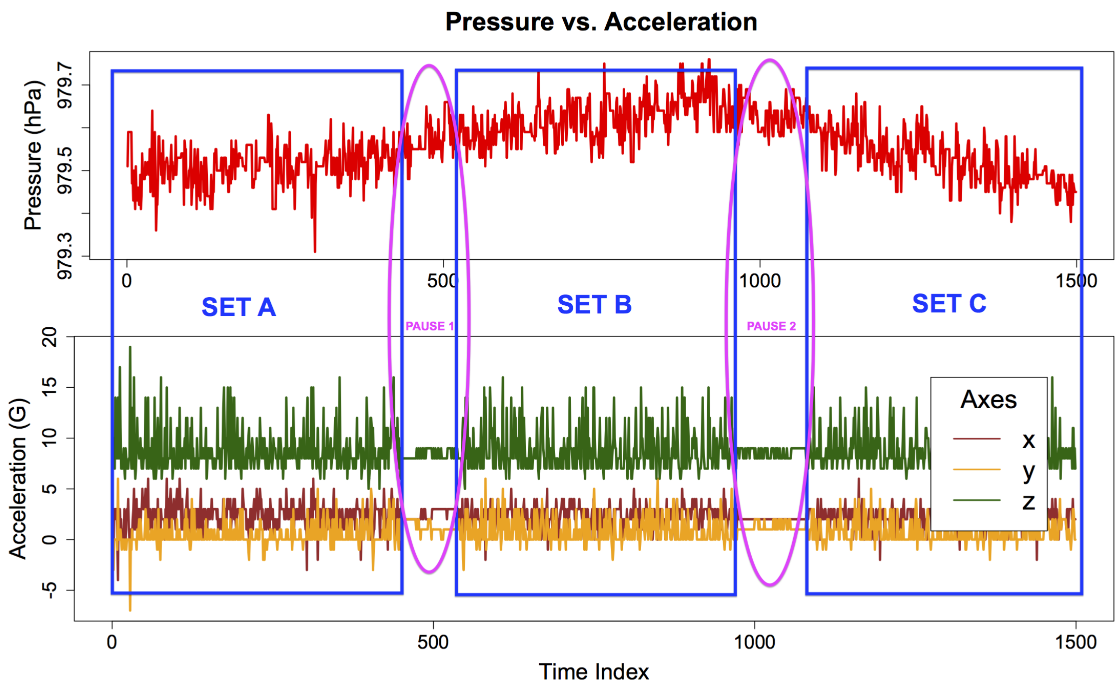


Figure 33: Division of moving and stationary partitions each at 7 and 2 minutes, respectively

Table 8: Sample result of calculated variance between partitions of moving and stationary user

Partition Name	Variance
Set A	0.0021
Pause 1	0.0018
Set B	0.0025
Pause 2	0.0013
Set C	0.0032
Pause 3	0.0017

for fixed setups to provide accurate and precise readings uniformly without having to consider the effect of movement. However, as we are dealing with portable sensors, this is one aspect that we need to further consider for producing reliable measurements similar to that of fixed weather stations. We hypothesize then, that the variance of the sensor is higher during movement than when the user is inactive, and thus, we can presume that the sensor is unstable and stable, respectively. To verify this, an upper one-tailed F-test was performed between phases of walking and inactivity as shown in the results of Table 9. The results show that the F value for all comparisons is greater than the $F_{critical}$ values, which rejects the null hypothesis that the variances are equal and proves that the variance of pressure values during movement is higher than when the user is inactive.

Table 9: F-test result for a sample observation

Partition	F	$F_{critical}$
Set A vs. Pause 1	1.18	0.92
Set B vs. Pause 2	1.91	1.49
Set C vs. Pause 3	1.82	1.45

Overall, we found that the smartphone pressure sensor reading has an uncertainty value of ± 2 when compared with Vaisala WXT520 from our first experiment. Moreover, we verified via an upper one-tailed F-test that the variance of the smartphone-based pressure readings is higher during user movement than when the user is stationary. Although the pressure readings were not explicitly proven to be accurate in the experiments, this would still imply that the embedded sensor is more stable in providing pressure readings if the user handling the device is stationary as opposed to when the user is moving.

7.3 Smartphone-based sensor calibration via pairwise gossip

Based on our findings on the effect of motion on smartphone sensor readings of pressure, we present our heuristic-based pairwise gossip algorithm. To calibrate embedded smartphone sensors with respect to a fixed weather station, our algorithm relies on the variance of the pressure readings. It is also based on the actual difference of the pressure readings between the smartphone-based sensor and the fixed weather station. Gossip algorithms [5] are generally used for the classic estimation of values in a network by distributed averaging. This particular algorithm has its advantages for distributed averaging in sensor networks as it enables quick and efficient analysis of distributed data over sensor networks especially when faced with several constraints as emphasized in [2]. Such constraints include the lack of centralization, dynamically changing network topology, and sensor hardware limitations. The standard gossip algorithm is in the following form:

$$x(t+1) = W(t)x(t) \tag{3}$$

where $W(t)$ is random weight matrix and $x(t)$ is the current value of a node in a network. Ideally for gossip algorithms, the weights must converge to a value of 1. In actual pairwise gossiping as stated in [5], random pairs of neighboring nodes exchange their information and calculate the average of their values as some time t and updates their values with the average, thereafter. Note that, we apply the same principle of pairwise gossiping by maintaining a one-to-one pairing with the weather station to calibrate the embedded smartphone sensors. However, instead of a random weight assignment, we calculate the weights that equate to a unit value based on the variance of the smartphone-based measurement and the actual difference of the measurements between the smartphone-based sensors and reference weather station.

