An Analysis of Distance Estimation to Detect Proximity in Social Interactions

Venet Osmani, Iacopo Carreras, Aleksandar Matic, Piret Saar
CREATE-NET, via alla Cascata 56/D, Povo, Trento, Italy
{name.surname@create-net.org}

Abstract. In the area of human behaviour analysis, smartphones are opening new possibilities where a multitude of embedded sensors can be used to regularly monitor users' daily activities and interactions in a non-obtrusive way. In this paper we focus on proximity detection, which refers to the ability of a system to recognize the co-location of two or more individuals and infer interpersonal distances. We present *Comm2Sense*, our mobile platform to detect proximity among users exploiting sensing capabilities available in modern smartphones, namely Wi-Fi hotspot and Wi-Fi receiver. The platform estimates the distance between subjects applying data mining techniques to the analysis of the Wi-Fi RSSI. We describe the design and implementation of the platform, together with the technical solutions implemented in each module. We demonstrate that the proposed platform is able to achieve a resolution of 0.5 meters.

1 Introduction

Human behaviour monitoring is an area that has been getting increasing attention from the research community. Review of the current literature shows that the monitoring methods have taken a number of approaches in terms of sensing modalities, including video (Groh et al. 2012), purpose built devices (Sociometric Solutions, 2012) radio signal (Banerjee et al. 2010) or a combination of these modalities. However, relying on a monitoring infrastructure, such as video for example, limits the monitoring within the confines of the reach of the cameras thus reducing the scale of the experiments; while, dedicated devices may also affect natural behaviour of the subjects due to continual awareness of being monitored. As such a desirable solution relies on using already accepted devices that can fade in the background, while being fully mobile and not spatially confined within the reach of an infrastructure.

A good candidate that can address these issues is mobile phone. Once used for voice communication, mobile phones are becoming powerful mobile platforms, able to sense multitude of phenomena from their embedded sensors, such as GPS, accelerometer, compass, Wi-Fi. Given their personal nature and the rate of adoption, mobile devices are increasingly becoming a proxy to users' habits and daily patterns. This is opening new possibilities in the area of human behaviour analysis, where many sensors typically embedded in smartphones can be used to regularly monitor users' daily activities and interactions in a non-obtrusive way.

adfa, p. 1, 2011.

Considering these characteristics of smart phones, in this paper we focus on proximity detection, which we define as the ability of a system to recognize the co-presence of two or more individuals, closely related to their social interactions. Proximity among individuals has a relevance to a number of different areas, including (but not limited to) i) healthcare - where the amount of socialization may have a direct, positive impact on the self-reported mood of people (House et al. 1988), ii) social network analysis - where interactions are seen as a proxy towards the social graph of a person (Eagle et al. 2009), iii) productivity - where social interactions are shown to be correlated to the productivity and (Fischbach et al. 2008) iv) epidemiology - where contacts among people represent the main cause behind the spreading of an epidemic (Madan, 2010). This paper builds on previous work presented at PerMoby workshop (Carreras 2012) and adds a new set of experiments pertaining to distance estimation when dealing with diverse hardware and different environments, in addition to explaining a new method on automatic calibration that addresses the diversity of hardware/environment issues.

2 Related Work

A myriad of solutions for recognizing social interactions that rely either on external infrastructure or dedicated devices has been proposed. The most utilized approach in various studies was the Social Badge (Sociometric Solutions, 2012), a pendant-like hardware that is worn at chest-level that use an infrared sensor which detects another badge in proximity and within the angle of view. It also includes the audio analysis for recognizing ongoing social interactions. However, this section reviews the related approaches that rely on the use of commodity devices, focusing on proximity and speech activity detection.

Related peer-based techniques reported in the literature aimed either to detect proximity of mobile devices or to estimate distance between devices and map them in a virtual plane. Current work on smart phone sensing to detect proximity has relied mostly on using Bluetooth to sense nearby devices. However, Bluetooth scans provide imprecise information about distance between devices since the communication range is in the order of ten meters. Due to this limitation, long-term measurements coupled with various statistical methods were necessary to acquire understanding of social networks or friendship structures (Eagle and Pentland 2005; Hidalgo et al. 2007; Eagle 2005). In order to compare the system proposed in this paper with state of the art systems, Table 1 selects the most significant achievements in peer-based distance estimation that can be implemented in smart phones.

Project	Accuracy	Method
Virtual Compass (Banerjee et al. 2010)	50th percentile error: 0.9m, 90th percentile: 2.7m	Wi-Fi + Bluetooth
BeepBeep (Peng et al. 2007)	50th percentile error: within 2cm	Acoustic-based
NearMe (Krumm et al. 2004)	RMS error: 10m-20m	Comparing Wi-Fi fingerprints

Relate System (Hazas et al. 2005)	50th percentile error: 2cm – 4cm	Ultrasound
Our System	50^{th} percentile error: 0.5m	Wi-Fi RSSI Analysis

