Next2Me: Capturing Social Interactions through Smartphone Devices using WiFi and Audio signals

Jon Baker
University of Kent
Canterbury, United Kingdom
j.baker@kent.ac.uk

Christos Efstratiou
University of Kent
Canterbury, United Kingdom
c.efstratiou@kent.ac.uk

ABSTRACT

Typical approaches in detecting social interactions consider the use of co-location as a proxy for real-world interactions. Such approaches can under-perform in challenging situations where multiple social interactions can occur in close proximity to each other. In this paper, we present a novel approach to detect co-located social interactions using smartphones. Next2Me relies on the use of WiFi signals and audio signals to accurately distinguish social groups interacting within a few meters from each other. Through a range of real-world experiments, we demonstrate a technique that utilises WiFi fingerprinting, along with *sound fingerprinting* to identify social groups. Experimental results show that Next2Me can achieve a precision of 88% within noisy environments, including smartphones that are placed in users' pockets, whilst maintaining a very low energy footprint (<3% of battery capacity per day).

CCS CONCEPTS

• **Human-centered computing → Ubiquitous and mobile computing systems and tools**; *Collaborative and social computing devices*;

KEYWORDS

Smartphone sensing, social sensing, WiFi, audio

ACM Reference Format:
Jon Baker and Christos Efstratiou. 2017. Next2Me: Capturing Social Interactions through Smartphone Devices using WiFi and Audio signals. In *Proceedings of the 14th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services (MobiQuitous 2017).* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3144457.3144500

1 INTRODUCTION

Social interactions represent a significant part of our daily lives. They are considered a significant aspect of the quality of people's daily lives [5], as well as an important activity that enables collaboration and creativity [16]. In recent years there has been an increasing interest in developing technologies that can capture the social behaviour of people. Within working environments, analysis of the social behaviour of employees has been shown to reflect the performance and productivity of teams [17]. Within the health and well-being domain, long-term tracking of social behaviour has been used as indicator for changes in mental health and perceived quality of life [21].

People-centric sensing technologies have been employed in a number of scenarios to develop systems that can passively capture social interactions. Wearable and mobile devices (e.g. smartphones) have been used to infer social behaviour by analysing the mobility patterns of individuals [18]. Traditionally, most of the approaches that have been used in such scenarios assume that proximity between individuals is an indicator of social interaction. This may be a valid assumption for certain situations (e.g. people participating in a meeting), but the assumption may not hold when considering situations where social interactions take place within crowded environments, involving multiple social groups. Such scenarios are very common in the daily lives of people. Having a chat in a crowded café, or interacting with different people during a networking session in a conference, are common situations where proximity may not be sufficient to correctly identify the people involved in an interaction.

In this work, we attempt to develop a system that can accurately capture social interactions within challenging scenarios where multiple social groups interact within close proximity of each other. In order to distinguish the closely located social groups, we rely on analysis of audio captured by the smartphones of the participants and aim to identify which social group they participate in. Our hypothesis is that the sound patterns captured by the smartphones of the people participating in a conversation is sufficiently different from the sound patterns of people not involved in that interaction event, or participating in a different social interaction even if the groups are within a few meters from each other. Intuitively, we consider that people participating in a conversation that takes place in a noisy or crowded environment have the tendency to raise their voices enough to be heard by the people involved in the conversation. This natural behaviour is enough to produce distinct sound patterns that are very similar for the people participating in the conversation, and sufficiently different from the sound patterns captured by the smartphones of people in near-by social groups. We demonstrate the design of a system that relies on a combination of WiFi fingerprinting and Audio fingerprinting captured by smartphones that are in the participants' pockets, or on tables in frond of them. Through a range of controlled experiments and a real-world deployment in a noisy café, we demonstrate that the system can achieve an average precision of 88%, while maintain a power consumption of less than 3% of the battery life per day.

MobiQuitous 2017, November 7–10, 2017, Melbourne, VIC, Australia

ACM ISBN 978-1-4503-5368-7/17/11…$15.00
https://doi.org/10.1145/3144457.3144500

2 RELATED WORK

Traditional techniques in passively capturing social interactions involves the use of RF technologies as a means of capturing the co-location of users. Examples include systems that use Bluetooth on smartphones [13, 15] or specialised wearable RF tags [1, 3] to detect face-to-face proximity. Although such techniques can offer an approximation of the social behaviour of users, ultimately co-location does not always imply social interaction.

WiFi fingerprinting has been widely used as a way of localising users within buildings [8, 9]. As such, the technique has also been used to detect co-location between users in environments where sufficient WiFi infrastructure is present [10]. However, such techniques suffer from similar limitations to other proximity based approaches where the co-location estimation does not imply social interaction. Recently, there has been significant work on the capture of social interaction passively through the collection of WiFi traces of users' smartphones using the WiFi infrastructure of a building [7, 23]. These techniques allow the tracking of social interaction without the need for the users to install a particular application on their phone. They allow the passive tracking of large numbers of users, but require access to the WiFi infrastructure of a given environment. In practical terms, these techniques can only be employed in certain environments, and do not allow the capture of social interactions throughout the daily lives of participants. Moreover, the passive tracking of smartphones without the need for an app installation raises significant privacy issues. Smartphone OS such as iOS have recently employed MAC obfuscation techniques to avoid such passive tracking thus rendering these approaches infeasible.

Since most social interactions contain speech between participants, it makes sense to use audio recordings as a technique for conversational detection. Some work uses the on-board microphones to record audio and use it for speaker recognition or conversational turns [6, 12, 18, 24]. Other work uses audio for indoor localisation [20] or proximity detection [22]. The work done in DSP.Ear [6] presents a smartphone system that extracts emotions from speech signals, estimates the number of speakers in a room [24], detects the identity of speakers and identifies common ambient sounds. This type of work relies heavily on the use of machine learning techniques, requiring training of the voice recognition component through appropriate sound datasets, often involving sound samples by the user. In many scenarios, such an approach would be infeasible for large scale deployment limiting the applicability of the approach.

3 MOTIVATION

Current approaches in capturing social interactions tend to rely on secondary signals such as co-location, as proxy for an actual social interaction. Indeed, when individuals are close to each other there is a high probability that they are interacting with each other. However, there are numerous scenarios where such approaches can lead to erroneous results. People working in shared office environments may be co-located but not interacting; when interacting with people in busy places, such as a restaurant or a social event, co-location may involve more people than somebody is interacting with. In order to enable a more accurate detection of social interaction, there is a need to move beyond co-location.

Audio has been shown to be a more accurate indicator of actual social interaction, as a means of capturing the actual conversations of people involved. However, relying on heavy-weight speech recognition or speaker recognition requires personalised voice training [18]. Such approaches do not scale well, as machine learning algorithms need to be trained with voice sample of participants.

