Enhancing Human Social Interactions via Unmodified Smart Phone Ensembles

Edmund S.L. Lam
Carnegie Mellon University
sllam@andrew.cmu.edu

Ali Elgazar
Carnegie Mellon University
aee@cmu.edu

ABSTRACT

In an era of everywhere computing, connectivity is ubiquitous and smart devices are quite literally weaving themselves into the very fabric of our daily lives. Yet while people are more 'connected' than ever before, they risk becoming increasingly disconnected on a personal level. Smart devices, as they are used today, threaten to replace, rather than enhance or encourage human social interactions. They are generally geared to help us socialize *alone*, but more can be done to harness their capabilities to encourage us to socialize *together* (i.e. face-to-face). This paper aims to motivate more research into utilizing current technologies to enhance direct human social interactions. We discuss our ongoing work in exploring and developing new and less conventional methods of exploiting today's hardware (through the use of widely available, unmodified smart phones) to achieve enhanced face-to-face human social interactions, and ultimately a more engaging, collaborative/communal user experience when gathered in groups.

Author Keywords

Mobile games, Mobile ensembles, Multi-user, Smart phones, Human interaction, Social applications, Android

ACM Classification Keywords

H.5.m [**Information Interfaces and Presentation (e.g., HCI)**]: Miscellaneous

INTRODUCTION

Along with wearables, smart devices - whether they are sending push notifications from our wrists, or monitoring our health and fitness telemetries, or connecting us instantly (and incessantly) with friends, family and colleagues - are transforming our habits, behaviours, and ways of interacting with each other. Indeed, this transformation has many benefits. However, it is not entirely positive: it has in countless occasions already begun to replace, rather than enhance or facilitate human social interactions[1]. Too often it has brought along many anti-social behaviours - most commonly for example, using a mobile phone in a social setting is seen as being rude [Teevan et al., 2014]. Such negative perceptions are unsurprising given how using a smart device is often a solitary endeavour. When using our mobiles, we are usually encouraged to interact with others across the cloud rather than with the people next to us. Unfortunately, mobile applications that help promote human social interactions are far too few. There are positive examples of active research in this direction, however most require specialized setup of additional hardware (e.g., LCD touch surfaces [Goh et al., 2014, Ballagas et al., 2011]), or require modifications to standard mobile devices (e.g., embedded multi-antenna receivers in smart phones [Lucero et al., 2011]). There is little focus on harnessing the existing capabilities of smart phones currently widely proliferated and available on the mass market to enhance human social interaction.

This paper describes our on-going research into exploiting *mainstream* smart phones, using *existing hardware* (e.g., Wireless network, NFC, vibration), and *without modifying* factory default software (e.g., rooting the operating system) and *without introducing* new hardware (e.g., shared touch displays, auxiliary wearables). Even though our current studies are limited to mobile games, we believe that the features that they exploit can be adapted for mobile applications of a wider context. For instance, to enrich feature assortments of event mobile apps platforms (e.g., DoubleDutch, GuideBook), or human productivity enhancing mobile tools [Böhmer et al., 2013, Schuler et al., 2014]. We discuss three features that we have explored, namely using NFC as an actuation, making use of static orientation of smart phones and finally time synchronized haptic events across smart phones. In each section, we give a brief outline of the related works, describe the applications we have implemented and discuss our preliminary findings. Finally in the last section, we discuss our future works. Note that our applications are open-source and available for download at https://github.com/sllam/comingle . They are implemented with Android SDK and the CoMingle framework [Lam et al., 2015b, Lam et al., 2015a].

NFC AS AN ACTUATION

Over the years, Near Field Communications (NFC) has no doubt established itself as a primary feature in a smart

This work was made possible by grant JSREP 4-003-2-001, *Effective Parallel and Distributed Programming via Join Pattern with Guards, Propagation and More*, from the Qatar National Research Fund (a member of the Qatar Foundation). The statements made herein are solely the responsibility of the authors.
.