Table 1 – Comparison of proximity/distance estimation system

Relate System (Hazas 2005) calculates the relative position of devices relying on custom ultrasound hardware. This approach provides a very accurate estimate of distance, with the median accuracy in the order of centimetres, but it requires ultrasound emitters/receivers that are not available in standard smart phones. Techniques that rely on ultrasound or detecting the phase offset of transmitted radio waves are difficult to implement using the hardware and APIs available on commodity mobile phones (Banerjee et al. 2010). NearMe compares clients' list of Wi-Fi access points and signal strengths to compute the distance between devices. Unlike localization system based on Wi-Fi fingerprints, NearMe does not rely on calculating an absolute location thus it requires no calibration and minimal setup. This method achieves relatively low accuracy in comparison to other systems with an RMS (Root Mean Square) error of 10 to 20 meters. BeepBeep (Peng et al. 2007) is a highly accurate acoustic-based system for estimating distance between devices, which requires only a set of commodity hardware – a speaker, a microphone and a form of device-to-device communication. Each device emits a sound signal and collects its own and a signal from its peer. Distance estimation is based on counting the number of samples between these signals and exchanging the time duration with its peer thus calculating two-way time of flight. The approach requires wireless communication for coordinating devices and for exchanging the time duration. Noisy environments impact the accuracy of the system while the devices that are not in earshot cannot be detected; this limits applicability for smart phones considering the fact that they are typically carried in places that affect sound propagation including pockets, cases and bags especially when moving. Virtual Compass (Banerjee et al. 2010), which is to the best of our knowledge, the only approach similar to ours that exploits transmitting mechanisms embedded in smart phone and performs RSSI analysis. Translating RSSI to distance was performed with empirical propagation models enhanced by incorporating the uncertainty which provided the average accuracy of 3.4m and 3.91m when Bluetooth and Wi-Fi (respectively) were tested separately using nine devices in a 100m² indoor area. The fusion of the two transmitting mechanisms achieved the median error of 1.41m for nine devices while in the case of two devices in the same area the median error was 0.9m and the 90^{th} percentile error was 2.7m. In comparison to the distance estimation method proposed in this paper, the advantages of Virtual Compass includes estimating positions of devices in 2D plane, algorithms for energy efficient use and not requiring training phase. However, our system provides higher accuracy with the use of solely Wi-Fi, does not require communication between devices and broadcasting the distance to each of peers, while training phase is facilitated with a fast calibration method which makes the approach adaptive to different applications, environments and phone models.

3 Our Approach

In contrast to previous studies, we propose Comm2Sense, a mobile platform that exploits sensors embedded in mobile phones to detect the proximity between two (or more) individuals with the median accuracy of 0.5 m. In order to establish benchmarks for the recognition of interpersonal distances related to social interactions, we refer to the study of proxemics. The overall aim of this work is to provide and evaluate a method that can be widely deployed as a mobile phone application in order to detect spatial context of the subjects in social interactions. The distance estimation is based on the ability of modern devices to act both as Wi-Fi transmitters (known as tethering or portable hot spot mode) and Wi-Fi receivers. Our approach maps RSSI values to distances relying on supervised learning, thus trading-off between the accuracy in distance estimation and the user effort in signal fingerprint collection. The reason for using a more costly method in terms of the end user effort is the fact that one of the pre-dominant factors affecting RSSI patterns is receiver's characteristics (Bhagwat et al. 2004) that can lead to better system's accuracy. This hypothesis was tested in the experiments, demonstrating that environmental factors have less prevailing impact on RSSI patterns than receiver's characteristics due to relatively short distances and no obstacles between receiver and transmitter. Unlike time-consuming measurements typically required for fingerprinting methods, the user effort is decreased to only a couple of minutes to calibrate the phone signal while achieving a comparable accuracy to full fingerprinting method.

The main contribution of this paper can be summarized as follows:

- We present Comm2Sense, a platform for distance estimation. This is achieved by estimating distances between smartphones through data mining techniques applied to the Wi-Fi RSSI.
- We introduce a duty-cycle for letting smartphones alternatively act as portable hot spots and Wi-Fi clients. This is a necessary requirement to enable phones to mutually detect each other.
- We present the analysis and evaluation of the distance estimation system using real-world scenarios.
- We present a method for deploying the system, without the need for time demanding calibration procedures.

The section that follows details the system design goals, architecture design and its integral components.

4 System Design

In this section, we present the design of our Comm2Sense system, intended to infer interpersonal distances using technologies available in modern smartphones. The distance between two users that carry smartphones (one phone acting as a Wi-Fi receiver, the other one as a Wi-Fi transmitter) is estimated through the Wi-Fi RSSI analysis and data mining techniques. The overall approach is shown in Figure 1.

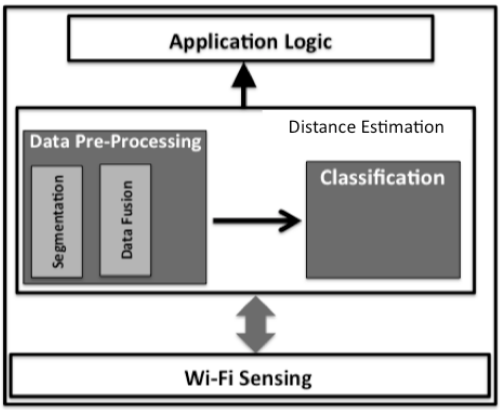

Figure 1 Comm2Sense software architecture

4.1 Design Goals and Architecture

The main design goals of the Comm2Sense system can be summarized as follows:

• fine granularity distance recognition: we aim to detect distances between two or more smartphones with a resolution less than 1m, thus overcoming the limitation of the current smartphone-based systems which mostly infer only co-location of subjects (such as Bluetooth scans (Eagle and Pentland 2005; Hidalgo et al. 2007; Eagle 2005) or indoor positioning (Krumm et al. 2004)). We envision that the proposed platform can afford a wide range of applications scenarios;

• non reliance on external infrastructure: the design of Comm2Sense system allows the operation in a fully distributed fashion, without relying on an external infrastructure or on additional hardware but smartphones. Distance is estimated solely on the basis of the information acquired through the smartphone sensing;

• ease of deployment: we aim to provide the platform which is easily deployable without requiring expert knowledge or performing time-consuming configuration operations.