Let us first consider the following sensing model equation [5] as in Eq. (4:)

$$z_i(t) = H_i\theta + w_i(t) \tag{4}$$

where θ is the value that we want to estimate with our actual pressure readings in the smartphone. In our case, we assign it as our reference value which is the weather station measurement. Meanwhile, H_i and $w_i(t)$ are the gain and offset of the system in place, respectively. Ideally, $H_i = 1$ and $w_i(t) = 0$ are true if, for instance, the embedded pressure sensor in the smartphones behaves similar to Vaisala. However, in reality, we have the effects of the surrounding environment, user activity, sensor limitations, etc.

To explain this concept further, we formulate the following heuristics-based pairwise gossip algorithm to adjust and update the pressure readings in the smartphone as in Eq. 5:

$$x_i(t+1) = W_\alpha \theta_i(t) + W_\beta x_i(t) \tag{5}$$

where $x_i(t+1)$ is our updated pressure reading ($z_i(t)$ or y). Meanwhile, $W_\alpha \theta_{i(t)}$ ($H_i\theta$ or mx) is our reference value θ for some ratio of W_α. Finally, $W_\beta x_i(t)$ ($w_i(t)$ or b) is some ratio of W_β based on the variance $\varrho^2 = (x_i(t) - \overline{x_i(t)})^2$ and actual difference $d(t) = |\theta(t) - x_i(t)|$. The current heuristics algorithm is applicable to a one-to-one calibration of smartphone-based data with a reference weather station. To use the algorithm, the scenario requires that the smartphone is measuring within coverage area of the weather station and consequently, located adjacent to the weather station. Thus, Eq. 5 presently does not take distance into consideration in the calculation. Furthermore, it follows that $W_\alpha + W_\beta = 1$ considering that the weights ideally converge to one. And as for our heuristics-based pairwise algorithm, since we only need to compare two values every time, we simply assigned $W_\alpha = 1 - W_\beta$ where W_β is the ratio of $\frac{\varrho^2}{|\theta(t) - x_i(t)|}$. To further understand the process of obtaining W_α and W_β, we will use a data set of raw barometric pressure logged by all devices used from one of our measured events as in Table 10.

Table 10: Sample raw data from a 10-minute event

Time	Samsung S3(1)	Samsung S3(2)	Galaxy Nexus	Samsung S4
14:00:00	991.39	991.11	989.57	989.16
14:00:01	991.35	991.14	989.60	989.16
14:00:02	991.33	991.16	989.58	989.16
⋮	⋮	⋮	⋮	⋮
14:09:59	991.23	990.90	989.58	989.14

Referring to $S3(1)_{raw}$ as a more specific example for our calculation process, we first determine the median per minute of the raw pressure readings $S3(1)_{raw}$. This will produce $S3(1)_{\widetilde{x}}$ values that are in the similar temporal resolution as the weather station measurements of pressure. Refer to Table 7 for a sample result of these median values per minute for each device. Then, we calculate the variance of the pressure readings of the raw data of each device per minute or $Var_{S3(1)_{raw}}(t)$ for example. Next, we obtain the absolute difference of pressure readings between the calculated median per minute, $AbsDiff_{Vaisala-S3(1)_{\tilde{x}}}(t)$ for instance, at each device and of the weather station measurements. Refer to Table 11 for the results of Samsung S3 as an example of these calculations.

Table 11: S3(1) calculation result of Variance and Absolute Difference

Time	Vaisala	$S3(1)_{\tilde{x}}$	$Var(t)$	$AbsDiff(t)$
14:00	991.10	991.28	0.0049859	0.17978
14:01	991.10	991.29	0.0037714	0.19140
14:02	991.10	991.31	0.0037007	0.20952
14:03	991.10	991.34	0.0090520	0.23600
14:04	991.00	991.29	0.0046557	0.29380
14:05	991.10	991.27	0.0047261	0.16914
14:06	991.00	991.24	0.0055904	0.23865
14:07	991.00	991.24	0.0054008	0.23590
14:08	991.00	991.31	0.0056691	0.31200
14:09	991.00	991.24	0.0041641	0.24316

Then, we consider the effects of the user motion via the variance of the smartphone-based readings and the actual difference of the readings between the smartphone-based sensor and the weather station. We do this by calculating the ratio between $\varrho^2_{(S3(1)_{raw}}(t)$ and $|Vaisala - S3(1)_{median}|$, which we refer to as our W_β. Then, we obtain W_α by $1 - W_\beta$ considering the prior weight condition that requires the weights equal to one. Finally, using these calculated weights, we can update the value of $x_i(t+1)$ as in Table 12.

The resulting adjustments have significantly transformed the measurements and those measurements are now very close to the reference values as shown in the comparison graph in Fig. 34. Each device model essentially has different values of W_α and W_β as reflected in some sample values in Table 13 and Table 14, respectively.

Table 12: S3(1) calculation result of $W(t)$ and $x_i(t+1)$

Time	Vaisala	S3(1)	W_α	$W_\beta = 1 - W_\alpha$	$x_i(t+1)$
14:00	991.10	991.28	0.027733	0.97227	991.10
14:01	991.10	991.29	0.019704	0.98030	991.10
14:02	991.10	991.31	0.017663	0.98234	991.10
14:03	991.10	991.34	0.038356	0.96164	991.11
14:04	991.00	991.29	0.015847	0.98415	991.00
14:05	991.10	991.27	0.027943	0.97206	991.10
14:06	991.00	991.24	0.023425	0.97657	991.01
14:07	991.00	991.24	0.022894	0.97711	991.01
14:08	991.00	991.31	0.018170	0.98183	991.01
14:09	991.00	991.24	0.017125	0.98287	991.00