[1]By human social interactions, we refer to face-to-face human social interactions, not interactions mediated by computerized intermediaries (e.g., online gaming, social media).

phone's arsenal [Burkard, 2012], best known for its mainstream use to facilitate human social interaction in the form of peer-to-peer data transfer (e.g., Android Beam [Stern, 2011]). Yet new emerging applications of NFC rarely focus on explicitly enhancing human social interactions: common applications include facilitating fast and easy payment in modern integrated point of sales systems (e.g., Google Wallet [Google, 2015], RFID-Sim [Zou et al., 2010]), transactional management (e.g., mobile ticketing [Ghiron et al., 2009]), indoor localization [Ozdenizci et al., 2011] and health monitoring [Zhang and Li, 2011]. While they are excellent applications that improve human productivity, they are nonetheless not features that explicitly promote human interactions. The rare few exceptions that we know of are Assassins, an NFC-based mobile game [Cobb et al., 2013], and Junction, an NFC platform for building multi-party sessions [Dodson et al., 2010].

Real-time Battleships

We explored using NFC as an instrument to encourage human interactions: instead of treating it just as a means of data transfer, we have used NFC rather as form of synchronized actuation between two parties. Because of the physical constraints of NFC connections, requiring devices to be in close proximity, direct interaction with another human being is typically mandatory, in order to solicit consensus on establishing the connection. This feature is implemented in a simple multi-player mobile game, inspired by the traditional game "Battleships" with a few added twists: (1) more than two players can participate. Each armed with an evenly matched fleet of ships, players engage each other in a free-for-all guessing game to hunt down and destroy each other's fleet. (2) It is not turn based but instead, players can fire at an opponent whenever their in-game UI permits (guns require time to load). (3) players can form loose alliances through verbal negotiations, in the midst of the chaos. A player fulfills his/her obligation as an ally by establishing NFC connection with another party, after which he/she has a limited period of time to fix damage portions of his/her ally's ships.

The NFC feature in Real-time Battleship was implemented with the NFC library of the Android SDK, particularly the library for device to device NFC connections. However, the features used are not entirely standard: a typical application that uses NFC assumes that the application (the Android `Activity` class) on the receiver device is not active, hence an NFC "bump" invokes a fresh instance of that application activity on the receiver device, with the initial context information (`Intent`) containing the data sent by the NFC sender. For our application, NFC connections are intended to be actuations embedded within a game session, and during the life-cycle of the application, multiple NFC connections are expected to be made between various participates of the game. Hence establishing an NFC connection should not invoke a new instance of the application, rather be intercepted by the existing instance. Thankfully, the Android SDK's NFC library is robust enough to implement this without modification: our implementation does this with Android's foreground dispatch system, intercepting NFC data particularly sent by device of the same game session and using them within the context of the current game. Each game session establishes an underlying wireless network connection between all devices (currently available on LAN, Wifi-Direct or Bluetooth). An NFC connection is transient and only delivers unique identifier of the device that initiated the connection. All other in-game data is transmitted through the underlying wireless network connection.

Preliminary Findings

We wanted to observe changes (if any) in social dynamics in the real-time battleship game, simply by including the NFC feature as an additional dimension of interaction. During our prototyping and beta testing, we tried two versions of the game: one with the NFC feature included, while the other with the repairing functionality initiated traditionally with a UI push button. Both variants were tested with up to five players per session. Preliminary results are positive. While in general, our audience liked the real-time battleship game, they felt that the first version (no NFC) was an "as good as online" game: having players in proximity made little difference in the overall experience. Even though verbal exchange was encouraged for establishing ad-hoc alliances, it was not a novel experience, since verbal communication (through Voice Over IP) has long been a basic feature in online multi-player games. With the NFC "bump" to establish repair functionality, players seem more encouraged to attempt the process of negotiating alliances. The physical requirements of establishing an NFC connection explicitly seemed to affirm players' verbal intentions. Perhaps by forcing players to explicitly acknowledge each others' physical presence, it has in many occasions also created more opportunities for players to interact with each other. More interestingly, with the increased momentum of interactions, verbal exchanges were more profound (not in the abusive way) as more complicated forms of human interactions begin to integrate with the game play (e.g., politicking, deception), making the game more exciting and engaging.