We have designed and implemented Comm2Sense on the Android platform. The overall software architecture is shown in Figure 1. The system has been designed as a middleware platform, which is decoupled from the application logic. The middleware autonomously detects proximity and issues callbacks to the application layer, whenever one is detected. The application is then responsible for handling such information in the most appropriate way, depending on the scenario being supported.

In the sections that follow, we will describe each system component.

4.2 Wi-Fi Sensing

The Wi-Fi sensing module exploits the ability of modern smartphones to operate both in transmitting mode (colloquially known as portable hot spot - PHS) and as Wi-Fi clients. This allows discovery of peers close by, when acting as a Wi-Fi client, but also allows the Wi-Fi client to be discovered by other peers, when acting as PHS. The

Wi-Fi sensing module detects the presence of nearby PHSs, captures RSSI values and forwards such data to the data pre-processing component.

1) Duty Cycling

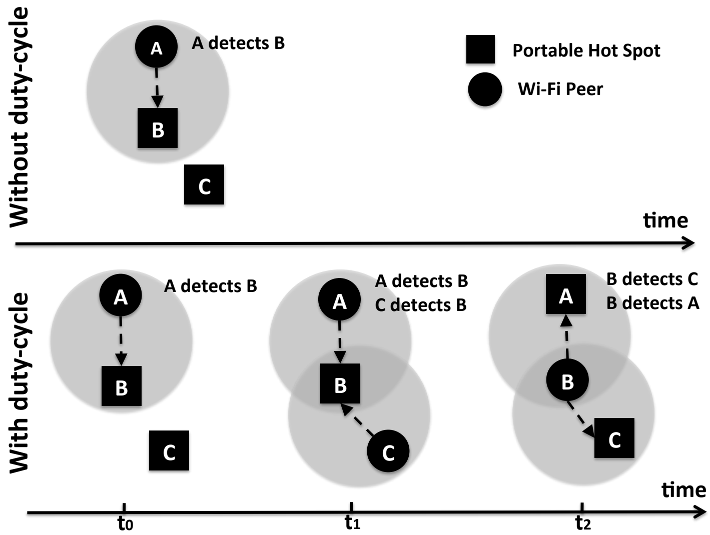

Figure 2 Proximity detection through RSSI analysis with and without the duty-cycle

In a “static” setting, nodes either act as Wi-Fi clients or PHS. However, Wi-Fi clients can detect the presence of PHS while the opposite does not hold, thus making proximity detection unidirectional. This issue is visualized in the upper part of Figure 2: as it can be seen, while node A is able to detect the presence of node B (in PHS mode), the same does not apply to nodes B and C, since both operate in PHS mode and can only be discovered. In order to overcome this limitation and make proximity detection bi-directional, each node implements a duty-cycle, acting as PHS and Wi-Fi client in alternating manner. This is illustrated in the lower part of Figure 2, where nodes change their role in cycles. By time t2 all nodes in the Wi-Fi communication range are able to discover each other. Smart phones, which support simultaneous activation of PHS and Client modes do not require duty cycling method to detect bi-directional proximity; however, such models of smartphones are still uncommon on the market.

The duty-cycle is depicted in Figure 3 and it encompasses four states characterized by the following permanence time:

- THSM: time spent in Portable Hot Spot (PHS) Mode. In this state, nodes are discoverable by other peers, but cannot discover other peers. This parameter can be configured at design time.
- TSM: time spent in Scan Mode (SM). In this state, nodes are searching for other peers at regular time intervals TSP. In this state nodes can discover other PHSs, but are not discoverable by other peers. This parameter can be configured at design time.

- TWUM: time required for switching from PHS to SM Mode. This time interval is composed by the time needed to switch OFF portable hot spot and the time needed to disconnect the Wi-Fi client. In this time interval, nodes are neither discoverable nor can discover other peers.
- TWDM: time required for switching from SM to PHS Mode. This time interval is composed by the time needed to switch OFF the Wi-Fi client and the time needed to disconnect the portable hot spot. In this time interval nodes are not discoverable, nor can discover other peers.

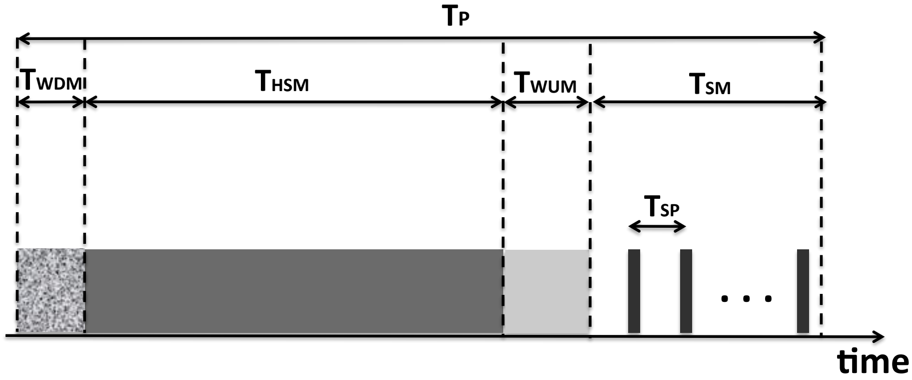

Figure 3 Duty cycle implemented by Comm2Sense nodes

The duty-cycle is defined at design time and determined by the time-resolution to be achieved. In this case, time-resolution refers to the minimum amount of time that two nodes are required to stay in proximity in order to detect each other. Specific randomization is introduced in order to avoid synchronized duty-cycling of nodes. As an example, the time spent in PHS mode is not THSM, but rather uniformly distributed in [0,2 × THSM]. In order to properly identify the time spent in each of the four states, we first measured the time needed for tearing up and down a portable hot spot. In the following table, we report the measurements performed using HTC Nexus S..

