SocialCount
Detecting Social Interactions on Mobile Devices

Isadora Vasconcellos e Souza[1], João Carlos Damasceno Lima[1,2], Benhur de Oliveira Stein[1,2]
and Cristiano Cortez da Rocha[3]

[1]*Programa de Pós-Graduação em Informática, Universidade Federal de Santa Maria (UFSM), Santa Maria, RS, Brazil*

[2]*Departamento de Linguagens e Sistemas de Computação, Universidade Federal de Santa Maria (UFSM), Santa Maria, Brazil*

[3]*Centro de Informática e Automação do Estado de Santa Catarina (CIASC), Florianópolis, SC, Brazil*

Keywords: Social Interactions, Context-aware, Mobile Applications.

Abstract: With mobile devices increasingly powerful and accessible to the majority of the population, applications have begun to become increasingly intelligent, customizable and adaptable to users' needs. To do this, context-aware applications are developed. In this work, we create an approach to infer social interactions through the identification of the user's voice and to recognize their social context. Data from the social context of the user has been useful in many real-life situations, such as identifying and controlling infectious disease epidemics.

1 INTRODUCTION

In recent years, mobile devices have started to be part of everyday life for most people. In the beginning users acquired the devices just to make calls, but over the years it has become a tool to optimize and help users in various types of tasks. In this way, the applications had the need to obtain data of human factors such as activity, social interactions, health, etc. to customize processes. Thus, mobile computing began to develop context-sensitive applications.

Context-aware systems offer entirely new opportunities for application developers and end-users, gathering context data and adapting the behavior of systems accordingly. Especially in combination with mobile devices, these mechanisms are of high value and are used to greatly increase usability. (Baldauf et al., 2007)

Mobile devices are used to interact with other users through calls, messages or social networks. In addition to being mostly close to the user. As such, they are a great tool for detecting social interactions, both face-to-face and virtual. The SocialCount application proposed in this paper makes the inference of social interactions face-to-face. These data can be used in several areas, such as: marketing (word-of-mouth), business (enterprise community detection) and health (infectious disease control).

In section 2 we present the concepts of context-aware computing, social-aware computing, ubiquitous computing, social context and social interaction and how these concepts relate. Section 3 discusses related works. Section 4 presents the SocialCount methodology. Section 5 presents a case of use of infectious disease control with the use of social interaction data. Finally, in section 6 is the conclusion.

2 CONTEXT-AWARE COMPUTING

The Context-aware Computing emerged to address the challenges of mobile computing, where applications have begun to explore the changing environment in which they are run (Schilit et al., 1994). It is desirable that mobile device applications and services react to their current location, time, and other attributes of the environment, and adapt their behavior according to changing circumstances as context data can change rapidly (Baldauf et al., 2007). Such context-aware software adapts their functions, contents, and interfaces according to the user's current situation with less distraction of the users. Thus, aim at increasing usability and effectiveness by taking environmental context into account (Temdee and Prasad, 2017).

The word "context" is very comprehensive, it is possible to find different definitions proposed by dif-

Vasconcellos e Souza, I., Damasceno Lima, J., de Oliveira Stein, B. and Cortez da Rocha, C.
SocialCount.
DOI: 10.5220/0006696605110518
In *Proceedings of the 20th International Conference on Enterprise Information Systems (ICEIS 2018)*, pages 511-518
ISBN: 978-989-758-298-1

ferent authors. "Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between an user and an application, including the user and applications themselves" (Dey, 2001). Three important aspects of context are: where you are, who you are with, and what resources are nearby. This includes lighting, noise level, network connectivity, communication costs, communication bandwidth and even social situation (Schilit et al., 1994).