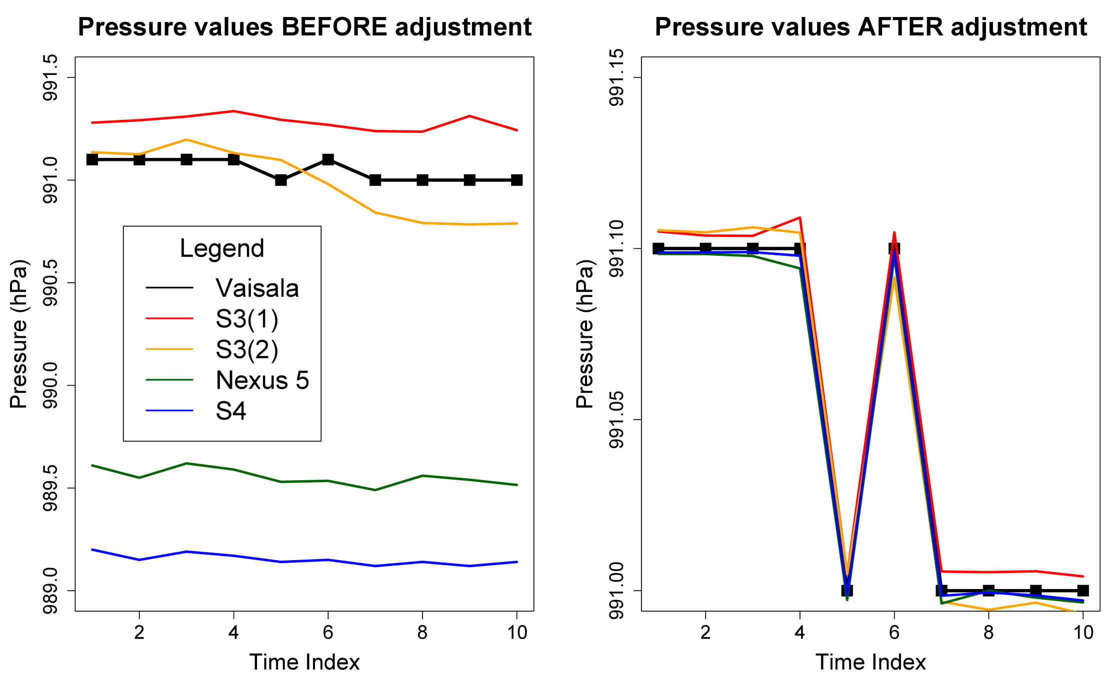


Figure 34: Comparison of pressure per minute before and after adjustments

Table 13: Calculated W_α of different device models

Time	Samsung S3(1)	Samsung S3(2)	Galaxy Nexus	Samsung S4
14:00	0.97227	0.84940	0.99894	0.99937
14:01	0.98030	0.81662	0.99896	0.99942
14:02	0.98234	0.93642	0.99852	0.99945
14:03	0.96164	0.85621	0.99615	0.99890
14:04	0.98415	0.95145	0.99813	0.99932
14:05	0.97206	0.92999	0.99847	0.99957
14:06	0.97657	0.97966	0.99752	0.99920
14:07	0.97711	0.97323	0.99998	0.99971
14:08	0.98183	0.98382	0.99859	0.99926
14:09	0.98287	0.96791	0.99768	0.99845

Table 14: Calculated W_β of different device models

Time	Samsung S3(1)	Samsung S3(2)	Galaxy Nexus	Samsung S4
14:00	0.027733	0.15060	0.0010565	0.00062344
14:01	0.019704	0.18338	0.0010387	0.00057949
14:02	0.017663	0.063578	0.0014766	0.00055123
14:03	0.038356	0.14379	0.0038511	0.0010974
14:04	0.015847	0.048554	0.0018666	0.00067643
14:05	0.027943	0.070009	0.0015258	0.00043013
14:06	0.023425	0.020340	0.0024820	0.00079598
14:07	0.022894	0.026771	0.000017361	0.00029228
14:08	0.018170	0.016183	0.0014091	0.00074171
14:09	0.017125	0.032091	0.0023210	0.0015459

In summary, we formulated a heuristic-based pairwise gossip algorithm that adjusts the smartphone measurement values with respect to the weather station measurement. Prior to this, we verified that the variance is higher for instances when the user is moving as opposed to when it is stationary. The difference in variance can be linked to the stability and instability of the embedded smartphone sensors. Therefore, to employ this finding, we calculated the weights in accordance with the ratio of the variance of the raw pressure data and the actual difference between the median pressure data and the reference weather station values. These weight calculations apply to calibrating embedded smartphone sensors with fixed weather stations as an established ground truth. Moreover, the weights W_α and W_β is not constant over time. In real measurements, therefore, we can calibrate smartphone-based measurements based on the weights even when the user is moving. For instance, in a setup where the user is located within the coverage of a fixed weather station, the established ground truth measurements would most likely have a larger percentage in the calibration. If the ground truth measurements are presumed to be accurate, these values can be representative estimates of the synoptic ground weather condition. Thus, the percentage of the supporting weather information from the smartphone sensor data is dependent on the weights whereby the effect of movement is mitigated via the variance and absolute difference. As a result of a one-to-one fixed setup, the smartphone sensors would be updated with values closer to the representative estimate.

7.4 Summary

This section disccussed a heuristic-based pairwise gossip algorithm to adjust embedded smartphone pressure sensor measurements. Based on our experiments with the smartphone pressure sensors, we found that the pressure sensors of the different Samsung smartphone models we used have a certain precision value compared with Vaisala WXT520 which we established as our referential ground truth. Moreover, the pressure readings were verified to be unstable when the user is moving compared when the user is stationary. Thus, to adjust accordingly, we consider the effect of user activity while the device is measuring from inside a shoulder bag by integrating the variance of the raw pressure readings with respect to the actual difference from the reference weather station as our weight ratio. These weight ratio are then consolidated with the pairwise gossip algorithm which updates the pressure reading of the embedded smartphone sensor.