	Min (ms.)	Mean (ms.)	Max (ms.)	Std (ms.)
PHS activation	n.a	915	n.a.	n.a.
PHS deactivation	1246	1305	1764	206
Wi-Fi activation	4452	4573	4990	76
Wi-Fi deactivation	7	25	103	16

Table 2 Time needed for turning on and off PHS in HTC Nexus One

Activation of all the PHS functions may require up to 10 seconds; however, it requires less than 1 second for the PHS to start broadcasting its SSID (Service Set IDentifier), thus being discoverable by Wi-Fi clients in proximity. Since the Wi-Fi sensing is based only on the RSSI analysis available from the SSID, without the need to establish a data connection, this can be assumed as the time required to activate a PHS.

Having accurate measurements of this would require dedicated equipment and therefore we do not provide complete statistics in the Table 2.

Regarding parameters of the Comm2Sense duty cycle measured in our experiments, TWUM (PHS deactivation and Wi-Fi client activation) was 5878 ms while TWDM (Wi-Fi client de-activation and PHS activation) was 1025 ms. The Tsp, as measured from dedicated experiments, can be configured to be approximately 0,5 seconds.

The performed experiments proved that Wi-Fi duty-cycling is indeed feasible using off-the-shelf smartphones. Clearly, the granularity that it is possible to achieve is constrained by the time needed to bring up and bring down the PHS. However, for the targeted application scenarios (e.g., monitoring people social interactions during daily activities) we do not require a very fine-grained time granularity, which allows us to maintain a loose duty-cycle, thus reducing the impact on the normal utilization of the phone. Furthermore, in the short-term we expect smartphones to be equipped with more efficient wireless technologies. This includes for instance Wi-Fi direct, a wireless technology specifically targeted to dynamic ad-hoc networks that should allow a much faster network discovery and setup.

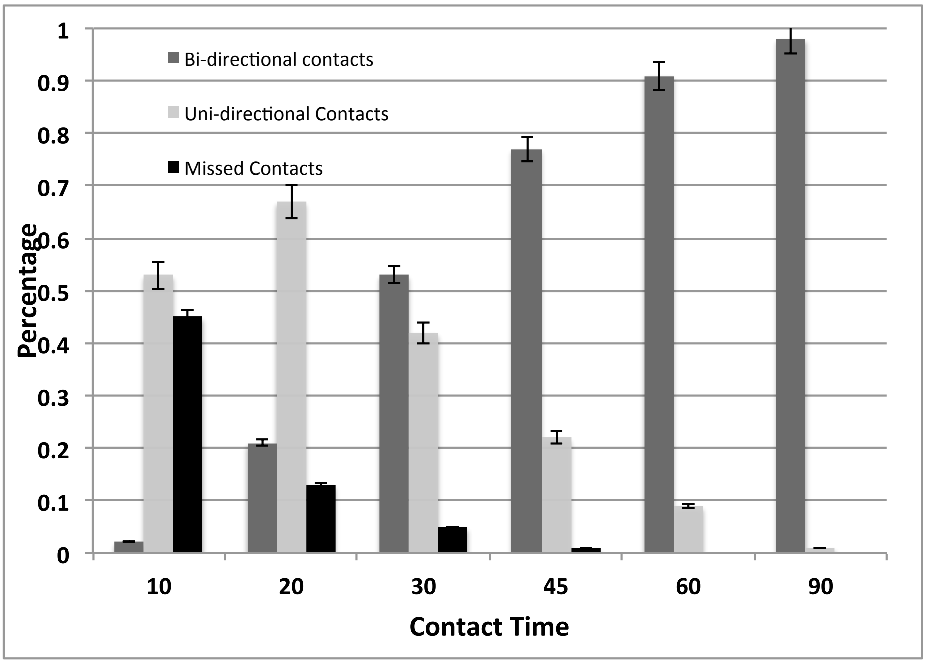

Figure 4 Proximity detection probability (THSM = 10 s, TSM = 10 s, TWUM = 6.878 s, TWDM = 1.025 s)

In order to properly project the values of the duty cycle, we implemented a simulator which analyzes the system parameters (T_{WUM}, T_{WUM}, T_{HSM} and T_{SM}), and provides the probabilities of (i) missing a contact (ii) having uni-directional contacts (only one node detecting the presence of the other) (iii) having bi-directional contacts (nodes mutually discovering each other). We have run 100 simulations for each of the considered cases, varying the random seed for each run. Figure 4 presents the detection probability, together with the corresponding 95% confidence interval, in the case of two nodes meeting, with T_{HSM} = 10 s., T_{SM} = 10 s T_{WUM} = 5.878 s and T_{WDM} = 1.025 s. As can be seen from the graph, using a contact duration of 60 s., nodes can be mutually detected with 80% probability.

Such probability, while not fully excluding the possibility of missing contacts, may be sufficient for application scenarios where relevant social interactions last for more than 60 seconds, such as healthcare scenarios or social network analysis.