The context can be divided into five subcategories: Environmental context, Personal context, Social context, Task context and Spatio-temporal context (Kofod-Petersen and Mikalsen, 2005). These sub-categories are defined as (Cassens and Kofod-Petersen, 2006):

- Environmental context: This part captures the user's surrounding, such as things, services, people, and information accessed by the user.
- Personal context: This part describes the mental and physical information about the user, such as mood, expertise and disabilities.
- Social context: This describes the social aspects of the user, such as information about the different roles an user can assume.
- Task context: the task context describes what the user is doing, it can describe the user's goals, tasks and activities.
- Spatio-temporal context: This type of context is concerned with attributes like: time, location and the community present.

One of the key features of a context-aware application is not to be intrusive. In other words, the application needs to detect user context information without any kind of intervention that can change the current context. If the user is working and the application alerts him to make a decision or provides some information, the user's status can be changed from "working" to "using mobile", in this way the application changes the user's current task context. Therefore, context-aware computing uses the notions of ubiquitous computing.

Ubiquitous computing can also be called synonymously the generalized computation (Temdee and Prasad, 2017). However, there are some differences. The concept of pervasive computing implies that the computer is embedded in the environment invisibly to the user. The computer has the ability to obtain information from the environment in which it is shipped and use it to dynamically build computer models, such as controlling, configuring and tuning the application to better meet the needs of the device or user. Ubiquitous computing comes from the need to integrate mobility with the functionality of pervasive computing. In other words, any moving computing device can dynamically construct computational models of the environments in which we move and configure their services depending on the need (de Araujo, 2003).

Since the beginning of social networks, the researchers are using the sensing data to understand human behavior, mobility, and activity, and ultimately helping to solve social problems (Yu et al., 2012). For this reason, context-aware computing has generated a new emerging research topic in computer science called social-aware. "Humans, however, are social beings. Hence, the notion of social context-awareness (in short social awareness) extends the vision of context-aware computing" (Kabir et al., 2014).

Socially aware applications are services that exploiting any information that describes the social context of the user, like social relations, social interactions, or social situations and embody the ability to trace and model ongoing social processes, structures, and behavioral patterns (Ferscha, 2012).

Social context is defined as a set of information derived from direct or indirect interactions among people in both virtual and physical world. Direct interaction contains face to face conversation, video conferencing, etc.. Indirect interaction includes colocating for a period of time, joining the same event, etc. (Kabir et al., 2014). The SocialCount recognizes the user's social interactions to get information about the user's social context.

According to the sociological approach social context is the way that people can relate easily, including the culture in which the individual lives and has been educated and the people and institutions with whom he interacts (Kolvenbach et al., 2004) (Carter, 2013). Some authors add places and activities to the concept (Adams et al., 2008). Therefore, according to the definition of Schilit et al., the social context addresses two of the main aspects of the context: where are you and who you are with.

Biamino proposed a more specific interpretation that would meet the point of view of pervasive and ubiquitous computing. They suggest that the social context can be represented through networks. "In our vision social contexts are more similar to social aggregations or social groups, identified as a number of nodes in a given location, linked by some kind of ties (relations) that determine their nature" (Biamino, 2011). The authors define a social context as a 3-tuple that, describe the network:

$$cxt = < Size, Density, TypeofTies > \quad (1)$$

In this paper the definitions of *Size*, *Density* and *Type of Ties* were adapted from the work of Biamino. The *Size* represents the number of nodes in a defined location:

- Small (n $\leq$ 5): a network with a small number of nodes.
- Private (5 < n $\leq$ 20): a network with a few nodes;
- Open (20 < n $\leq$ 50): a relatively large network;
- Wide (n > 50): a network with a very large number of nodes;

Density represents the number of connections between the nodes:

- Clique: a fully connected graph;
- Easy: a graph with easy to close triangles;
- Hard: a graph with many isolated nodes and hard to close triangles;

Type of Ties is defined by the main type of relations between the nodes of the network:

- Unknown: no relation exists between two nodes;
- Acquaintance: two nodes are not close friends, but they interact with each other;
- Friends: two nodes with a friendship-kind of relation;

SocialCount uses data collected from social interactions to generate social graphs and classify the context criteria according to 3-tuple. In the next section are presented and discussed related works that also collect social interactions.