POOR MAN'S INDOOR POSITIONING

Indoor positioning [Curran et al., 2011] has over the years been a grand challenge, especially in the context of positioning and orientation of mobile devices [Jovicic et al., 2013]. We have explored the prospects of using indoor positioning of unmodified smart phones for table-top games, however while several techniques come close to plausible solutions, each fell short for different reasons. For instance, several techniques are suitable and work on unmodified smart phones but require specialized hardware setup (e.g., touchscreen LCD display as table [Goh et al., 2014], controlled setup of modified commercial LED luminaries [Kuo et al., 2014]), [Lucero et al., 2011] demonstrate a promising photo sharing app that allows users to pass pictures by swiping towards each other, but requires multi-antenna receivers to be embedded under the screens of the devices. Approaches that use accelerometers and gyroscopes on smart phones are typically plagued by the notorious double integral issue (most succinctly explained in [Google, 2013b]), and relies heavily on various derivatives of Kalman filtering (e.g., [Paul and Wan, 2009]) to obtain more practical levels of accuracy. However, such approaches are highly specialized to the application, making implementation very costly, and furthermore are not likely (at this moment) to be accurate or responsive enough for table top interactive applications. [O'Sullivan, 2014] comes closest to our requirements, exploring the use of an unmodified smart phone's front-facing camera and wireless communication to exchange captured images of the ceiling above a group of devices to determine relative positioning between the devices. While this approach is promising, it requires more refinement in accuracy and recovery from camera obstruction (e.g., by a user's face).

From this initial set back, we rationalize that we have yet to consider (and accept) less glamorous solutions. For example, Chrome racer [Theory, 2013] is a novel distributed mobile game that allows players to race on a contiguous racing track across their smart phones. Smart phones are statically aligned based on unique indicators rendered on each device, once a racing group has been formed. While no doubt, a more sophisticated solution would to have each device magically recognize each other's positioning, this "poor man's" positioning has offered a simple alternative that has in no way ruined the novel gaming experience of racing across smart phones. Our experiments here are aimed at pushing what we call, "poor man's table top positioning", to its limit. To demonstrate this, we have implemented two distributed mobile games that we discuss in the following.

Scramble

Inspired by Chrome Racer [Theory, 2013] and Pass-them-around [Lucero et al., 2011], we begin exploring the idea of combining static alignment and orientation swiping from the two applications respectively. We implemented a competitive word game called Scramble. Loosely based on the popular board game Scrabble, Scramble has a few key differences: (1) it is played between four to six players, divided into two opposing teams. (2) Instead of a common board, each player has his/her own private list of words. (3) players get rid of a word from their list by extending it with letters from their personal rack. If the new extended word is correct, it can be toss at at an opponent. (4) when overwhelmed by words, a player can pass a word to an ally. When a player's word list is filled up with more than ten words, the player is booted from the game and the team which has all its players booted loses the game. To emphasize on each player's physical presence, each device renders a unique static orientation layout of all other players, that is consistent as a whole: players of each team are assumed to be in a row, each facing a unique opponent. On each device, icons representing each opponent are rendered on the top of the screen, towards the general direction of each opponent, while allies' icons are rendered on the sides (for the case of the player in the middle. Players at the end points of each row can only pass words to the ally in the middle). Even though the magic of dynamic tracking of peer device orientation is not present, static orientation still offer players a mental correspondence between the directional actions they conduct on their device's touch screens, with the physical presence of their opponents or allies.

Phone Hurdling

Phone Hurdling is a game where two teams of players race to help their virtual runner jump across their smart phones. The game is played as a competition between 2 teams of 3-5 players. Each team has its own runner rendered on a randomly selected device designated as the *current track*. The runner will move across the screen (left to right) at a constant speed. Meanwhile, another randomly selected team mate's device is assigned as the contiguous connection of the racing track (i.e., the *next track*). The owner of next track must scramble to place his/her smart phone to the right of current track, before the runner reaches the end of the current track device. As runners progress across devices, their speed increases, making it progressively harder for the players to keep up. The team which fails to keep their runner hurdling across a phone loses the game.