2) Configuring the Transmission Power

The accuracy of the proposed approach strongly depends on the transmitting power of the Wi-Fi radio interface: the higher the transmission power, the larger the range and therefore the larger discovery radius. Furthermore, a higher transmission power leads to a reduced accuracy for estimation of distances in short range.

In order to evaluate the feasibility of distance estimation based on Wi-Fi RSSI, we explored the RSSI dependence on distance for three different transmitting power levels: 32 dBm (1.6 W) – maximal available power level, 0dBm (1 mW) - minimal power level, and medium power level of 13 dBm (20 mW). We performed the experiments recording 300 samples with the sampling rate of 1Hz for distances between 0,5 m and 8 m, every 0.5m for the three power levels. Experiments were run on an HTC Desire phone, however we do not expect considerable discrepancies for other Android smartphones. The transmitting power of 0 dBm provided the smoothest and the most monotone characteristics thus proving to be the best fit for short distance estimation (Figure 5). Therefore, we used 0 dBm as the reference transmission power for our system.

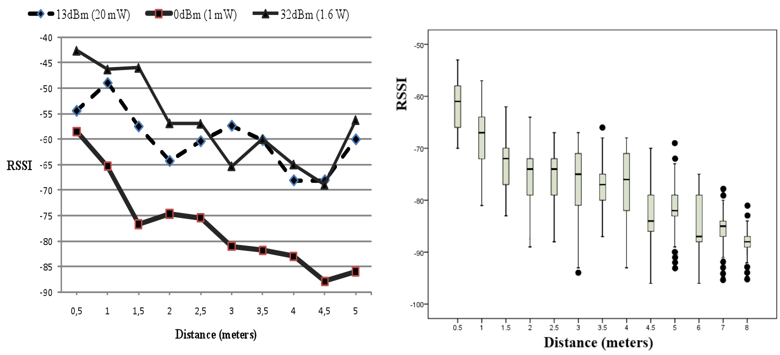

Figure 5 – RSSI dependence on the distance (three different power levels) and signal variance at 0dBm

Setting the transmission power to 0 dBm and controlling the switching on and off of the portable hot spot can be done via software, but requires installation of a modified firmware (Cyanogen-Mod) based on the Android operating system, which allows deeper access to system settings. We tested out the implementation over HTC Nexus One, Samsung Nexus S, HTC Desire and HTC Desire S phones.

4.3 Distance estimation

The distance estimation module consists of two parts: (i) a pre-processing module which is in charge of preparing the data generated by the data acquisition module for classification (ii) a classification module which determines the number of nodes in proximity and estimates their distance.

1) Data-preprocessing

The data-preprocessing module receives streams of raw data from the Wi-Fi sensing module, which are then prepared for the distance estimation module. This process is summarized in Figure 6.

The pre-processing consists of a segmentation phase, where raw data is aggregated over discrete time intervals, and the most relevant features for proximity detection are calculated. With respect to RSSI values, we grouped consecutive samples within 20 seconds and calculated signal characteristics for each group separately. This corresponds to approximately 20 samples, although in practice it may be less. It turned out that among the various tested signal characteristics (such as standard deviation, minimum and median), the combination of the mean and maximal value was proven empirically to provide the highest accuracy in distance estimation.

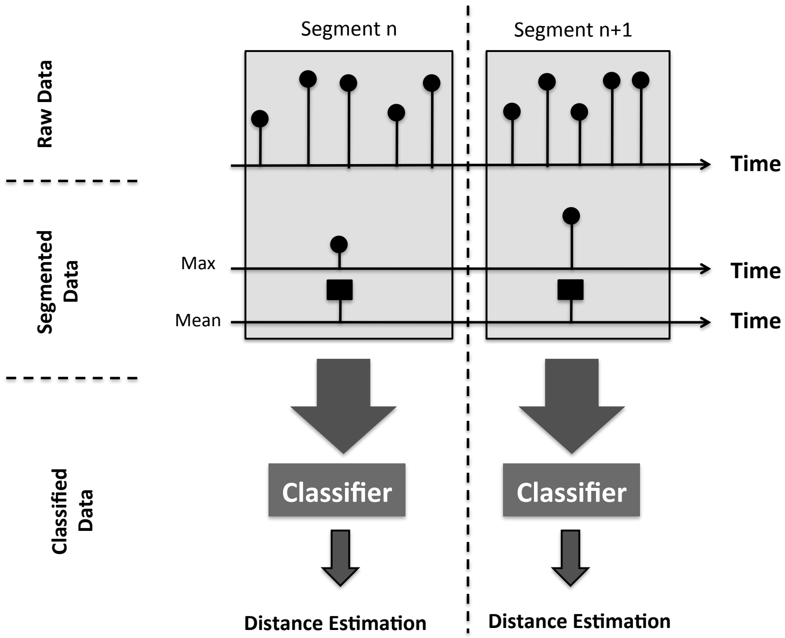

Figure 6 Distance estimation process

2) Distance estimation classifier

In order to estimate proximity between two devices, we used a classification algorithm. The motivation for this approach was in the instability and fluctuations of the Wi-Fi signal, typically due to environmental factors. From our initial study, we verified that a simple RSSI threshold analysis (assigning ranges of RSSI values to corresponding distances) did not suffice to obtain the required system accuracy. Therefore the classifier uses a set of features to estimate the distance with higher accuracy than the accuracy of a simple statistical approach.