In this work we aim to develop a system that combines co-location and audio sensing to accurately detect social interactions in challenging environments. Such environments involve close co-location of social groups, interacting within busy environments. In our approach we do not require prior training of the system with audio samples, neither from the users or the environment. Instead, we rely on the comparison between sound signals captured by the users' phones, as indicators of close proximity. Our motivation is based on the assumption that sound signals will have unique patterns for the people participating in a conversation, and are different from those in nearby conversations. Even in a noisy environment, people tend to talk louder to make sure that their conversation is heard by the participants from within the same group. Intuitively, we expect that sound samples captured by smartphones within a social interaction will have similar sound patterns, containing primarily the voices of the people participating. In our overall system, we utilise co-location as a means to trigger audio sensing when there is a high probability of social interaction. Sparsely sensed audio samples are then used to formulate a *"sound fingerprint"*. Comparisons between sound fingerprints are then used to discover the social networks of co-located users. The proposed approach does not require any special infrastructure, and can be used in any environment where there are sufficient WiFi signals to facilitate co-location sensing.

4 PRELIMINARY STUDY

In order to develop a robust system for capturing social interactions using smartphones, we initially attempted to explore the extent to which WiFi based proximity detection can enable the identification of such interactions. Furthermore, we also tried to explore how audio signals can be analysed to further assist in identifying social interactions. Our aim was to explore whether the combination of these two modalities can lead to a robust social sensing system that does not require prior training.

For the preliminary study, we developed a data collection app for Android smartphones. The app was running continuously, capturing WiFi and audio data in the background. Specifically, the app scanned for available WiFi access points every 10 seconds, and recorded the MAC addresses of the access points and the RSSI value of the signal strength received by each access point. At the same time, the application recorded audio continuously for the duration of the experiment.

Our aim was to target "challenging" scenarios where groups of people interact within close proximity of each other. We set up two experiments: "Experiment 1" representing a typical social interaction of a single group during a meeting, and "Experiment 2" where two groups were interacting within the same room in close

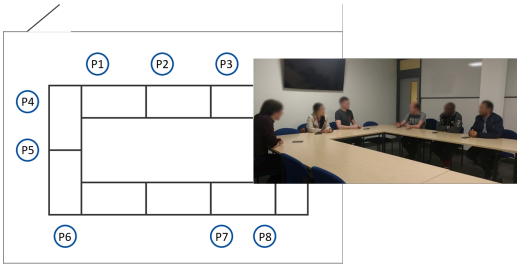


Figure 1: Set up of the meeting scenario experiment (Experiment 1)

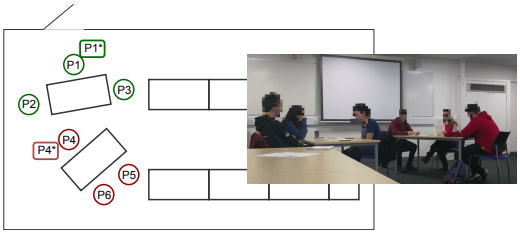


Figure 2: Set up with two groups interacting within the same room in close proximity (Experiment 2)

proximity. Specifically, the first experiment involved 10 participants joining a meeting and sitting around a large table. The participants were asked to keep their phones on the table during the meeting (see Figure 1). The data collection app was installed on the participants' phones before joining the meeting. In the second experiment, two groups of participants where asked to engage in two separate meetings within the same room. The experiment involved 6 participants (3 female, 3 male) who were asked to join the meetings. The participants were split into two groups of 3 people each, sitting on two tables with no more than 1 meter distance between them (see Figure 2). The participants were asked to place their smartphones on the table during the meeting, and two participants (one from each group) were asked to place an additional smartphone in their pocket. The two groups were asked to engage in conversations while sitting in close proximity to the other group.

4.1 WiFi Signals

Following similar work from [8], we explored how WiFi fingerprinting can be utilised to detect co-location in these experiments. Specifically, every smartphone device scans periodically every 10s for available WiFi access points which transmits at 2.4 GHz, and collects the received signal strength indicator (RSSI) and basic service set identifier (BSSID) for each access point response. These values are then used to generate a WiFi fingerprint for each participants' phone and subsequently estimate co-location between participants.

4.1.1 WiFi Fingerprint and Proximity. A WiFi scan performed at time t generates a signal strength vector

$$S_t = \{ap_1 : rss_1, \ldots, ap_n : rss_n\}$$

where each access point ap is identified by its MAC address, rss is the received signal strength value for ap. We generate a WiFi fingerprint for the smartphone of each participant, by aggregating multiple WiFi scans over a sliding window of $w = 60s$ with 33% overlap. Consider $SW_t = \{S_i : i \in (t - w, t)\}$ to be the set of subsequent scans within the window w. The WiFi fingerprint at time t is:

$$F_t = \{ap_1 : rss'_1, \ldots, ap_n : rss'_n\},$$

where $ap_i \in SW_t, \forall ap_i, rss'_i = avg(rss_i : rss_i \in SW_t)$. As it was shown in [10], the RSSI values captured by different smartphone models can vary significantly even when collected under identical conditions. In order to allow appropriate comparisons between WiFi fingerprints, the recorded RSSI values are normalised and converted to positive scale, by dividing them with the maximum RSSI within the fingerprint:

$$rss_i^{norm} = \frac{rss'_i + 100}{rss'_{max} + 100}$$

where rss'_i is the RSSI value for access point i i and rss'_{max} is the maximum RSSI value for the entries of averaged fingerprints.

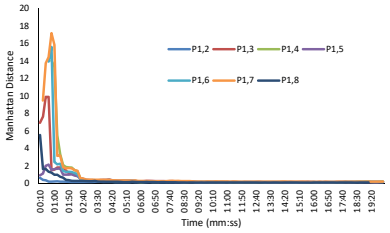
The generated fingerprints are then used to estimate the proximity between participants. Specifically, fingerprints generated by different participants are compared using a similarity function. We assume that the level of similarity is an indication of proximity between these participants. We applied the Manhattan distance as a similarity function, as it demonstrated the highest level of discrimination between different co-location distances when compared to Euclidean distance and Tanimoto similarity. Specifically, for any two fingerprints that were compared, each fingerprint was extended by adding access points that only appeared in the other fingerprint. The added access points were given an RSS value of 0dB. The similarity metric between the fingerprints was given by the Manhattan distance, with an additional division of common count to provide scaling [8]:

$$distance = \frac{1}{n} \sum_{i=1}^{n} |rss_i^a - rss_i^b|$$

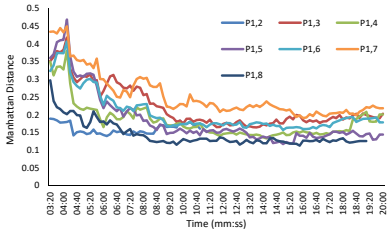
where n is the number of elements in the intersection between the two fingerprint sets: $n = |A \cap B|$, and rss_i^a and rss_i^b are the normalised RSS values for the access point i captured by the two devices a and b.