By adjusting the sensor measurements accordingly, we can provide an almost accurate and precise synoptic weather information to general forecasting services. Moreover, as this information can possibly be densely available due to the popular use of smartphones, general weather forecasting services can mitigate errors at the sensor level with this particular calibration method. Thus, this paper contributes a straightforward and heuristic linear estimation using the principles of pairwise gossip. A limitation of this method, however, is that the smartphone requires to be located nearby a weather station at present.

8. Conclusion

This dissertation discusses on the use of mobile devices as solutions for mitigating risks before and after disaster events. When devastating disasters occur, available preparation time is typically insufficient. Especially when individuals are not informed on time, they are most likely to be vulnerable to casualties. Moreover, after the disaster has occurred, there is hardly any communication resource with damaged infrastructures. To overcome these challenges, therefore, applications such as the SOSCast was developed and experimentation with smartphones and commercial weather sensors were performed. Considering the above goals, this study provides two main contributions that are listed as follows:

- Introduce a way to communicate between smartphones even without the conventional Internet-based communication particularly after a disaster has occurred

- Introduce a novel way of using smartphones as a weather sensor in addition to its communication capabilities to inform individuals of the incoming risk before a disaster occurs

With new technologies emerging in ten years or so, this research will evolve with having the developed smartphone application in this study, be seamlessly used in cases of emergency during disasters. Instead of having to have it pre-installed when smartphone users would simply ignore or uninstall it, the application would be made available only when it is time for use. With SOSCast, for example, it can be made available for installation at the same time when an early warning is issued during a large-scale earthquake. Also, during the time when small satellites are prevalent, SOSCast can be broadcast via these satellites and immediately be used by victims even when communication channels on ground are unavailable. As for detecting complex weather phenomena such as heavy rain events before they occur or affect a certain area, pervasive sensing with the popular use of smartphones will contribute much to providing actual in-situ data. Big data is now becoming key to understanding complex phenomenon especially for weather forecasting. In the next decade within the phase of the Internet of Things (IoT), in-situ observation of weather conditions can now be possible with newly developed smartphone technologies and devices. These new devices can pervasively sense weather information as provided by individuals, contributing to a more accurate detection and information sharing of devastating weather phenomena. As these are yet far from reality, this research on mobile-based applications will continue to explore new solutions for disaster risk mitigation.

The following discussion presents a quick summary of each chapter focusing on issues that needed to be resolved in this study.

8.1 Summary and Discussion

Chapter 2

Extreme weather phenomenon such as heavy precipitation and earthquakes are occurring more frequently than the previous years. Recent reports claim that these circumstances are attributed to climate change. In that regard, more devastating events are about to happen and the most vulnerable are individuals who are regarded as "disadvantaged", i.e., in developing or least-developed countries. Several actions have been done to mitigate the risks brought by the effects of the rising global temperatures. However, more efforts are yet to be done in managing emergency situations, particularly in terms of rescue operations and preparation.

Communication is most important during post-disaster situations especially for rescue operations. For most of the cases, however, communication channels are also damaged during disaster and can become unmanageable when rescuing people for at least within 72 hours. This is the most critical time duration when victims can survive the longest especially when trapped under damaged infrastructures and buildings. If communications from victims to rescuers is possible, then it would be easier to locate the victims despite being trapped underneath the rubbles.

One of the ways to be prepared is to be informed of the approaching risk, that is, by observation and forecast. General forecasting services typically make use of the combination of satellite image, radar, and in-situ weather stations to make forecasts. As the equipments to acquire these information may be costly, spatial coverage may also be compromised. Thus, some areas may not have available weather information which makes people in these areas unaware of the impending danger.

These are only a few of the reasons why the focus is on emergency communications and in-situ weather measurements and observations. More importantly, as individuals these days typically carry with them their mobile devices, it is advantageous to take this matter to provide solutions to a select few challenges with emergency management.

Chapters 3 and 4

Chapter 3 discusses one specific example on how emergency communications can be implemented using smartphones in a scenario of a devastating earthquake. In this case, we assume that communication lines are cutoff as it has been damaged along with infrastructures and buildings. The next best thing that victims who are trapped inside damaged buildings and infrastructures have is their mobile devices, or their smartphones. The smartphone application, namely the SOSCast, has been developed to enable direct communications between smartphones even without using the Internet. Normally, the way to exchange data is by using the Internet as a channel to send and receive messages to and from recipients far and wide. However, as the Internet is unavailable due to destroyed communication towers, it has become impossible to do so. On the other hand, smartphones have built in communication devices (WiFi and BT) that, in fact, enables these devices to communicate directly. With that said, SOSCast was developed over BT to transmit and receive SOS messages to another device that can use BT as well.

During the development of the SOSCast application, several issues like message duplication, speedy battery depletion, and limited communication coverage. In that regard, improvements have been accordingly done to the SOSCast application to resolve these problems as described in Chapter 4.

Generally, there may be cases when the immobilized person is unable to obtain GPS information as the debris may prevent GPS connection with the smartphone. When the GPS information of the immobilized victim is unavailable, the rescuers can simply refer to the GPS information of the propagator or the mobile victim to whom the immobilized victim had prior connection with. Note that, whenever the propagator or mobile victim establishes BT connection with an immobilized victim, they also log the GPS information from where they had connection. In this way, the rescuers can search for the trapped victim in the nearby perimeter based on the mobile victim's acquired GPS location.