3 RELATED WORK AND DISCUSSION

The related works were selected with the purpose of presenting the state of the art and the most used methodologies of detection of social interactions in mobile devices. The main characteristics raised for detecting interactions are: interpersonal distance, user location, relative position and conversation activity.

The most common approach used by researchers to recognize social interactions between two individuals is the Bluetooth ID search (BTID) or Wi-Fi service ID (SSID) of nearby devices. All devices/people found are classified as social interactions. This method was used in CenceMe (Miluzzo et al., 2007), SoundSense (Lu et al., 2009), E-Shadow (Teng et al., 2014), PMSN (Zhang et al., 2012), among others. This approach is simple and does not require specialized hardware and sensors, but the accuracy is limited by the range of Bluetooth (about 10 meters) and Wi-Fi (approximately 35 meters for indoor environments).

DARSIS (Palaghias et al., 2015) was developed to quantify social interactions in real time. The relative orientation of the users was used to obtain face direction and interpersonal distance. The proximity between the participants of the interaction is calculated through the RSSI samples of the user device's Bluetooth that are trained by learning machine (MultiBoostAB with J48). Samples are taken from three device position combinations: screen to screen, screen to back and back to back. The proximity is classified as public area, social zone, personal zone and intimate zone. The relative orientation of the user is known through uDirect (Hoseinitabatabaei et al., 2014) which identifies the relative orientation between the Earth's coordinates and the user's locomotion and predicts the direction of the face without requiring a fixed position of the device.

In Multi-modal Mobile Sensing of Social Interactions (Matic et al., 2012) they used a set of methods for the sensing of the interactions: the interpersonal distance, the relative position of the user, the direction of the face and the verification of speech activity. The interpersonal distance between the devices is captured in RSSI, where one device works as a Wi-Fi access point (Hot Spot) and another as a Wi-Fi client. The relative position is calculated by the position of the torso in relation to the coordinates of the Earth, always considering the same position of the mobile device. The speech activity is detected by an accelerometer installed on the user's chest. The device configured as a Wi-Fi access point is characterized as an intrusive process because the user generally does not use his mobile device for this purpose. Likewise the accelerometer on the chest because is not a commonly used device.

SCAN (Social Contex-Aware smartphone Notification system) (Kim and Lee, 2017) detects the user's social context and blocks smartphone notifications so as not to distract the user while he or she is interacting. The system sets breakpoints to release notifications according to the following criteria: silence (when there is no conversation for 5 seconds or more), movement (when a person in the group leaves the table), user alone (when the person is alone waiting for friends) and use (when the other person participating in the interaction is using the smartphone). Social interactions are known through identification of close people and conversation. SCAN periodically searches for BLE beacons to detect the presence of other people and announce their own presence, the BLE bea-

Table 1: Comparative table of relation works and SocialCount.

	Intrusive	Detection from speech sound	Detection from who is speaking	Approach of interaction detection
CenceMe, SoundSense, E-Shadow, PMSN	No	No	No	Interpersonal distance (ID Bluetooth e Wi-Fi)
DARSIS	No	No	No	Interpersonal distance (RSSI Bluetooth), user relative orientation and face direction
Multi-model	Yes	Yes	No	Interpersonal distance (RSSI Bluetooth), user relative orientation and detection from speech sound
SCAN	No	Yes	No	Interpersonal distance (BLE beacons) and detection from speech sound
SocioGlass	Yes	No	No	Image detection with Google Glass
SocialCount	No	Yes	Yes	Interpersonal distance (ID Bluetooth), detection from speech sound and detection from who is speaking

cons have been chosen for deployment because they do not require pairing and connection actions, and have a low power consumption. For the detection of conversation was used the algorithm YIN (De Cheveigné and Kawahara, 2002) that estimates the fundamental speech frequency and identifies the human voice.