We used the WiFi scanning dataset to estimate the distance metric of the participants in the two controlled experiments. In Experiment 1, all participants joined a group meeting in the same room. The pair-wise distance metric for all participants over time clearly shows that the WiFi fingerprint can identify the participants joining the meeting (Figure 3a). Indeed the WiFi fingerprint comparison can clearly capture the sequence of arrival of the participants for example. However, when exploring the WiFi fingerprint similarities during the meeting we can see significant variations although the participants did not change their location during the meeting (Figure 3b).

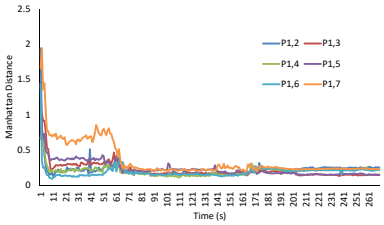
In our second experiment we attempt to explore how WiFi fingerprinting could be applied in the case of co-located social interactions. In that experiment, we have two groups of 3 people each, interacting in close proximity to each other (less 1 meter distance between the groups). Looking at the similarity of the WiFi fingerprints (Figure 3c), there is no obvious pattern that helps identify the two interacting groups. Furthermore, we explored the overall distribution of the similarity measurement (as Manhattan distance)



(a) Data of co-location during the meeting scenario, showing the pattern of co-location including participant 7 who arrived late



(b) Data from Figure (a) extracted from the section where all participants are interacting (minutes 3 to 20), and zoomed in



(c) Graph showing the Manhattan distance over time for the closely co-located social groups

Figure 3: Plots of the pair-wise distance calculated over the WiFi fingerprints captured by the participants' smartphones.

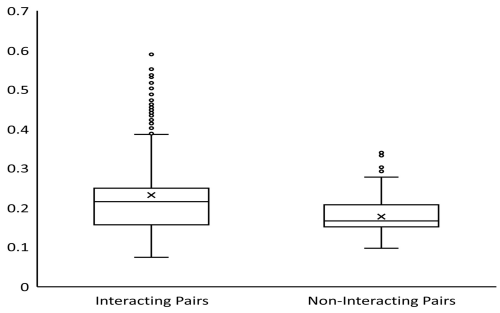


Figure 4: Box diagram showing the mean and spread of similarity values for interacting pairs vs non-interacting pairs

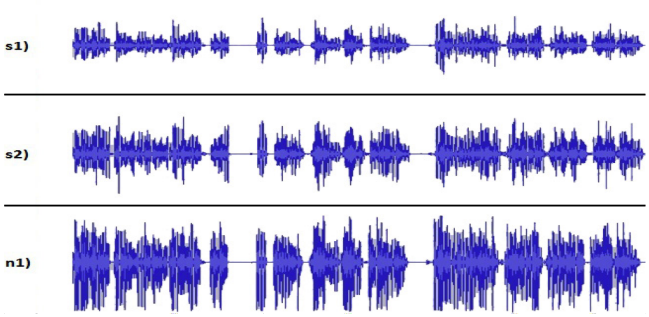


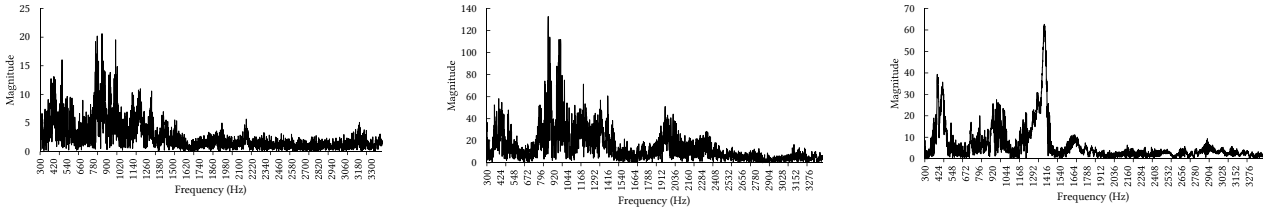
Figure 5: Sound waveforms captured by two Samsung S5 devices "s1", "s2" and a Nexus 5 "n1" recording the same speech segments at the same distance from the source. The devices have different gains.

between the pairs of participants that were interacting with each other, and compared it with the distribution of measurements between pairs that were not interacting. As shown in Figure 4, the two distributions may have slightly different median value, however, both distributions overlap significantly. This is a clear indication that for such close proximity between social groups, WiFi fingerprinting alone may not be sufficient to distinguish the two groups.

4.2 Audio Signals

Considering the limitations of using WiFi signals alone to detect social interactions, we explored the feasibility of relying on the capture of audio signals as a way of distinguishing social groups that are in close proximity to each other. One of our early intuitions was to explore how the amplitude of the sound signals can be used to identify the distance of the device from a speaker. This was an approach similar to the work shown in [4]. Before attempting to analyse audio sensing during the social interaction experiments, we first explored how audio signals are captured by different devices. We set up a test where a range of smartphone models were placed at the same distance from a speaker (approximately 1m) and captured the same audio of a human speaking. The smartphone devices used included two Samsung S5, two Nexus 5, a Motorola G and a Vodafone Smart Ultra 7. In order to avoid any irregular shaping of the captured signal, we ensured that the auto-gain function on the devices was not active. Audio was captured at 16 KHz sampling rate. We manually extracted the speech segments from the audio recordings and inspected the captured audio signals. As shown in Figure 5 different devices capture sound signals at different energy levels. Even the same models, can have significant differences. For example, the two Samsung S5 devices captured the same speech sounds at an average amplitude of -28.59 dB and -38.53 dB, while the Nexus 5 captured the same source at -34.24 dB. Such difference could be attributed to differences in the hardware design, wear and tear, or just dust that is accumulated around the microphone. These results demonstrate that to utilise the volume of the captured sound to estimate distance would require extensive calibration to identify a normalisation coefficient for each device. Although some solutions exist to calibrate the different gains between the microphones [14], this would require a supervised calibration phase whenever new smartphones are introduced to the system, leading to a system that is not scalable and not fit for unsupervised deployment. Furthermore, the differences between same phone models show that it would not be possible to construct a generic calibration database for different phone models.