Moreover, at the time of writing, new communication technologies like the iBeacon [21, 7, 3] are emerging and may have to replace general BT for DTN applications. iBeacon was proposed by Apple, which can function on both Android and IOS, implementing BT Low Energy (BLE) wireless technology that makes it efficient with power consumption. Using iBeacon makes it easy for both indoor and outdoor location identification with the correct settings. Compared with the BT pairing process, iBeacon needs a battery-powered iBeacon device and UIID, Major and Minor value settings with the smartphone device to identify with the iBeacon device. It is, thus, appropriate that SOSCast may simply have to adapt iBeacon when establishing connections between immobilized and mobile victims without the inconvenience of approving BT pairing requests. In this regard, however, the message collection process with representative nodes may have to remain the same as in the case of the improved SOSCast.

Chapters 5, 6, and 7

Chapter 5 discusses the investigation of a point in-situ weather observation and measurement system for pre-disaster risk preparation. In consideration of limited resources especially in developing or least-developed countries, the focus of the investigation revolves around finding devices with least cost and can be easily implemented in a system. With that said, received signal level (RSL) was sought as a conventional and additional information to describe the onset or occurrence of localized heavy rains. Moreover, environmental data from current models of smartphone devices with built-in sensors were integrated into the system to analyze the correlation and discover an alternative way of obtaining pinpoint weather data.

However, having only RSL data is insufficient to actually describe current weather conditions. Thus, as in Chapter 6, other mobile sensors were introduced and measurement experiments were performed to determine how these sensors can be useful. In this sense, we explore other possibilities of also using existing handheld sensors or built-in sensors in smartphones as additional information to

RSL in describing in-situ point weather.

To simply make sure that this integration is reliable, calibration experiments by measurement were also performed as described in Chapter 7. In the simplest setup, pairwise gossip was implemented to calibrate measurements taken from different mobile devices and sensors. Note that such calibration is important because small errors at the lowest level of measurements can greatly affect the final output, i.e., forecasts.

The general approach of using RSL and other sensors to detect localized events such as heavy rains in this research can be applicable to other natural hazards with similar characteristics. As aforementioned, RSL is mainly affected by the amount of water vapor in the atmosphere. That is, increased humidity can be assumed when increased attenuation is observed when measuring along the link. In that regard, severe thunderstorms or squalls can be other hazards that can be detected through RSL-based approach. For other natural hazards that are the by product of thunderstorms like tornadoes, we may not be able to directly identify such with our approach. As it can have similar onset characteristics with thunderstorms, the mechanism of its formation can be different, which is primarily attributed to wind activities. Nevertheless, it can be a potential focus in future works similarly with other natural hazards globally that are difficult to detect at microscale.

8.2 Future Works

Plounge App: SNS for weather observation

In continuation with the calibration of smartphone-based weather measurements, an application designed to obtain more data called the Plounge App was developed (Fig. 35).

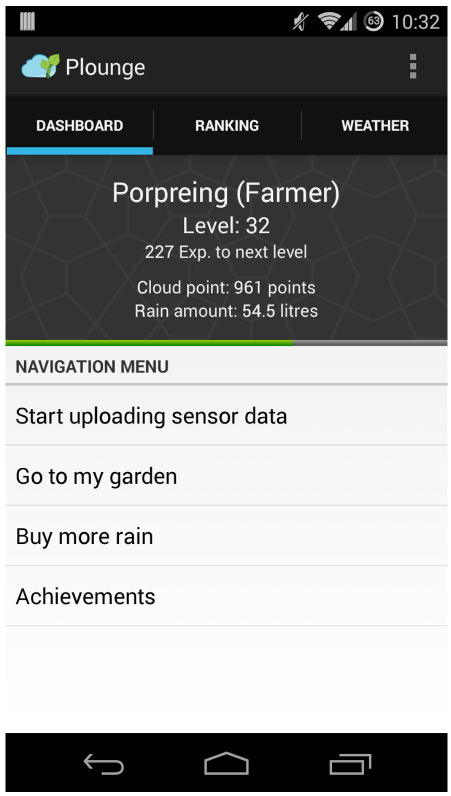


Figure 35: Plounge App dashboard screenshot

It is currently an Android-based smartphone application that integrates social networking service (SNS) with weather measurements from built-in sensors running on the background. The application originated from a small-scale experimentation of smartphone-based weather measurements. The need to acquire more data stemmed from the issue of whether a network of point weather sensors, as such in the form a smartphone device, can actually describe current weather conditions.

The idea of Plounge App revolves around a game theory of gaining points to grow virtual plants and achieve goals in competition with friends online (Fig. 36)

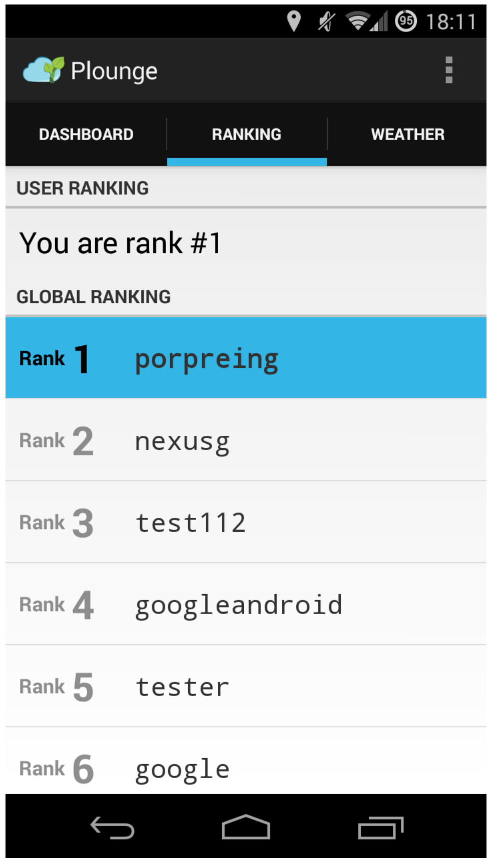


Figure 36: Plounge App ranking among friends

Users of Plounge App can participate in an SNS-based online game while actually harvesting important weather data to support the realization of an pin-point weather information database via a network of in-situ observation system. So far, the development of the application is ongoing with several test measurements from within the campus using only 7 smartphone devices with built-in environmental sensors.