The outcomes of this preliminary evaluation can be summarized as follows:

- Naive Bayes with Kernel Density Estimation (KDE) classifier proved to be the most accurate option. However, several classification techniques that we tested demonstrated similar performance in distance estimation. In terms of accuracy, it is possible to obtain a median estimation error (50th percentile) of approximately 0.5m.
- the classification accuracy pre-dominantly depended on the models of phones that were used for building the training and test set. In particular, performance degraded if different phones were used to train and evaluate the classifier. This is due to the fact that RSSI patterns highly depend on the receiver characteristics (Bhagwat et al. 2004), which are likely to be different across differentphone models. In order to address this issue, we have come up with a method where calibration can be performed automatically.

Detailed results of the classifier and a description of the experimental set up, follow in the next section.

5 Experimental set up and results

Our testbed consisted of seven smart phones including four different models, namely *HTC Desire*, HTC Desire S, *Samsung Nexus S* and *HTC Nexus One* with modified firmware to allow adjustment of transmitting power. Different phone units were distinguished by MAC addresses. Measurements were taken in three offices with dimensions of *12x8m*, *6x5m* and *6x3m*, a balcony of *12x2.5m* and a meeting room of *10x8m*. For testing the system's accuracy we used a pair of phones – one in transmitting and the other one in receiving mode.

Following a grid of 0.5m, RSSI was measured for 5 minutes on each distance between phones starting from 0.5m to the point in which either signal degraded to its minimal level or it was the furthest accessible point within room dimensions, which corresponded to the maximal distance in the experiments between 5 and 8 m.

Performance estimation was done by applying cross-validation – RSSI pattern captured in one out of five environments (three offices, balcony and a meeting room) which was used as training set while measurements from four remaining environments were used for testing. In this manner, the procedure was repeated to cover all the combinations regarding distinct training and test sets across five environments. The RSSI characteristics were calculated over every block of 20 samples and queried separately to estimate the unknown distance. The cumulative distribution function of the distance estimation error was plotted to evaluate the system's accuracy.

Figure 7 shows the system's accuracy in the case of using the same phone (the same model) acting as a receiver in both training and test phase. The median estimation error (50^{th} percentile) of approximately 0.5m was achieved using all the three classification algorithms. Naive Bayes with KDE showed a slightly better overall performance, providing distance estimation with 50th percentile error of 0.5m and 95th percentile error of 2m.

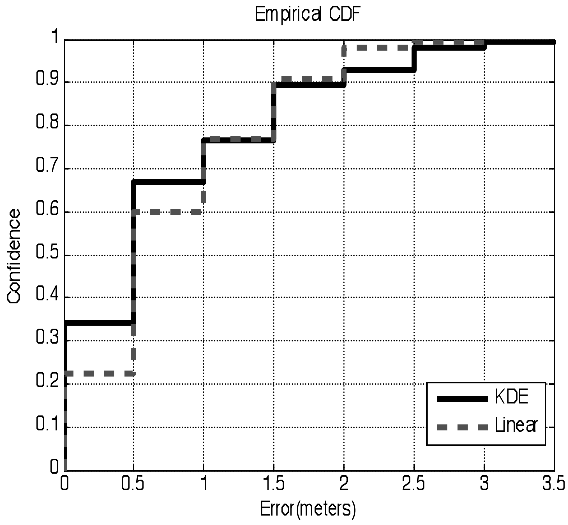

Figure 7 – Cumulative distribution function of the distance estimation errors (same receiving phone for training and test phases)

When different models of phones were used for training and test phase, the system's accuracy significantly degraded (Figure 8). In this case the median error was approximately 1m, while 95th percentile error was 3m. This occurred due to the fact that RSSI patterns highly depend on the receiver characteristics (Bhagwat et al. 2004), which are likely to be different across different phone models.

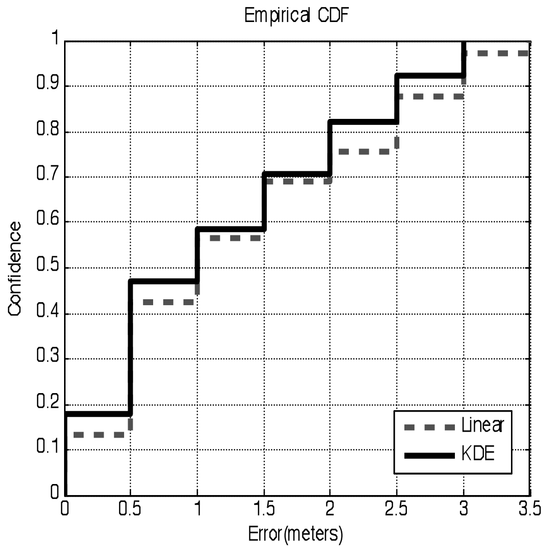

Figure 8 – Cumulative distribution function of the distance estimation errors (different receiving phone for training and test phases)

5.1 Automatic calibration

Our findings imply that, in order to obtain the desired distance estimation accuracy, it would be necessary to acquire a RSSI training set for each new phone model. As such, acquiring training set for each new phone model and for each training distance (0.5 m, 1 m ... 4.5 m, 5 m) is laborious, time consuming and generally unfeasible task. To address this problem, we opted to measure RSSI for a couple of minutes at a fixed distance (in this case 1m) and then build the training set utilizing the following propagation model:

$$P(d)[\mathrm{dBm}] = P(d_o)[\mathrm{dBm}] - 10n\log(\frac{d}{d_0}) - X \qquad (1)$$

The model describes the received signal strength *P(d)* as a function of the signal power $P(d_0)$ at a reference distance d_0 from the transmitter phone and the distance d from the transmitter and the emitter. In the equation, n is the path loss exponent, X is a component which reflects the sum of losses induced by each wall between the transmitter and receiver. This model, while being sensitive to reflections and multiple-path propagation, proved to be sufficiently accurate in case of line of sight, which is what we needed when estimating distances. The values of n and X depend on the environmental conditions (such as the building layout) (Bahl et al. 2000) and we have found empirically that the best value for the coefficient n was 1.5, while X was zero as there were no walls or other obstacles between points.