SocioGlass (Xu et al., 2016) to promote additional information about the people who the user is interacting. There are 28 biographical information items that are classified into 6 groups: work, personal, education, social, leisure and family. The system uses Google Glass and an Android application that communicate via Bluetooth. Interactions are detected through facial recognition, Google Glass is responsible for providing the image of the individual who is participating in the interaction, the application receives the image, performs the processing and searches for a combination in the local database. When you do the recognition, the information related to the person in question is displayed on the Google Glass screen. The authors also implemented a smartphone-only version, where the face image is captured by the device's camera and the information is displayed on the mobile screen.

To analyze the related works, we insert them into a bus stop scenario. In this case, there are many people waiting for buses, some people are talking, others are quiet looking toward the cars. The Table 1 presents a comparative board between related works and SocialCount.

The works that only approach the interpersonal distance to consider an interaction are submitted to have a low accuracy, mainly in situations of the real life. According to the scenario, many people are physically close and do not interact with each other. In this case several interactions would be considered wrongly. The good thing about this methodology is that Bluetooth and Wi-Fi technology are compatible with most smartphones available in the market.

DARSIS obtained a good accuracy in verifying interpersonal distance. In addition, they used the direction of the face to identify the interactions, which is a good approach considering that a person tends to direct the face to the other person when communicating. However, they do not consider the conversation. Considering the scenario, this can easily lead to mistakes, where two people may be standing next to each other and looking in opposite directions.

Multi-modal and SCAN consider distance and conversation. The Multi-modal uses an intrusive approach to the verification of the conversation, which damages the naturalness of the user's daily actions. SCAN uses the YIN algorithm for the same purpose, which identifies the human voice without the need for external resources to the smartphone. There may still be errors in the scene, as people around the user may be talking.

SocioGlass used images captured with Google Glass. This can be considered an intrusive procedure, since few users own the device and use it regularly. The authors have made a version that works only on the smartphone, but users need to focus the camera on the person's face, which detracts from the usability of the application.

To solve the problems mentioned in the scenario, SocialCount uses a set of approaches. Interpersonal distance is implemented through Bluetooth, the detection of the conversation by the YIN algorithm and the detection from who is speaking. This last question is fundamental for the correct consideration of interactions. In the next section the SocialCount methodology is described in detail.

4 SocialCount

SocialCount is a mobile application developed for the Android platform. Its main purpose is to detect the user's social interactions and provide data that describes the social context. The interactions considered by the application are only face-to-face, that is, interactions mediated by a means of communication are not considered.

For inference of social interactions, SocialCount detects: human voice, who is speaking, location, and nearby devices. Based on related work, the differential of SocialCount is the use of the recognition of who is speaking for the inference of social interactions. During the conversation, the application checks whether the speaker is the user or someone near him. The recognition was developed with widely known methodologies, and the focus of this work is not to elaborate a new method.

Figure 1 shows how SocialCount works. The application is responsible for recognizing human voice in the environment, recording audio, locating nearby people and identifying the current location. The server recognizes who is speaking and stores the data in the database.

The application remains listening for the presence of human voice in the environment. When SocialCount detects voice, it records about 8 seconds and sends the audio to the server. The YIN (De Cheveigné and Kawahara, 2002) algorithm is used to identify human voice in the environment. The algorithm is developed by the TarsosDSP (Six et al., 2014) library, which performs real-time audio processing.

The server verifies that the user is participating in the conversation through voice prints previously stored in the database. Voice prints are a set of audios that contain speech frequency. To record the audios were elaborated phrases that contained the consonant phonemes of the native language in several vowel contexts.