Inside the Plounge App, the user can also view the current weather conditions only from within the friendship network (Fig. 37). For now, this is a simple text-based pop-up whenever a user hovers over a point that is displayed on the map.

Figure 37: Image of the text-based data on a map

Reliability evaluation of context-tagged weather data

More than the amount of data needed, there also needs to be a way to manage and analyse these information to make sure that data in the Plounge App users is reliable. Following this idea, the first step is to "tag" each observed information with the actual situation that the device is in. For instance, based on the environmental data taken from the built-in sensors as in Table 15, it is possible to infer if the device is kept in the shirt or pants pocket, or placed inside a bag. Ideally, when making background measurements, the device is in a stable state and outside any compartment to make it measure the current environment with maximum reliability as possible. However, in reality, the smartphones are meant to be used as it is, and thus, it is securely kept by its owner inside their pockets or bags for most of the time. With such conditions, eliminating errors and increasing

Data	Type
Accelerometer	3D, numeric, integer
Orientation	3D, numeric, integer
Light	numeric, constant
Proximity	binary (near or far)
Humidity, Temperature, Pressure	numeric

Table 15: Environmental data from smartphones

the reliability of, i.e., humidity, temperature, and pressure data by integrating the error that is caused by erroneous measurements by smartphones in actual use like when kept inside the pocket.

References

[1] Gerald Bieber, Jörg Voskamp, and Bodo Urban. Activity recognition for everyday life on mobile phones. In Constantine Stephanidis, editor, *Universal Access in Human-Computer Interaction. Intelligent and Ubiquitous Interaction Environments*, volume 5615 of *Lecture Notes in Computer Science*, pages 289–296. Springer Berlin Heidelberg, 2009.

[2] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. Gossip algorithms: design, analysis and applications. In *Proc. of the IEEE International Conference on Computer Communications*, volume 3, pages 1653–1664, Mar. 2005.

[3] Giorgio Conte, Massimo De Marchi, Alessandro A. Nacci, Vincenzo Rana, and Donatella Sciuto. Bluesentinel: A first approach using ibeacon for an energy efficient occupancy detection system. In *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, BuildSys '14, pages 11–19, New York, NY, USA, 2014. ACM.

[4] Davis. Vantage Pro2, 2013.

[5] A.G. Dimakis, S. Kar, J.M.F. Moura, M.G. Rabbat, and A. Scaglione. Gossip algorithms for distributed signal processing. *Proceedings of the IEEE*, 98(11):1847–1864, Nov. 2010.

[6] DOST-Project NOAH. Automated surface observing system (ASOS), April 2014.

[7] H.K. Fard, Yuanzhu Chen, and Kyung Kook Son. Indoor positioning of mobile devices with agile ibeacon deployment. In *Electrical and Computer Engineering (CCECE), 2015 IEEE 28th Canadian Conference on*, pages 275–279, May 2015.

[8] Colin J. Ferster and Nicholas C. Coops. A review of earth observation using mobile personal communication devices. *Computers & Geosciences*, 51(0):339 – 349, Feb. 2013.

[9] Davide Figo, Pedro C. Diniz, Diogo R. Ferreira, and João M. Cardoso. Preprocessing techniques for context recognition from accelerometer data. *Personal and Ubiquitous Comput.*, 14(7):645–662, October 2010.

[10] S.M. George, Wei Zhou, H. Chenji, Myounggyu Won, Yong Oh Lee, A. Pazarloglou, R. Stoleru, and P. Barooah. Distressnet: a wireless ad hoc and sensor network architecture for situation management in disaster response. *Communications Magazine, IEEE*, 48(3):128–136, 2010.

[11] E. Lin J.J. Pereira R.T. Corlett X. Cui G.E. Insarov R.D. Lasco E. Lindgren Hijioka, Y. and A. Surjan. *Climate Change 2014: Impacts, Adaptation, and Vulnerability. Part B: Regional Aspects. Contribution of Working Group II to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change.* Cambridge University Press, 2014.

[12] Seyed Amir Hoseini-Tabatabaei, Alexander Gluhak, and Rahim Tafazolli. A survey on smartphone-based systems for opportunistic user context recognition. *ACM Comput. Surv.*, 45(3):27:1–27:51, Jul. 2013.

[13] Theus Hossmann, Franck Legendre, Paolo Carta, Per Gunningberg, and Christian Rohner. Twitter in disaster mode: Opportunistic communication and distribution of sensor data in emergencies. In *Proceedings of the 3rd Extreme Conference on Communication: The Amazon Expedition*, ExtremeCom '11, pages 1:1–1:6, New York, NY, USA, 2011. ACM.

[14] F. Ichikawa, J. Chipchase, and R. Grignani. Where's the phone? A study of mobile phone location in public spaces. pages 1–8, Nov. 2005.

[15] IFRC. *World Disasters Report 2013 Focus on technology and the future of humanitarian action.* Imprimerie Chirat, Lyon, France, 2013.

[16] François Ingelrest, Guillermo Barrenetxea, Gunnar Schaefer, Martin Vetterli, Olivier Couach, and Marc Parlange. Sensorscope: Application-specific sensor network for environmental monitoring. *ACM Trans. Sen. Netw*, 6(2):17:1–17:32, March 2010.

[17] ITU. Ict facts and figures: The world in 2015, May 2015.

[18] Jane Louie Zamora and Shigeru Kashihara and Suguru Yamaguchi. Measurement Results of IEEE802.11a WiFi Signal Attenuation Under Different Rain Conditions. In *IEICE Technical Report, MoMuC2012-3*, pages 19–24, May 2013.