As amplitude alone was not considered a sufficient feature to identify social groups, we attempted to explore if sound signals can reveal distinctive patterns that can help differentiate between



(a) A sample of frequency domain data from a device placed on "table a"

(b) A sample of frequency domain data from a different device placed on "table a"

(c) A sample of frequency domain data for a device placed on "table b"

Figure 6: Frequency spectrums of audio samples from different devices in Experiment 2, captured during the same time window.

people participating in the same conversation. In Experiment 2, participants were asked to place their phones on the table in front of them (tables a and b), while 2 participants had phones placed in their pockets (Figure 2). We analysed the sound signals captured by these devices, extracted samples of audio of 2 seconds and applied a Fast Fourier transformation (FFT) to look at the sound patterns in the frequency domain. We selected the frequencies between 300 Hz and 3,400 Hz, which is considered the speech range. This filtering allowed us to eliminate some of the noisy data that was captured by the phones in participants' pockets. Figure 6 shows the frequency patterns from two devices participating in the same conversation (Figures 6a and 6b), and a device on a different conversation nearby (Figure 6c). The patterns that we observe in this case show that devices participating in the same conversation have high energies around the same frequencies, while devices on different conversations show a significantly different pattern. Based on these observations, it is possible to devise a technique to extract a *"sound fingerprint"* that is based on the most significant frequencies of captured audio data, that can help distinguish between users participating in different conversations. Although the experiment involved participants in very close location, the difference in sound patterns can be explained by the natural tendency of people to speak loud enough so that all of their interacting participants can hear them. This in practice ensures that the sound captured by the phones in close proximity to the conversation is dominated by the speakers participating in that particular group.

Based on the findings of these preliminary studies, we aimed to design a system to detect social interactions using a combination of WiFi signals, as an early indicator that users are in close proximity, followed by audio sensing to identify smaller groups within the same area.

5 SYSTEM OVERVIEW

Next2Me is a mobile sensing system that can identify social interactions using WiFi and audio signals. The overall architecture can be seen in Figure 7. The system consists of sensing components running on a smartphone device, and a cloud service responsible for comparing datasets from multiple users. The system relies on WiFi fingerprinting to discover when a user is co-located with other users of the system. When co-location is detected, the participant's smartphones are triggered to perform sound sensing. The sound sensing subsystem is responsible for discovering similarities in the

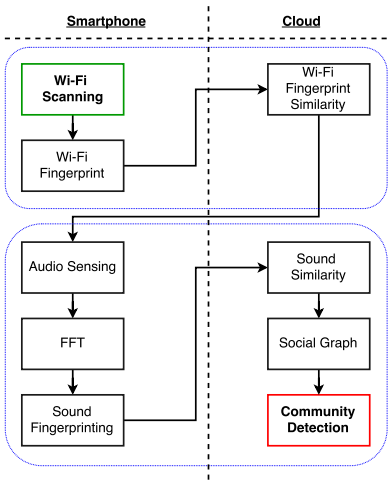


Figure 7: Block diagram overview of the entire system

sound patterns captured by the participating smartphones. The sound similarities are then used to identify a social network, as it is formed by the similarities of the sound signals. Finally, applying a community detection algorithm helps identify the sub-groups of people interacting within close proximity to each other. The following sections describe the system in more detail.

5.1 WiFi and co-location

The system relies on WiFi fingerprinting to detect co-location between users. Each smartphone device scans every 10s for nearby WiFi access points transmitting at 2.4 GHz. Using the signals strength information gathered from nearby access points, we construct a WiFi fingerprint as it was described in Section 4.1. The aggregated WiFi fingerprints, containing the normalised average signal strength values of access points over a window of 60s, are uploaded to the cloud. A cloud service is then responsible for estimating if two devices are co-located. Specifically, an adjusted Manhattan distance metric (as shown in Section 4.1) is calculated over the WiFi fingerprints of smartphones that are potentially co-located (i.e. contain at least one common access point in their set). Subsequent WiFi fingerprints are generated every 2.5 mins.

Deciding if two users are potentially participating in a social interaction according to proximity depends on two parameters:

the estimated distance between them, and the duration that they are co-located for. The selection of these parameters depends on the particular types of interactions that are targeted by the system. In this system, our objective is to capture significant social interactions that last for more than a few minutes. Specifically, we consider two devices to be co-located if the Manhattan distance is below a threshold of 0.8. Based on our preliminary experiments, the threshold was considered sufficient to discover co-location within less than 5m. Furthermore, if two users are co-located for more than a period of 5 mins, we consider this a potentially significant interaction. If these conditions are met, the cloud service triggers the co-located phones to initiate their sound sensing tasks.

5.2 Sound fingerprint

The preliminary analysis of audio signals showed that smartphones that are close to a social interaction can capture distinctive frequency patterns that can help distinguish the nearby social groups. In order to capture such patterns, we designed a technique that can capture a *"sound fingerprint"* that can represent the speech patterns detected over a time window of a few seconds. Our aim was to represent the sequence of sounds over that window as a fingerprint vector that can be easily compared with other fingerprints captured by nearby smartphones.

The sound sensing subsystem of the Next2Me system captures audio at a sampling rate of 16 KHz. This allows a Fast Fourier Transform (FFT) resolution of 8 kHz, but also provides a balance of higher quality signals. We use a window size of 2 secs, with a hamming window for calculating the FFT. The window of 2 secs was considered sufficiently large to allow more lenience with audio synchronisation across smartphones, considering that the on-board real-time clocks may not be perfectly synchronised. We extract the frequency bands between 300 Hz and 3,400 Hz which is the typical spectrum for human speech. Considering the results from the preliminary study, we can observe that frequency spectrums from devices around the same social interactions demonstrate high magnitudes around the same frequencies. Our aim is to use the significant frequencies in each sound sample as a way of comparing the sound patterns captured by different devices. As the sound capturing sensitivity varies across devices, we first need to reduce the variance on the sound spectrum that is produced. We apply a linearly weighted sliding average across the spectrum to smooth the results. Next, we sub-sample the smoothed spectrum to reduce the granularity. Specifically, we use a 30 Hz spectral window and calculate the average frequency magnitude for each window. The whole process produces a smoother frequency spectrum with 30 Hz granularity. From this spectrum, we define as *partial fingerprint* the set of the top n frequencies with the highest magnitude.