The verification is developed with the Recognito (Crickx, 2014), which is a library that performs text-independent recognition of speakers in Java. The library generates an universal template with all stored voice prints. Each audio input for checking who is speaking Recognito computes the relative distance using the variables: identified voice print (VPI), unknown voice print (VPU), and universal (UM) model. VPI represents the voice prints that have already been identified, VPU is the audio input and UM is the universal model that represents the average of all stored voice prints. "If you put them on a line, you can calculate the distance between IVP/UVP and the distance between UVP/UM. Based on those numbers, you can tell how relatively close the unknown voice print is to the identified one. The UM acts as a max distance value" (Crickx, 2014).

SocialCount can classify an interaction in two ways: participation and monitoring. In participation, the server identifies who is speaking is the user, so he is participating in the interaction. In monitoring, the server identifies who is talking is not the user, it is someone who was close to the user's device. Therefore, the user may not be participating in the interaction. Data from interactions stored as monitoring are only used to increase the accuracy of the inference of the interactions stored as participation.

After find who is talking, the server sends a response to the application. The application then searches for nearby devices. The Bluetooth ID of nearby devices is used as additional information to increase the inference accuracy of users who are participating in the interaction. If the user chooses not to turn Bluetooth on to save battery, SocialCount continues to function normally.

Finally, SocialCount detects the location of the interaction and sends all the data to the server to store. The stored data are used to generate social graphs. A social graph can represent the interactions between users of SocialCount at a particular location over a period of time or the interactions performed by a particular user. The Figure 2 presents social graphs in two different environments: a research lab and a restaurant.

To classify the Type of Ties (ToT), we use the equation proposed by Palaghias (2016) to calculate the confidence of each social relation between a pair of users and the average of interactions performed:

$$P(r) = \frac{Q(r)}{N} \tag{2}$$

where $Q(r)$ is the number of inferences of interactions that are related to social relation r and N is the total number of social interactions inferences. In order to adapt the classification according to the social characteristics of each user, we calculated the average number of interactions performed per node according to the equation:

$$M(n) = \frac{N}{T \cdot N} \tag{3}$$

where T is the total number of nodes that have interacted with the current node n. Then we classify ToT as follows:

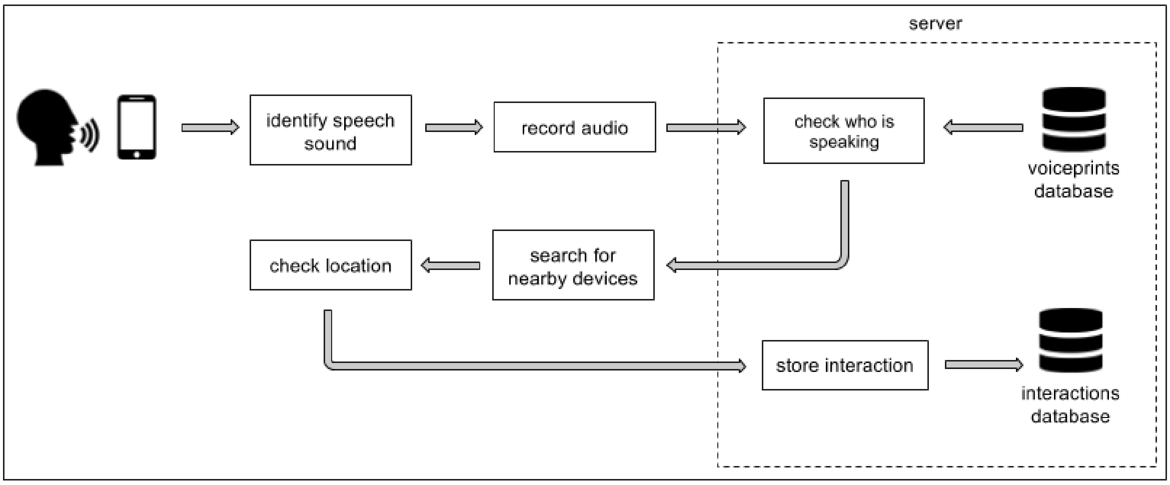

Figure 1: SocialCount flowchart.