Starting from the above equation, we synthesized a training set using the following procedure:

- collect a set of RSSI measurements at a pre-defined distance (for instance, 1 m);
- create a synthetic training set by applying propagation model to all the samples collected in step 1 and for all distances relevant to Comm2Sense application scenarios;
- train the classifier with the generated synthetic training set.

The proposed algorithm runs on the smartphone requires only an initial calibration by end-users in order to collect the RSSI measurements at a known distance. From this initial calibration, Comm2Sense will train a phone-specific classifier, which will be able to detect the proximity of any other PHS.

In Figure 7 we report a preliminary evaluation of the classifier, built applying automatic calibration method described above. The experiments were conducted in the five environments (three offices, balcony and a meeting room). As it can be observed, all the tested models provided the median accuracy of 0.5m, while the 95th percentile error was between 2 and 2.5m. These results show an improvement in comparison to the results where different phones were used without automatic calibration (see Figure 8), where the improvement was from 1m to 0.5m median error. However, as expected, the performance was slightly lower in comparison to using a single phone model, with median accuracy remaining unchanged at 0.5m (see Figure 7), while 95th percentile error went from 2m to 2-2.5m depending on the classifier used.

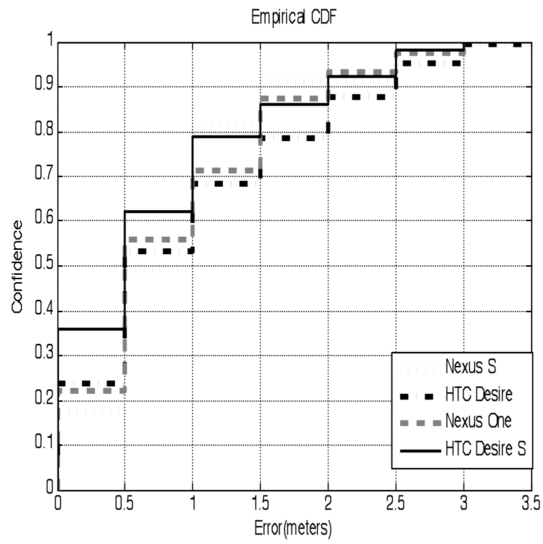

Figure 9 - Cumulative distribution function of the distance estimation errors - automatic calibration methods

In the next section we provide a description of Proxemics, which are set of measurable distances between people as they interact (Hall 1966) and we also describe how these distances can be estimated with our system.

6 Proximity and Social Interactions: Proxemics

The proposed system is envisioned to contribute to the detection of proximity between people as they interact in social settings, which we refer to as interpersonal distance. In order to establish benchmarks for the recognition of interpersonal distances related to social interactions, we refer to the study of proxemics. According to Hall, who defined the set of measurable distances between people as they interact (Hall 1966), there are four categories of interpersonal distance, including close phase (denoted with c) and far phase (denoted with f). For North American culture, categories of interpersonal distances include the following metrics: *intimate distance* (c: 0 – 0.15 m, f: 0.15 – 0.45 m), *personal distance* (c: 0.45 – 0.76 m, f: 0.76 – 1.2 m), *social distance* (c: 1.2 – 2.1 m, f: 2.1 – 3.6 m) and *public distance* (c: 3.6 – 7.6 m, f: 7.6 and more) (Hall 1966). Regarding proxemic behaviour, the four categories of interpersonal distance are typically utilized for the following activities: *intimate distance* for embracing, touching or whispering; *personal distance* for interactions among good friends or family members; *social distance* for interactions among acquaintances; *public distance* used for public speaking.

However, different cultures hold different standards of personal and social space; for example, in Latin cultures these distances are usually smaller than in Nordic cultures. Other parameters such as sex, age, extrovert/introvert personalities also affect setting interpersonal distances but considered to play a minor role (Groh et al. 2010). Yet, regardless of the metric corresponding to a different culture our distance estimation approach is easily scalable.

In order to infer interpersonal distances, we aim to distinguish the proximity of people with respect to personal, social and public space. The intimate space, which includes distances up to 0.45 m cannot be reliably detected using our method and we categorized all the recognized distances below 1.2 m as a personal space. The reason lies in the fact that the detection of such short distances between people is highly affected by the place of carrying the phone (such as a pocket, a case or a bag).

Our distance estimation accuracy is broken down based on the three classes – personal, social and public space and the results are presented in Table 3 in the form of a confusion matrix.

Ground-truth	**a) Same phone for training/test**			**b) Calibration method**		
	Personal	Social	Public	Personal	Social	Public
Personal	**81%**	19%	0%	**81%**	19%	0%
Social	0%	**67%**	33%	28%	**51%**	21%
Public	0%	17%	**83%**	2%	14%	**84%**

Table 3 - Break-down classification accuracy related to the categories of interpersonal distances defined by the study of proxemics

Distances related to personal and public space were recognized in more than 80% of cases both when a) performing training and test procedure with the same phone and b) using calibration method, as shown in Table 3. On the other hand, distinguishing social space from personal and public space resulted in lower accuracy, 67% and 51% with respect to different methods for acquiring the training set.