The obvious challenge to implement this game is of course, how to verify that the next track is positioned correctly (to the right of current track) before the runner reaches the end of current track. While this game concept could have easily been implemented on devices with specialized directional sensors (e.g., Stifeo [Merrill and Kalanithi, 2009]), implementing it on unmodified smart phones requires a solution for highly accurate and highly responsive indoor positioning, or perhaps some creative thinking and lowering of expectations. Particularly, our implementation recreates this experience by having the current track device render a button on its lower right corner, while the next track device renders another on its lower left corner. Verification is done by requiring the buttons to be simultaneously clicked (with a small time differential allowance). This makes it intuitive for players to physically place the phones side-by-side and simultaneously hit the buttons with one swift action. While clearly not as challenging as actual indoor positioning, the technical challenge of tracking the time difference between button clicks on separate devices was not entirely straightforward, given that clocks on unmodified smart phones are typically out-of-sync by seconds, or even minutes [Google, 2013a]. Our implementation was made simpler with the help of work in [Lam et al., 2015a] on a programmable abstraction for time-synchronized event across mobile devices. Particularly, the task of computing time offsets of out-of-sync devices is abstracted away and handled by our runtime system.

Preliminary Findings

Our preliminary findings show that these "poor man" alternative solutions work well in recreating the experience we desired. In the case of Scramble, static orientation of ally and opponent icons around each smart phone's UI display provided a quick way to disambiguate player-to-player operations through directional swiping. We have tested two versions of the game: (1) with static orientation and directional swiping, and (2) with the traditional UI click buttons. Preliminary results show that players preferred version (1), citing that there was a stronger sense of immersion with directional swiping resembling the action of tossing words at their friends. Furthermore, the pace of the game was faster and more enjoyable. When asked about the absence of actual dynamic orientation and tracking of peer devices, they agreed that it would definitely make the game even more interesting, but they would gladly accept the compromise (i.e., version (1)) than revert to version (2). Some also commented that the limitations on ally word passing (not all allies can pass words to all other allies) was a clear emphasis of the limitations of static orientation. An interesting suggestion was to replace the static orientation of allies, with NFC connection for passing words to allies. In the case of Phone Hurdling, players still found the novelty of the game enjoyable and the rush of getting phones in position exhilarating. Players however, do eventually attempt to challenge and cheat the system, by "human synchronizing" click events (i.e., shouting out “1-2-3”).

Indeed the absence of true positioning would eventually be noticed: be it because of a Scramble opponent’s futile attempt to physically dodge an attack, lazy Phone Hurdlers realizing that phones need not actually be side-by-side, or a tech savvy player calling the games “hacks” (or just simply being a wet-blanket). But the overall experiences of using the applications are not significantly sullied. Such “hacks” in general will not always work and there will always be applications that uncompromisingly require true indoor positioning. But we believe that these less glamorous solutions still have a part to play in a mobile software engineer’s arsenal of development tricks, to deliver features that enhances human social interactions via unmodified smart phones.

SYNCHRONIZED HAPTIC EVENTS

We are currently exploring more applications of distributed time synchronized haptic events across unmodified smart phones. Although games and applications in gaming consoles (e.g., Wii, PSP, etc..) occasionally exploit synchronized haptic events as a interactive feature, to the best of our knowledge, no extensive study in this has been done solely in the context of smart phones. In the following section, we discuss a mobile game that utilizes this feature, called the Mafia party game.

Mafia Party Game

Mafia is a party game played among eight or more players. Initially, a third of the players are secretly assigned to be criminals in the mafia, while the rest are innocent citizens. The game proceeds in day and night cycles: during the night, citizens sleep while criminals are free to silently roam and decide to murder one citizen. In the day, all citizens and mafia members (whose identities are unknown to the citizens) debate on who the mafia members are and vote on one suspect to execute. The citizens win when all mafia members are executed, while the mafia wins when the surviving mafia members outnumber the remaining citizens. In the traditional game, one person in the party acts as the moderator. The moderator does not participate as a player but manages the day and night transitions, while keeping track of the number of surviving players in each team.