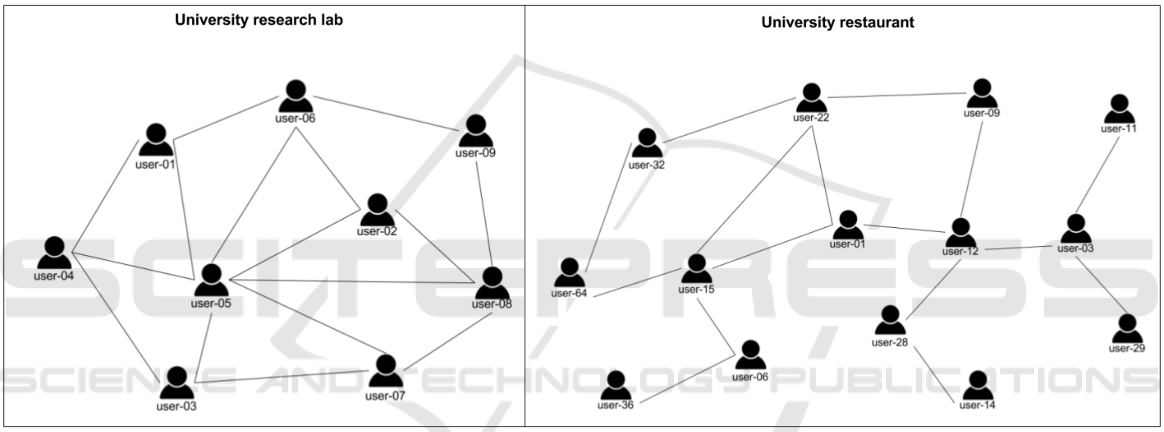

Figure 2: Examples of social graphs: (a) Social graph generated in a research lab, (b) Social graph generated in a university restaurant.

$$ToT = \begin{cases} P(r) = 0, & ToT = Unknown \\ P(r) < M(n), & ToT = Acquaintance \\ P(r) \geq M(n), & ToT = Friends \end{cases} \quad (4)$$

The social graph of the research laboratory presented in the Figure 2 is defined by:

$Researchlab =< Private, Easy, Friends >$.

It is private because it has 9 nodes, Easy because it is possible to close triangles easily and Friends because most social relations have $P(r) \geq M(n)$. And the graph of University restaurant is defined by:

$Restaurant =< Private, Hard, Acquaintance >$.

It is private because it has 14 nodes, Hard because it is difficult to close triangles and Acquaintance because most social relations have $P(r) < M(n)$.

5 USE CASE

The purpose of this section is to better address the benefits of bringing social awareness to mobile devices. We present a brief use case in which we describe how the inference of social interactions can help in real life events such as in combating the spread of infectious diseases.

Human contact is the most important factor in the transmission of infectious diseases (Clayton and Hills, 1993). Many diseases spread to human populations through contact between infectious individuals (people carrying the disease) and susceptible individuals (people who do not yet have the disease, but can get it) (Newman, 2002). These contacts generate networks called contact networks, which are networks of interaction through which diseases spread and determine whether and when individuals become infected and thus who can serve as an early and accurate surveillance sensor (Herrera et al., 2016).

Epidemiological models are based on the SIR model. A SIR model computes the theoretical number of people infected with a contagious illness in a closed population over time. The name of this class of models derives from the fact that they involve coupled equations relating the number of susceptible people S(t), number of people infected I(t), and number of people who have recovered R(t) (Weisstein, 2017). Based on this model, it is possible to determine if an epidemic is increasing or decreasing.

$$S + I + R = N \tag{5}$$

Traditionally, public health is monitored through research and the aggregation of statistics obtained from health care providers. These methods are expensive, slow, and can be biased. An infected person is only recognized after a doctor sends the necessary information to the appropriate health agency. Affected people who do not seek treatment, or do not respond to research, are virtually invisible to traditional methods.