In order to improve the robustness of the fingerprint against ambient noise, we produce the *sound fingerprint* for part of a social interaction by combining multiple partial fingerprints as a time series of sets with the top n frequencies (see Figure 8):

$$S = \{P_1, P_2, \ldots, P_k\}, \text{where } P_i = \{f_1, f_2, \ldots, f_n\}$$

devices can be compared by using the Jaccard index over their partial fingerprints. The Jaccard index ($\frac{|A \cap B|}{|A \cup B|}$) measures the similarity of two sets by estimating the number of common frequencies

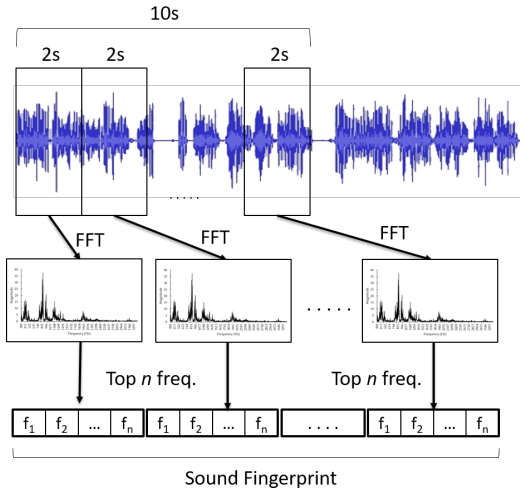


Figure 8: Generating a sound fingerprint representing 10s of captured audio.

over the total number of unique frequencies in the two sets. We define the similarity function for two sound fingerprints as the average Jaccard index of their partial fingerprints:

$$sim(S_a, S_b) = \frac{1}{k} \sum_{i=1}^{k} \left(\frac{|P_a^i \cap P_b^i|}{|P_a^i \cup P_b^i|} \right)$$

The output of the comparison of sound fingerprints gives us a metric that represents the proximity of people according to the sounds captured by their phones.

5.3 Community detection

The sound fingerprints captured by the smartphones are uploaded to the cloud. A cloud service estimates the similarity metric between sound fingerprints of all the co-located devices. This similarity metric is then used to produce a weighted graph that represents a social network of all the co-located devices. It is expected that smartphones of users participating in the same social group will have a higher similarity (i.e. weight) in the graph. Using the social graph, we extract communities by applying the Louvain community detection algorithm [2]. In order to overcome the resolution limit issue experienced in modularity based community detection, we use the resolution limit technique described in [11]. We experimentally chose a limit of 0.8 to allow smaller communities to be identified. The output of the community detection represents the output of the system, identifying the different groups interacting within close proximity of each other (Figure 9).

5.4 Fine-tuning parameters

In order to analyse and fine-tune the parameters of the system, we needed a scenario that involves a more complex setting than the preliminary studies. We conducted an additional study involving a social networking event (Experiment 3). We invited 7 participants (2 female, 5 male) to join a large meeting room and engage in a typical networking situation where they were asked to form smaller groups and freely discuss about their work (Figure 10). The

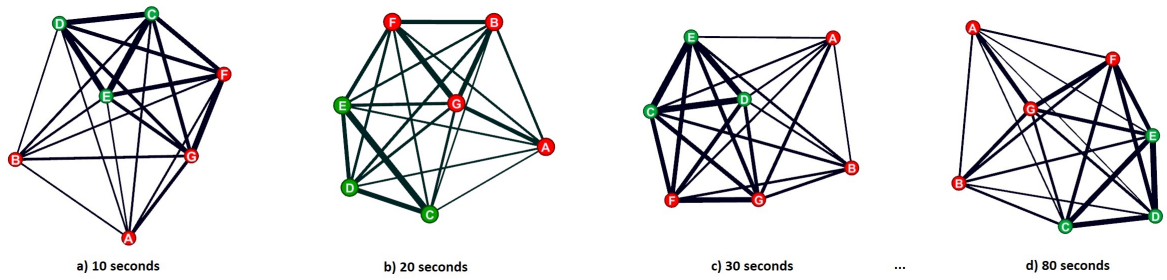


Figure 9: Community network graphs over time, at different time points since the start of the interaction (Experiment 2). Node colours reflect the grouping produced by the community detection algorithm.

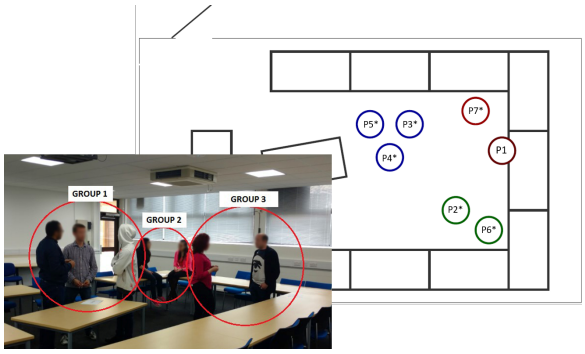


Figure 10: Example layout for the conference scenario during Stage 1

Stage	Groups		
1	(P1, P2, P3)	(P4, P5)	(P6, P7)
2	(P1, P2, P6, P7)	(P3, P4, P5)	–
3	(P2, P6)	(P1, P7)	(P3, P4, P5)

Table 1: The groupings of participants during the networking experiment. Participants changed the formed groups three times during the experiment. All participants had their phones in their pockets.

participants had the Next2Me app installed on their smartphone devices. All participants kept their smartphones in their pockets during the event. At regular intervals participants were asked to "mingle", changing the groups of people they talk with. Throughout the event, an observer kept track of the ground truth marking the actual groups that were formed. During the scenario, the groupings changed three times as shown in Table 1.

Using the dataset captured through this scenario, we attempted to fine-tune the parameters used for the sound fingerprint. Specifically, to identify the number of top frequencies selected for the partial fingerprints, and the total length of the sound fingerprint. In order to assess the quality of each configuration we calculated the number of nodes that were grouped with the majority of their correct social peers, against the number of nodes that were incorrectly placed in a group that did not involve their correct social peers. Throughout our analysis, we use the precision of the results: $\frac{C}{C+I}$ where C is the number of correctly grouped nodes, and I is the number of incorrectly grouped nodes. In these estimations, the notion of *false positive* and *false negative* are essentially the same.

In order to estimate the best parameters for the sound fingerprint, we selected a small sample of audio data captured in this experiment where there was definitely speech. Using a combination of numbers for the n top frequencies, and the total duration of the fingerprint, we achieved the best results when selecting the top 6 frequencies for each partial fingerprint and maintaining a fingerprint window of 10secs.

6 EVALUATION

In a real scenario, the Next2Me detection of social interactions does not necessarily need to run continuously throughout a co-location event. Instead, audio sensing can be used sparsely during a period to identify social groups. In our evaluation, we firstly attempted to estimate the average precision that can be achieved if sound fingerprinting is used only once during a social interaction using a 10 secs fingerprint. As it is shown in Figure 9, applying the sound fingerprinting technique at different intervals can have varying results. To estimate the average performance of the system, for each experiment we calculated the average performance for every 10 secs time window of the social interaction, using a 10 secs sliding window with 9 secs overlap.