According to the study of proxemics, distances related to personal and social space are used by subjects for different types of social interactions thus distinguishing these two categories from public space would correspond to inferring the distances relevant for social interactions. Table 4 presents the results when the system's accuracy is broken down further in two groups – social interactions related distances and public space distances. In 82% - 86% of cases our system successfully distinguished the two groups, demonstrating the potential for various applications, including social networking analysis, healthcare domain, epidemiological and psychological studies.

Ground-truth	**a) Same phone for training/testing**		**b) Calibration method**	
	Social	Public	Social	Public
Social Interaction Dis-	**82%**	18%	**86%**	14%

tances				
Public Space	17%	**83%**	16%	**84%**

Table 4 - Recognizing two groups of distances related to social interactions and public space

7 Conclusion

Estimating distance to detect proximity between people can be applied not only in evaluating social interactions, but also in a number of other areas, including wellbeing where amount of social interaction impacts the reported mood; social network analysis; productivity, which is correlated with social interactions; and epidemiology where one factor of disease spread is proximity between people. The system presented in this paper provides a good basis to detect these and other phenomena related to proximity between people. The system has been developed on Android platform, where it has the potential to be widely deployed and does not rely on an external infrastructure nor is Wi-Fi hardware dependent. Comm2Sense exploits the ability of modern smartphones to act both as portable hot spots and Wi-Fi clients, and applies classification algorithms to analyse RSSI signal in order to estimate distance between phone and in turn proximity between people. The results demonstrate the ability of the system to detect proximity with a median error of 0.5m over a wide range of environments, both indoor and outdoor and a number of different phone models. These results were sufficient to detect Social Interaction distances and Public Space distances with an accuracy of 86% and 84% respectively. Future directions of this work will be focused on using this system to investigate the above-mentioned phenomena, commencing with a study of correlation between social interactions and mood changes.

8 References

"Sociometric Solutions." [Online]. Available: http://www.sociometricsolutions.com/. [Accessed: 25-Mar-2012].

Bahl P. and V. N. Padmanabhan, "RADAR: an in-building RF-based user location and tracking system," Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No.00CH37064), vol. 2, pp. 775-784.

Banerjee N., S. Agarwal, P. Bahl, R. Chandra, A. Wolman, and M. Corner, "Virtual compass: relative positioning to sense mobile social interactions," Pervasive Computing, pp. 1-21, 2010.

Bhagwat P., B. Raman, and D. Sanghi, "Turning 802.11 inside-out," ACM SIGCOMM Computer Communication Review, vol. 34, no. 1, p. 33, Jan. 2004.

Carreras I., Matic A., Saar P., Osmani V., "Comm2Sense: Detecting Proximity Through Smartphones", PerMoby 2012 workshop, part of IEEE PerCom 2012 confer-ence, Lugano Switzerland, 2012

Eagle N. and A. (Sandy) Pentland, “Reality mining: sensing complex social systems,” Personal and Ubiquitous Computing, vol. 10, no. 4, pp. 255-268, Nov. 2005.

Eagle N. N., “Machine perception and learning of complex social systems,” Massa-chusetts Institute of Technology, 2005.

Eagle N., A. S. Pentland, and D. Lazer, “Inferring social network structure using mo-bile phone data,” Proceedings of the National Academy of Sciences (PNAS), vol. 106, no. 6, pp. 15274-15278, 2009.

Fischbach K., P. a. Gloor, and D. Schoder, “Analysis of Informal Communication Networks - A Case Study,” Business & Information Systems Engineering, vol. 1, no. 2, pp. 140-149, Dec. 2008.

Groh G., A. Lehmann, J. Reimers, M. R. Frieß, and L. Schwarz, “Detecting social situations from interaction geometry,” in IEEE International Conference on Social Computing/IEEE International Conference on Privacy, Security, Risk and Trust, 2010.

Hall E., The hidden dimnesion. New York: Double Day Anchor Books, 1966.

Hazas M., C. Kray, H. Gellersen, H. Agbota, G. Kortuem, and A. Krohn, “A relative positioning system for co-located mobile devices,” Proceedings of the 3rd internatio-nal conference on Mobile systems, applications, and services - MobiSys '05, p. 177, 2005.

Hidalgo C. A. and C. Rodriguez-Sickert, “The dynamics of a mobile phone network,” Physica A: Statistical Mechanics and its Applications, vol. 387, no. 12, pp. 3017-3024, 2007.

House J.S., K. R. Landis, and D. Umberson, “Social relationships and health.,” Scien-ce (New York, N.Y.), vol. 241, no. 4865, pp. 540-5, Jul. 1988.

Krumm J. and K. Hinckley, “The NearMe Wireless Proximity Server,” In Procee-dings of International Conference on Ubiquitous Computing, pp. 283-300, 2004.

Madan A., M. Cebrian, D. Lazer, and A. Pentland, “Social sensing for epidemiologi-cal behavior change,” 12th ACM international conference on Ubiquitous computing, 2010.

Peng C., G. Shen, Y. Zhang, and Y. Li, “Beepbeep: a high accuracy acoustic ranging system using cots mobile devices,” Proceeding SenSys '07 Proceedings of the 5th international conference on Embedded networked sensor systems, 2007.