Social graphs can act as networks of contacts to determine who are susceptible people (S) through the people with whom a contaminated user interacted. Figure 3 presents the social graph of "user-01" based on the contexts of Figure 2.

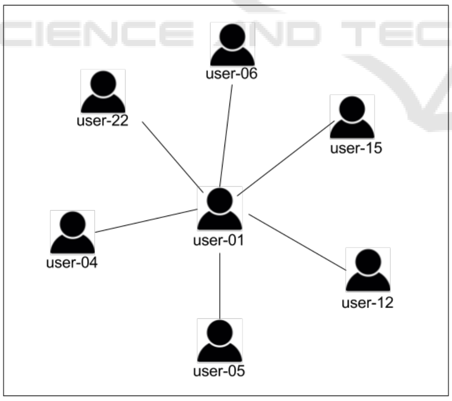

Figure 3: Social graph of "user-01".

If user-01 is the only one infected, 6 people may be susceptible to illness. The network is recursively extended for each infected user. In this way, users can be quickly found, informed and kept awake. Early identification of infected individuals is crucial in the prevention and containment of outbreaks of devastating diseases. The most effective way to fight an epidemic in urban areas is to quickly confine infected individuals to their homes. The agility of vaccination ranks second in effectiveness (Eubank et al., 2004). Information about people's social interactions can significantly reduce latency and improve the overall effectiveness of public health monitoring.

6 CONCLUSION

This work proposed a new approach for the detection of social interactions carried out in the daily life of the user. The approach consists of joining previously used methods such as: Bluetooth search to find people close to the user, localization and conversation identification. And new methods such as detection of the user who is talking. The collected data are used to generate social graphs that clearly demonstrate the relationships among users of a group.

The application collected satisfactory data for the development of social graphs capable of identifying people susceptible to infectious diseases. Providing the possibility for health professionals to intervene with agility in the control of epidemics. As future work, we intend to insert and adapt SocialCount in other areas, such as: sociology (identification of inclusion and social exclusion), business (employee relationship mapping), marketing (sales mapping by word-of-mouth), etc.

ACKNOWLEDGMENTS

The authors would like to thank CAPES for partial funding of this research and the UFSM/FATEC through project number 041250 - 9.07.0025 (100548).

REFERENCES

Adams, B., Phung, D., and Venkatesh, S. (2008). Sensing and using social context. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 5(2):11.

Baldauf, M., Dustdar, S., and Rosenberg, F. (2007). A survey on context-aware systems. *International Journal of Ad Hoc and Ubiquitous Computing*, 2(4):263–277.

Biamino, G. (2011). Modeling social contexts for pervasive computing environments. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on*, pages 415–420. IEEE.

Carter, I. (2013). *Human behavior in the social environment*. AldineTransaction.

Cassens, J. and Kofod-Petersen, A. (2006). Using activity theory to model context awareness: A qualitative case study. In *FLAIRS Conference*, pages 619–624.

Clayton, D. and Hills, M. (1993). *Statistical methods in epidemiology*. Oxford University Press.

Crickx, A. (2014). Recognito: Text independent speaker recognition in java. https://github.com/amaurycrickx/recognito.

de Araujo, R. B. (2003). Computação ubíqua: Princípios, tecnologias e desafios. In *XXI Simpósio Brasileiro de Redes de Computadores*, volume 8, pages 11–13.

De Cheveigné, A. and Kawahara, H. (2002). Yin, a fundamental frequency estimator for speech and music. *The Journal of the Acoustical Society of America*, 111(4):1917–1930.

Dey, A. K. (2001). Understanding and using context. *Personal and ubiquitous computing*, 5(1):4–7.

Eubank, S., Guclu, H., Kumar, V. A., Marathe, M. V., et al. (2004). Modelling disease outbreaks in realistic urban social networks. *Nature*, 429(6988):180.