For Experiment 2, two tables were positioned no more than 1m apart, phones were placed on the table, and two participants had additional phones in their pockets. We first estimated the average precision by including only the smartphones that were placed the desks, which resulted in 100% success rate (Table 2). This is a good result but somehow expected, considering that the experiment was performed in a quiet environment, and the smartphones were in the centre of conversations that were taking place. When combining the system with the smartphones placed in pockets, the average precision dropped to 88.3%. Exploring the results, we could see that the location of one pocket smartphones was quite close the second group, occasionally picking up stronger sound signals from the other table. Furthermore, the table itself acted as a barrier, blocking sound signals from the conversations reaching the pocket smartphones affecting the precision of the overall system.

In Experiment 3, where participants socialised within the same room, all smartphones were placed in participants' pockets. We ran our evaluation over the different stages where different groups were formed. The overall precision ranged from 74% to 91% (Table 2).

Experiment	Precision	Correct	Incorrect
Exp 2 - On table	1.00	66	0
Exp 2 - In pocket	0.88	99	13
Exp 3 - Stage 1	0.74	149	50
Exp 3 - Stage 2	0.91	238	21
Exp 3 - Stage 3	0.80	186	44

Table 2: Results for the two experiments involving social interactions of groups in close proximity.

Although these were encouraging results, they all relied on capturing a single sound fingerprint during a social interaction. Next, we explored how combining multiple sound fingerprints could improve the overall precision of the system.

6.1 Duty Cycling

We anticipate that precision on social interaction detection using sound fingerprints can be improved by combining multiple sound fingerprints captured over longer periods of time. There are cases where a randomly selected 10sec sound fingerprint may capture a situation where the actual social groups are not correctly mapped. Allowing more than one sound fingerprints to be inspected at different time points, offers more chances to discover the correct social group mapping. By applying a duty cycling scheme, there are ways to potentially improve the overall precision of the system while keeping the energy cost relatively low. Specifically, we explored the effects of a fixed-length duty cycled sensing, where a fixed number of sound fingerprints can be captured during a potential social interaction. When combining multiple fingerprints, we wanted to explore what is the number of consecutive sound fingerprints that we should use to improve precision, and how the length of the sleeping windows between them would affect the overall result.

Combining multiple fingerprints for the detection of social groups would involve modifications in the way that the weights in the social network graph are calculated. Specifically, when the social graphs are formed, the weight between two nodes includes the average fingerprint similarity over the number of sound fingerprints:

$$W_{a,b} = \frac{1}{k}\sum_{i=1}^{k} sim(S_{i,a}, S_{i,b})$$

where, $W_{a,b}$ is the weight between participants a and b, k is the number of sound fingerprints involved, and $S_{i,a}$ is the i-th sound fingerprint for participant a. After weights are estimated by combining multiple fingerprints, the same community detection algorithm is used to estimate the social groups that are formed.

Using the datasets from Experiment 3 (networking event), we tested the performance of the system when using 2 or 3 sound fingerprints, with a varying sleeping windows between them; ranging from continuous (no sleeping) to fingerprints captured with a 60sec gap between them. We calculated the performance of the system, with the duty-cycled scheme being applied at any time during the whole experiment and estimated the average precision of the results (Figure 11). The results show that using more than one sound fingerprint improves the overall precision, while the combination of three fingerprints reduces the variance that we observed in precision. Generally, we see that combining multiple fingerprints with a sleeping window of 40sec offers the best results. Specifically, a duty cycling scheme of 3 sound fingerprints with 40sec sleeping shows an average precision of 92% and a combination of 2 sound fingerprints with 40sec shows an average precision of 89%. Following this, we conclude that for a setup of 3 samples/40sec sleeping is appropriate for the high precision, while a 2 samples/40sec sleeping scheme offers a good balance of energy cost and precision.

6.2 Coffee Shop scenario

As a final step in the evaluation of the Next2Me system, we performed a *"real-world"* deployment where participants were involved in social interaction within a busy coffee shop. Six participants were invited to install the Next2Me app onto their phones and to meet in a busy coffee shop and socialise, forming two separate social groups and sitting at nearby tables (Figure 12). The event took place during a busy time where a number of other people were in the coffee shop, and engaged in conversations. The setup was selected to ensure the environment involved ambient noise of other people talking to each other. During the event, participants placed their phones on the table while two participants had a smartphone placed in their pocket. Note that the table in this scenario had a relatively lower height than the tables involved in previous scenarios.

The system was configured to perform WiFi scans to detect co-location, and trigger sound fingerprinting when participants were co-located for more than 5mins. The overall experiment lasted for 20 mins. We analysed the performance of the system using 3 sound fingerprints captured with a 40sec sleeping window between them. Using the devices involved in the scenario, the system achieves an average precision of 88%. When the pocket phones are not included in the estimation, the precision raises to 99.1%. This shows that phones situated without physical obstructions and in an open environment will perform well, and smartphones in a pocket will be clustered into communities with less precision due to the frequency-filtering effect of the pocket material.

7 ENERGY CONSIDERATIONS

The design of the Next2Me system relies heavily on sensing modalities that can have a significant impact on the battery life of the participants' smartphones. In this section we analyse the energy cost implications of using Next2Me. In our analysis we attempt to establish the average cost in the form of electric charge (measured in mAh) consumed by the Next2Me during a day. This estimation will allow us to consider the impact that the system would have on the battery life of common smartphones, with battery capacities in the rage of 2,800mAh (Samsung S5) to 3,220mAh (Nexus 6).

The WiFi fingerprinting subsystem relies on the periodic WiFi scanning to discover near-by WiFi access points. If we consider that the electric charge consumed during a WiFi scan is E_w, a WiFi fingerprint is generated using 6 scans, and a fingerprint is produced every s_w seconds, the overall cost of continuously running the WiFi scanning subsystem for a whole day is:

$$W_{total} = \frac{86,400}{s_w}(6 \cdot E_w + N_w) \quad (1)$$

where 86,400 is the total number of seconds in a day and N_w is the average energy cost of uploading data to the cloud.