In our Mafia game app, the role of the moderator is collectively assumed by the ensemble of smart phones. Initially, when the game is started, each player receives a private message, informing them of their identities (mafia or citizen). Day to night transitions proceed as follows: (1) 10 second before night time all devices emit a warning beep, prompting players to sit down, hold their devices faced down, and with their eyes closed. (2) Once night has arrived, all devices vibrate at the same time, indicating that players should fall asleep. (3) 5 seconds later, only the mafia devices vibrate once more, indicating that they can wake silently and begin their clandestine evil deeds. Night to day transitions proceed similarly, with a 10 seconds warning buzz to notify mafia members to silently proceed to their original positions and pretend that they have been sleeping through the night.

Mafia would have been a challenging application to implement, because of its requirement for precisely timed vibration events across an ensemble of smart phones. Similar to the Phone Hurdling game discussed earlier, its implementation was fairly straight-forward, with the time-synchronization problem abstracted away by [Lam et al., 2015a].

Preliminary Findings

Our experience with testing this game was encouraging. Most people felt that it introduces a novel way of using smart phones. Many suggest that the integration of smart phones into the game play has greater potential: while the current implementation simply replaces the human moderator with computerized intermediaries, the game play could be further enhanced if we also exploited the smart phones as personal interactive displays. Suggestions include having an interactive anonymous polling of mafia suspects and private messaging between parties. Complaints about the app mainly were centered on two things: (1) its current cosmetically unappealing state, and more interestingly (2) that some smart phone models have louder vibrations that could be too audible, making its unfortunate mafia owner less stealthy. Opinions on loud vibrating smart phones are mixed, with some believing that it makes the game unfair, while others feel that it provides an additional topic of debate about who the mafia members are.

However, the most profound observation about our Mafia app is interestingly not about the game play. Instead, it is related to our experiences in articulating the concept of the application. Particularly, we discovered that a common reaction upon first hearing about the game, was reluctance and reservations of the idea of playing the game online (as a party of one). This is of course, dismissed by our strong clarification that the game was absolutely not intended to be played online[2] but instead, played as shared interactive experience. Akin to observations made in [Teevan et al., 2014], this suggests a disturbingly strong inclination for mainstream smart phone users to assume that smart phones are instruments that replace, rather than enhance or promote human social interactions.

FUTURE WORK AND CONCLUSION

Most of our findings so far are mainly anecdotal experiences, consolidated during our prototyping and beta testing of the applications we have implemented. In future, we intend to derive and conduct more formal experimentations on the various techniques we have discussed in this paper. Particularly, we seek to gather more quantitative feedback from human subjects of various backgrounds and levels of tech savviness, and explore more applications for each of the techniques we discussed in this paper. We also intend to develop more proof-of-concept for the interaction techniques we have discussed here, as well as explore new interesting ways of exploiting unmodified smart phones for enhancing human social interaction. For instance, building social interactive features from state-of-the-art in-air gestures with unmodified smart phones [Song et al., 2014].

As the years go by, there will be more and more smart devices appearing in the mainstream market. Indeed even the smart phones of tomorrow will include arrays of hardware that we never dreamed possible. But we only hope that this paper would inspire two values: (1) do not forget the less glamorous solutions, and (2) new disruptive wearables of the future should, above all, still be about enhancing and promoting human social interactions and experiences.

[2] Additionally, we had to ferociously insist that we were not agoraphobic.

REFERENCES

[Ballagas et al., 2011] Ballagas, R., Revelle, G., Buza, K., Horii, H., Mori, K., Raffle, H., Spasojevic, M., Go, J., Cook, K., Reardon, E., Tsai, Y., and Paretti, C. (2011). Electric Agents: Combining Television and Mobile Phones for an Educational Game. In *IDC'2011*, pages 227–230, New York, NY, USA. ACM.