Ferscha, A. (2012). 20 years past weiser: What's next? *IEEE Pervasive Computing*, 11(1):52–61.

Herrera, J. L., Srinivasan, R., Brownstein, J. S., Galvani, A. P., and Meyers, L. A. (2016). Disease surveillance on complex social networks. *PLoS computational biology*, 12(7):e1004928.

Hoseinitabatabaei, S. A., Gluhak, A., Tafazolli, R., and Headley, W. (2014). Design, realization, and evaluation of udirect-an approach for pervasive observation of user facing direction on mobile phones. *IEEE Transactions on Mobile Computing*, 13(9):1981–1994.

Kabir, M. A., Colman, A., and Han, J. (2014). Socioplatform: a platform for social context-aware applications. In *Context in Computing*, pages 291–308. Springer.

Kim, C. P. J. L. J. and Lee, S.-J. L. D. (2017). "don't bother me. i'm socializing!": A breakpoint-based smartphone notification system.

Kofod-Petersen, A. and Mikalsen, M. (2005). Context: Representation and reasoning. *Special issue of the Revue d'Intelligence Artificielle on" Applying Context-Management*.

Kolvenbach, S., Grather, W., and Klockner, K. (2004). Making community work aware. In *Parallel, Distributed and Network-Based Processing, 2004. Proceedings. 12th Euromicro Conference on*, pages 358–363. IEEE.

Lu, H., Pan, W., Lane, N. D., Choudhury, T., and Campbell, A. T. (2009). Soundsense: scalable sound sensing for people-centric applications on mobile phones. In *Proceedings of the 7th international conference on Mobile systems, applications, and services*, pages 165–178. ACM.

Matic, A., Osmani, V., Maxhuni, A., and Mayora, O. (2012). Multi-modal mobile sensing of social interactions. In *Pervasive computing technologies for healthcare (PervasiveHealth), 2012 6th international conference on*, pages 105–114. IEEE.

Miluzzo, E., Lane, N., Eisenman, S., and Campbell, A. (2007). Cenceme–injecting sensing presence into social networking applications. *Smart sensing and context*, pages 1–28.

Newman, M. E. (2002). Spread of epidemic disease on networks. *Physical review E*, 66(1):016128.

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

Palaghias, N., Loumis, N., Georgoulas, S., and Moessner, K. (2016). Quantifying trust relationships based on real-world social interactions. In *Communications (ICC), 2016 IEEE International Conference on*, pages 1–7. IEEE.

Schilit, B., Adams, N., and Want, R. (1994). Context-aware computing applications. In *Mobile Computing Systems and Applications, 1994. WMCSA 1994. First Workshop on*, pages 85–90. IEEE.

Six, J., Cornelis, O., and Leman, M. (2014). TarsosDSP, a Real-Time Audio Processing Framework in Java. In *Proceedings of the 53rd AES Conference (AES 53rd)*.

Temdee, P. and Prasad, R. (2017). Context-aware communication and computing: Applications for smart environment.

Teng, J., Zhang, B., Li, X., Bai, X., and Xuan, D. (2014). E-shadow: Lubricating social interaction using mobile phones. *IEEE Transactions on Computers*, 63(6):1422–1433.

Weisstein, E. W. (2017). Sir model. http://mathworld.wolfram.com/SIRModel.html.

Xu, Q., Chia, S. C., Mandal, B., Li, L., Lim, J.-H., Mukawa, M. A., and Tan, C. (2016). Socioglass: social interaction assistance with face recognition on google glass. *Scientific Phone Apps and Mobile Devices*, 2(1):7.

Yu, Z., Yu, Z., and Zhou, X. (2012). Socially aware computing. *Chinese Journal of Computers*, 35(1):16–26.

Zhang, R., Zhang, Y., Sun, J., and Yan, G. (2012). Fine-grained private matching for proximity-based mobile social networking. In *INFOCOM, 2012 Proceedings IEEE*, pages 1969–1977. IEEE.