When a co-location incident is captured by WiFi scanning, and it lasts for more than 5mins, the sound fingerprinting subsystem

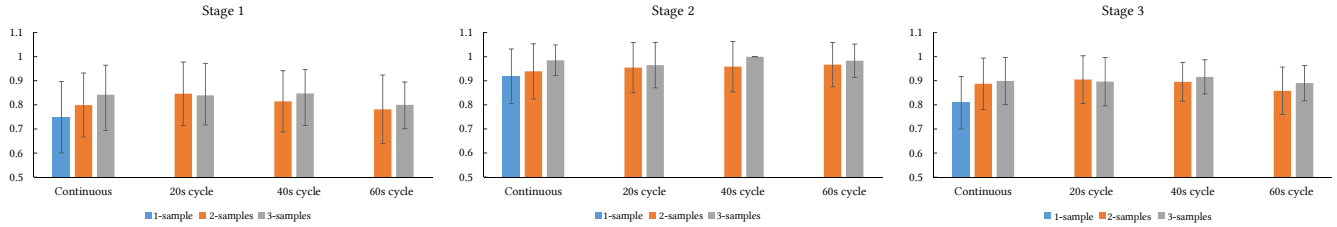


Figure 11: Effect of duty-cycling window size and number of sound sensing samples in the overall precision.

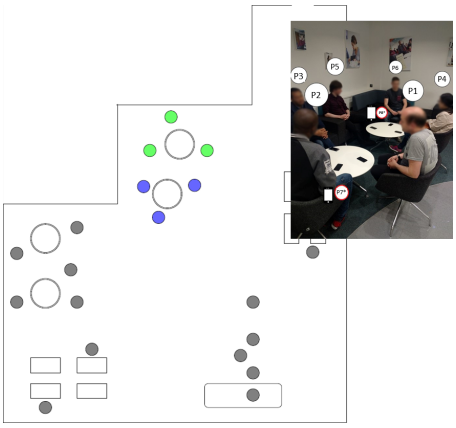


Figure 12: Experiment setup for the Café social interaction

is triggered. Each sound fingerprinting will involve 10sec of audio recording, and involves a CPU processing cost to perform a FFT over the sample. Subsequent sound fingerprints will then be uploaded to the cloud for comparison. We can model the additional energy for the sound fingerprinting subsystem caused by a single social interaction as:

$$S_{int} = 3 \cdot (E_{sense} + E_{FFT}) + N_s \quad (2)$$

where S_{int} is the cost for a single social interaction, E_{sense} and E_{FFT} are the costs for capturing audio for a sound fingerprint, and performing the FFT respectively. The data from 3 sound fingerprints would typically be uploaded to the cloud, incurring an additional N_s cost for network communication. From Equations (1) and (2) we can estimate the average energy for an individual who has on average k significant social interactions during their day:

$$E_{total} = W_{total} + k \cdot S_{int} \quad (3)$$

In order to estimate the average energy of the Next2Me system, we performed a number of lab measurements to estimate the energy consumption. We used the Monsoon Power Meter setup to intercept the current drawn from the battery of a phone. We run experiments using the Samsung Galaxy J3 smartphone. A base line current when a phone is not performing any activities was estimated to be 9.16mA (2.27mA in airplane mode). When the phone was set to perform WiFi scanning, the average current during the scanning, without the baseline, was estimated to be 93.24mA. Each scan lasted for approximately 0.78s which results in an electric charge of E_w = 72.73mAs. The average cost of data upload can vary significantly depending on the network infrastructure and external conditions. In order to estimate the impact of data upload using WiFi we use the energy cost per KB of 5mJ as it is estimated in [19] which results in consumed energy charge of N_w = 1.3mAs. In the final deployment of the system we set the WiFi scanning subsystem to perform a scan once very 2.5 mins (which would enable the detection of 5min colocation instances). From equation 1 we can then estimate that in case of a user who does not have any significant interactions during the day, the overall energy cost is:

$$W_{total} = \frac{86,400}{150}(436.38 + 1.3) = 252,138\text{mAs} = 70.03\text{mAh}$$

For a phone with a battery of 2,800mAh this would be 2.5% of the battery's capacity.

In order to estimate the impact of sound fingerprinting, we calculated the average energy cost of audio sampling, and performing a FFT over a sound sample. The average current for audio sampling without the baseline was estimated to be 32.84mA. For capturing an audio sample of 10s this would result in consuming an electric charge of $E_sense = 328.41mAs$. When performing a FFT over a 2s audio sample the average current (excluding the cost of audio sensing and baseline) is 56.14mA and the duration is 105ms. Therefore the cost of performing 5 FFTs for a sound sample of 10s would be $E_{FFT} = 29.47mAs$. From equation 2 we can estimate the additional cost of detecting a single social interaction as:

$$S_{int} = 3 \cdot (328.41 + 29.47) + 1.3 = 1074.94\text{mAs} = 0.29\text{mAh}$$

Assuming a case of a user who has about 20 significant interactions during the day, the additional energy capacity consumed by the system would be $E_{total} = 70.03 + 20 \cdot 0.29 = 75.83$mAh. This results to 2.7% of the battery's capacity. These results demonstrate that Next2Me has a very small impact on the smartphone's battery life and would be appropriate for continuous sensing.

8 DISCUSSION

In the design of Next2Me we focus on the detection of significant social interactions, that last for more than 5-mins. The technique is robust for such social events, but would not be appropriate for capturing short time, serendipitous interactions that last for only a few seconds. Although such short interactions are beyond the scope of this work, the proposed technique could potentially be adapted with more aggressive use of sound sensing to capture such short events. However, such approach would increase the power cost of sound sensing, and would require further exploration in adaptive sensing approaches to mitigate energy issues. Furthermore, the physical environment can have a significant impact on

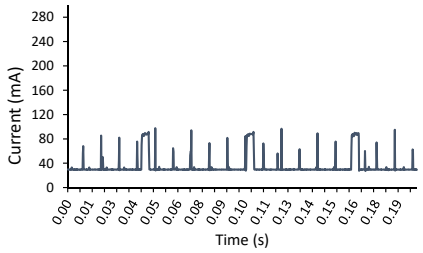


Figure 13: Current over time for the continuous recording of audio at 16kHz sampling rate

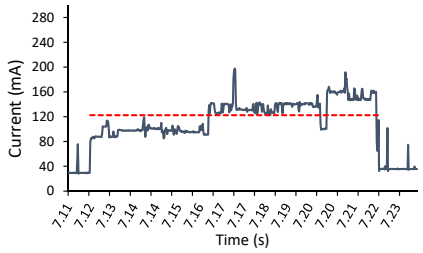


Figure 14: Current over time for one FFT of a 2-second audio sample (includes audio sensing and baseline).

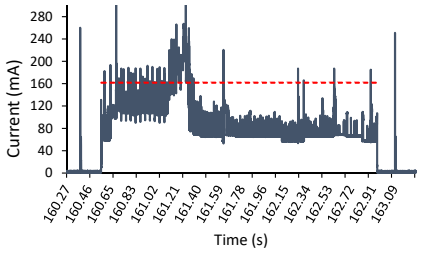


Figure 15: Current over time for one Wi-Fi scan

the performance of Next2Me. In this work we demonstrate that the proposed system is robust against smartphone placement in participants' pockets, but further investigation would be necessary to fully explore the impact of the environment, such as higher/lower ceilings or significant acoustic echo.