[19] Japan Meteorological Agency. Observations, April 2014.

[20] Y. Kawahara, H. Kurasawa, and Hiroyuki Morikawa. Recognizing user context using mobile handsets with acceleration sensors. pages 1–5, May 2007.

[21] M. Kohne and J. Sieck. Location-based services with ibeacon technology. In *Artificial Intelligence, Modelling and Simulation (AIMS), 2014 2nd International Conference on*, pages 315–321, Nov 2014.

[22] N.D. Lane, E. Miluzzo, Hong Lu, D. Peebles, T. Choudhury, and A.T. Campbell. A survey of mobile phone sensing. *Communications Magazine, IEEE*, 48(9):140–150, Sept 2010.

[23] Franck Legendre. 30 years of ad hoc networking research: What about humanitarian and disaster relief solutions? what are we still missing? In *Proceedings of the 1st International Conference on Wireless Technologies for Humanitarian Relief*, ACWR '11, pages 217–217, New York, NY, USA, 2011. ACM.

[24] N. J C Libatique, G.L. Tangonan, R. Gustilo, W.K.-G. Seah, C. Pineda, M.L. Guico, G. Abrajano, R. Ching, J.L. Zamora, A. Espinosa, A.C. Valera, R. Lamac, H. Dy, J. Pusta, E. M. Trono, A. Gimpaya, J.R. San Luis, S. J. Gonzales, and A. T. Lotho. Design of a tropical rain - disaster alarm system: A new approach based on wireless sensor networks and acoustic rain rate measurements. In *Instrumentation and Measurement Technology Conference, 2009. I2MTC '09. IEEE*, pages 1337–1340, 2009.

[25] Guangwen Liu, Khan Muhammad Asif Hossain, Masayuki Iwai, Masaki Ito, Yoshito Tobe, Kaoru Sezaki, and Dunstan Matekenya. Beyond horizontal location context: Measuring elevation using smartphone's barometer. In *Proc. of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*, pages 459–468, 2014.

[26] Weiquan Lu et al. Communications support for disaster recovery operations using hybrid mobile ad-hoc networks. In *The 32nd IEEE Conference of Local Computer Networks*, pages 763–770, 2007.

[27] Henar Martín, Ana M. Bernardos, Josué Iglesias, and José R. Casar. Activity logging using lightweight classification techniques in mobile devices. *Personal Ubiquitous Comput.*, 17(4):675–695, April 2013.

[28] Bill McGuire. *Waking the giant: How a changing climate triggers earthquakes, tsunamis, and volcanoes.* Oxford University Press, New York, USA, 2012.

[29] Kevin Ni, Nithya Ramanathan, Mohamed Nabil Hajj Chehade, Laura Balzano, Sheela Nair, Sadaf Zahedi, Eddie Kohler, Greg Pottie, Mark Hansen, and Mani Srivastava. Sensor network data fault types. *ACM Trans. Sen. Netw.*, 5(3):25:1–25:29, June 2009.

[30] Sergio F. Ochoa and Rodrigo Santos. Human-centric wireless sensor networks to improve information availability during urban search and rescue activities. *Information Fusion*, 2013.

[31] A. Overeem, J. C. R. Robinson, H. Leijnse, G. J. Steeneveld, B. K. P. Horn, and R. Uijlenhoet. Crowdsourcing urban air temperatures from smartphone battery temperatures. *Geophysical Research Letters*, 40(15):4081–4085, Aug. 2013.

[32] N. Ramakrishnan, E. Ertin, and R.L. Moses. Gossip-based algorithm for joint signature estimation and node calibration in sensor networks. *Selected Topics in Signal Processing, IEEE Journal of*, 5(4):665–673, Aug 2011.

[33] Samsung. Samsung GALAXY Nexus, 2014.

[34] Samsung. Samsung GALAXY S4, 2014.

[35] Samsung. Samsung GALAXY SIII, 2014.

[36] O. Sendik and H. Messer. A new approach to precipitation monitoring: A critical survey of existing technologies and challenges. *Signal Processing Magazine, IEEE*, 32(3):110–122, May 2015.

[37] Inc. Sensorcon. Sensordrone, 2013.

[38] Eiichi Shiraishi. Trends of technology for observing and forecasting localized rains. *Science & Technology Trends Quarterly Review*, (32), July 2009.

[39] Rui Tan, Guoliang Xing, Xue Liu, Jianguo Yao, and Zhaohui Yuan. Adaptive calibration for fusion-based cyber-physical systems. *ACM Trans. Embed. Comput. Syst.*, 11(4):80:1–80:25, January 2013.

[40] Vaisala. Vaisala Weather Transmitter WXT520 Access to Real Time Weather Data, 2013.

[41] Inc. Weathernews. Sorayomi Master, 2013.

[42] World Bank. World development report 2010: Development and climate change, 2010.

[43] Xian Wu et al. Emergency message dissemination system for smartphones during natural disasters. In *Proc. of ITS Telecommunications (ITST)*, pages 258–263, 2011.

[44] Zhaowen Xing and Le Gruenwald. Managing concurrent execution of transactions in mobile ad-hoc network database systems: an energy-efficient approach. *Distrib. Parallel Databases*, 31(2):183–230, June 2013.

[45] Jane Louie Fresco Zamora, Shigeru Kashihara, and Suguru Yamaguchi. A Smartphone-based Warning System for Guerilla Rain. In *IEICE Technical Report, MoMuC2012-3*, pages 9–14, May 2012.

Appendix

A. List of Publications

A.1 Journal

1. Jane Louie Fresco Zamora, Shigeru Kashihara, and Suguru Yamaguchi, "Calibration of Smartphone-Based Weather Measurements Using Pairwise Gossip", The Scientific World Journal, February 2015.

2. Louie Zamora, Noriyuki Suzuki, Hiroaki Takemoto, Shigeru Kashihara, Suguru Yamaguchi, "Securing SOS Messages in Uncommunicable Areas via Information Sharing Cluster", Special Section on Information and Communication Systems for Safe and Secure Life, IEICE Transaction Fundamentals, Vol. E98-A, No. 8, August 2015.

A.2 International Conference

1. Jane Louie Fresco Zamora, Naoya Sawada, Takemi Sahara, Shigeru Kashihara, Yuzo Taenaka, Suguru Yamaguchi. Surface Weather Observation via Distributed Devices. In Proc. of 2014 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), May 2014.

2. Jane Louie Fresco Zamora, Noriyuki Suzuki, Hiroaki Takemoto, Shigeru Kashihara, Yuzo Taenaka, Suguru Yamaguchi. Battery-saving Message Collection Method for Disrupted Communication Service Areas. In The 11th Annual IEEE Consumer Communications & Networking Conference (CCNC), January 2014.

3. Jane Louie Fresco Zamora, Shigeru Kashihara, Suguru Yamaguchi. Radio Signal-based Measurements for Localized Heavy Rain Detection using Smartphones. In Proc. of IEEE Global Humanitarian Technology Conference (GHTC), October 2013.

4. Noriyuki Suzuki, Jane Louie Fresco Zamora, Shigeru Kashihara, and Suguru Yamaguchi. SOSCast: Location Estimation of Immobilized Persons

through SOS Message Propagation. In Proceedings of the 4th International Conference on Intelligent Networking and Collaborative Systems (INCoS 2012), September 2012.

5. Noriyuki Suzuki, Jane Louie Fresco Zamora, Shigeru Kashihara, and Suguru Yamaguchi. Using SOS Message Propagation to Estimate the Location of Immobilized Persons. In Proceedings of the 18th annual international conference on Mobile computing and networking (MobiCom 2012), Demo session, August 2012.

A.3 Technical Report

1. Hiroaki Takemoto, Jane Louie Fresco Zamora, Shigeru Kashihara, Yuzo Taenaka, Mineo Takai, Shigeru Kaneda, and Suguru Yamaguchi, "Evaluation of information-Sharing Method among Disaster Victims in A Collapsed Communication Service Area," IEICE Tech. Rep., vol. 113, no. 304, MoNA2013-43, pp. 11-16, Nov. 2013. (In Japanese)

2. Shigeru Kashihara, Yuzo Taenaka, Jane Louie Fresco Zamora, Suguru Yamaguchi.Problem Analysis and Proposed Solutions for SOSCast. IEICE Tech. Rep., Vol.113, No.168, MoNA2013-23, pp. 59-62, Aug. 2013. (In Japanese)

3. Jane Louie Zamora, Shigeru Kashihara, and Suguru Yamaguchi. Measurement Results of IEEE802.11a WiFi Signal Attenuation Under Different Rain Conditions. IEICE Tech. Rep., vol. 113, No. 56, MoNA2013-4, pp. 19-24, May 2013.

4. Noriyuki Suzuki, Jane Louie Fresco Zamora, Shigeru Kashihara, Suguru Yamaguchi. Performance Evaluation of An Information Collection Method for Improving Battery Consumption and Searching Efficiency in A Communication-Disrupted Area. IEICE Tech. Rep., vol.112, no.493, MoMuC2012-73, pp. 329-334, Mar. 2013. (In Japanese)

5. Maricris Marimon, Jane Louie Zamora, Diego Javier Reinoso, Noriyuki Suzuki, Gemalyn Abrajano, Shigeru Kashihara. Radioactivity Level Monitoring

System via Wireless Sensor Networks. IEICE Tech. Rep., vol. 112, no. 404, MoMuC2012-58, pp. 87-88, Jan. 2013.

6. Maricris Marimon, Jane Louie Zamora, Diego Javier Reinoso, Noriyuki Suzuki, Gemalyn Abrajano, Shigeru Kashihara. An Implementation Design of a Radioactivity-Level Monitoring System via Wireless Sensor Networks. IEICE Tech. Rep., vol. 112, no. 404, MoMuC2012-64, pp. 107-112, Jan. 2013.

7. Noriyuki Suzuki, Jane Louie Fresco Zamora, Shigeru Kashihara, Suguru Yamaguchi. Collecting Disaster Victims' Information by Wi-Fi Direct and Bluetooth. IEICE Tech. Rep., vol. 112, No. 308, MoMuC2012-39, pp. 39-44, Nov. 2012. (In Japanese)

8. Jane Louie Fresco Zamora, Shigeru Kashihara, and Suguru Yamaguchi. A Smartphone-based Warning System for Guerilla Rain. In IEICE Technical Report, MoMuC2012-3, pp. 9-14, May 2012.

9. Noriyuki Suzuki, Toru Tsuda, Toshifumi Saito, Kyohei Moriyama, Jane Louie Fresco Zamora, Shigeru Kashihara, Kazutoshi Fujikawa, Suguru Yamaguchi. SOSCast : Design and Implementation of Rescue Message Propagating Application. IEICE Tech. Rep., Vol.111, No.384, MoMuC2011-45, pp. 49-54, Jan. 2012. (In Japanese)

10. Toru Tsuda, Noriyuki Suzuki, Toshifumi Saito, Kyohei Moriyama, Jane Louie Fresco Zamora, Shigeru Kashihara, Kazutoshi Fujikawa, Suguru Yamaguchi. SOSCast : Rescue Call Propagating Application over Store-Carry-Forward Communication using Bluetooth Devices. IEICE Tech. Rep., Vol.111, No.384, MoMuC2011-52, pp. 83-84, Jan. 2012. (In Japanese)