9 CONCLUSION

We developed a system that can use WiFi and Audio signals captured by smartphone devices to detect social interaction. The proposed system detects the social interactions between people in various environments by capturing their co-location using WiFi, and accurately distinguishing social encounters through the capture and analysis of *"sound fingerprints"*. Our technique can achieve a high precision at low energy overhead, regardless of sound blocking material such as pockets, and can be robust to background noise.

Acknowledgements

We thank the participants who helped collect data in the social settings for all of the experiments, as well as James Alexander Lee & Hamed Alsufyani for their continued support and assistance.

REFERENCES

[1] A. Barrat, C. Cattuto, V. Colizza, J. Pinton, W. Van den Broeck, and A. Vespignani. 2008. High resolution dynamical mapping of social interactions with active RFID. *arXiv preprint arXiv:0811.4170* (2008).

[2] V. D. Blondel, J. Guillaume, R. Lambiotte, and E. Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment* 2008, 10 (2008), P10008.

[3] C. Brown, C. Efstratiou, I. Leontiadis, D. Quercia, C. Mascolo, J. Scott, and P. Key. 2014. The architecture of innovation: Tracking face-to-face interactions with ubicomp technologies. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 811–822.

[4] A. Clifford and J. Reiss. 2011. Proximity effect detection for directional microphones. In *Audio Engineering Society Convention 131*. Audio Engineering Society.

[5] D. Datta, P. P. Datta, and K. K. Majumdar. 2015. Role of social interaction on quality of life. *National Journal of Medical Research* 5, 4 (2015), 290–292.

[6] P. Georgiev, N. D. Lane, K. K. Rachuri, and C. Mascolo. 2014. Dsp. ear: Leveraging co-processor support for continuous audio sensing on smartphones. In *Proceedings of the 12th ACM Conference on Embedded Network Sensor Systems*. ACM, 295–309.

[7] H. Hong, C. Luo, and M. C. Chan. 2016. SocialProbe: Understanding Social Interaction Through Passive WiFi Monitoring. In *Proceedings of the 13th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*. ACM, 94–103. https://doi.org/10.1145/2994374.2994387

[8] X. Hu, J. Shang, F. Gu, and Q. Han. 2015. Improving Wi-Fi Indoor Positioning via AP Sets Similarity and Semi-Supervised Affinity Propagation Clustering. *International Journal of Distributed Sensor Networks* 11, 1 (01/14; 2017/02 2015), 109642. https://doi.org/10.1155/2015/109642 doi: 10.1155/2015/109642; 28.

[9] M. B. Kjægaard and C. V. Munk. 2008. Hyperbolic location fingerprinting: A calibration-free solution for handling differences in signal strength (concise contribution). In *Pervasive Computing and Communications, 2008. PerCom 2008. Sixth Annual IEEE International Conference on*. IEEE, 110–116.

[10] M. B. Kjærgaard, M. Wirz, D. Roggen, and G. Tröster. 2012. Mobile sensing of pedestrian flocks in indoor environments using wifi signals. In *Pervasive Computing and Communications (PerCom), 2012 IEEE International Conference on*. IEEE, 95–102.

[11] R. Lambiotte, J. Delvenne, and M. Barahona. 2008. Laplacian dynamics and multiscale modular structure in networks. *arXiv preprint arXiv:0812.1770* (2008).

[12] Y. Lee, J. Song, C. Min, C. Hwang, J. Lee, I. Hwang, Y. Ju, C. Yoo, M. Moon, and U. Lee. 2013. SocioPhone: Everyday Face-to-Face Interaction Monitoring Platform Using Multi-Phone Sensor Fusion. *Proceeding of the 11th annual international conference on Mobile systems, applications, and services - MobiSys '13* (2013). https://doi.org/10.1145/2462456.2465426

[13] S. Liu, Y. Jiang, and A. Striegel. 2014. Face-to-face proximity estimation using bluetooth on smartphones. *IEEE Transactions on Mobile Computing* 13, 4 (2014), 811–823.

[14] Z. Liu, Z. Zhang, L. He, and P Chou. 2007. Energy-based sound source localization and gain normalization for ad hoc microphone arrays. In *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, Vol. 2. IEEE, II–761.

[15] N. Palaghias, S. A. Hoseinitabatabaei, M. Nati, A. Gluhak, and K. Moessner. 2015. Accurate detection of real-world social interactions with smartphones. In *Communications (ICC), 2015 IEEE International Conference*. IEEE, 579–585.

[16] J. E Perry-Smith and C. E Shalley. 2003. The social side of creativity: A static and dynamic social network perspective. *Academy of management review* 28, 1 (2003), 89–106.

[17] M. R. Carillo, E. Papagni, and F. Capitanio. 2008. Effects of social interactions on scientists' productivity. *International Journal of Manpower* 29, 3 (2008), 263–279.

[18] K. K. Rachuri, M. Musolesi, C. Mascolo, P. J. Rentfrow, C. Longworth, and A. Aucinas. 2010. EmotionSense: a mobile phones based adaptive platform for experimental social psychology research. In *Proceedings of the 12th ACM international conference on Ubiquitous computing*. ACM, 281–290.

[19] A. Rice and S. Hay. 2010. Decomposing power measurements for mobile devices. In *Pervasive Computing and Communications (PerCom), 2010 IEEE International Conference on*. IEEE, 70–78.

[20] I. Rishabh, D. Kimber, and J. Adcock. 2012. Indoor localization using controlled ambient sounds. In *Indoor Positioning and Indoor Navigation (IPIN), 2012 International Conference*. IEEE, 1–10.

[21] T. E Seeman. 1996. Social ties and health: The benefits of social integration. *Annals of epidemiology* 6, 5 (1996), 442–451.

[22] B. Thiel, K. Kloch, and P. Lukowicz. 2012. Sound-based proximity detection with mobile phones. In *Proceedings of the Third International Workshop on Sensing Applications on Mobile Phones*. ACM, 4.

[23] G. Vanderhulst, A. Mashhadi, M. Dashti, and Fahim Kawsar. 2015. Detecting human encounters from WiFi radio signals. In *Proceedings of the 14th International Conference on Mobile and Ubiquitous Multimedia*. ACM, 97–108.

[24] C. Xu, S. Li, G. Liu, Y. Zhang, E. Miluzzo, Y. Chen, J. Li, and B. Firner. 2013. Crowd++: Unsupervised speaker count with smartphones. In *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*. ACM, 43–52.