[Böhmer et al., 2013] Böhmer, M., Saponas, T. S., and Teevan, J. (2013). Smartphone use does not have to be rude: Making phones a collaborative presence in meetings. In *MobileHCI'2013*, pages 342–351, New York, NY, USA. ACM.

[Burkard, 2012] Burkard, S. (2012). Near field communication in smartphones. *Dept. of Telecommunication Systems, Service-centric Networking, Berlin Institute of Technology*.

[Cobb et al., 2013] Cobb, J., Lett, J., McSwain, J., and Schwarz, L. (2013). Assassins NFC-based Android Game, AT&T Mobile Hackathon. https://www.facebook.com/media/set/?set=a.502165156485955.1073741826.3174.

[Curran et al., 2011] Curran, K., Furey, E., Lunney, T., Santos, J., Woods, D., and McCaughey, A. (2011). An Evaluation of Indoor Location Determination Technologies. *J. Locat. Based Serv.*, 5(2):61–78.

[Dodson et al., 2010] Dodson, B., Bojinov, H., and Lam, M. S. (2010). Touch and run with near field communication (nfc). *Dept. of Computer Science, Satnford University*.

[Ghiron et al., 2009] Ghiron, S., Sposato, S., Medaglia, C., and Moroni, A. (2009). Nfc ticketing: A prototype and usability test of an nfc-based virtual ticketing application. In *NFC'2009*, pages 45–50.

[Goh et al., 2014] Goh, W. B., Chen, M., Trinh, C. H., Tan, J., and Shou, W. (2014). The MOY Framework for Collaborative Play Design in Integrated Shared and Private Interactive Spaces. In *CHI'2014*, pages 391–400, New York, NY, USA. ACM.

[Google, 2013a] Google (2013a). Android Issue Tracker: User App cannot set System Time. https://code.google.com/p/android/issues/detail?id=4581.

[Google, 2013b] Google (2013b). Google Tech Talk, Sensor Fusion on Android Devices: A Revolution in Motion Processing. https://www.youtube.com/watch?v=C7JQ7Rpwn2k.

[Google, 2015] Google, I. (2015). Google Wallet. https://www.google.com/wallet/.

[Jovicic et al., 2013] Jovicic, A., Li, J., and Richardson, T. (2013). Visible light communication: opportunities, challenges and the path to market. *Communications Magazine, IEEE*, 51(12):26–32.

[Kuo et al., 2014] Kuo, Y.-S., Pannuto, P., Hsiao, K.-J., and Dutta, P. (2014). Luxapose: Indoor Positioning with Mobile Phones and Visible Light. In *MobiCom'2014*, pages 447–458, New York, NY, USA. ACM.

[Lam et al., 2015a] Lam, E., Cervesato, I., and Elgazar, A. (2015a). Programmable Orchestration of Time-Synchronized Events Across Decentralized Android Ensembles. In *WiMob'2015*, Abu Dhabi, UAE.

[Lam et al., 2015b] Lam, E., Cervesato, I., and Fatima Haque, N. (2015b). Comingle: Distributed Logic Programming for Decentralized Mobile Ensembles. In Holvoet, T. and Viroli, M., editors, *DisCoTec(Coordination)'2015*, pages 51–66, Grenoble, France. Inria, Springer LNCS 9037.

[Lucero et al., 2011] Lucero, A., Holopainen, J., and Jokela, T. (2011). Pass-them-around: Collaborative Use of Mobile Phones for Photo Sharing. In *CHI'2011*, pages 1787–1796, New York, NY, USA. ACM.

[Merrill and Kalanithi, 2009] Merrill, D. and Kalanithi, J. (2009). Sifteo, Interactive Game Cubes. https://www.sifteo.com/cubes.

[O'Sullivan, 2014] O'Sullivan, A. (2014). Multiphono: Relative positioning of co-located mobile devices. *Dept. of Computer Science, University of Maryland*.

[Ozdenizci et al., 2011] Ozdenizci, B., Ok, K., Coskun, V., and Aydin, M. (2011). Development of an indoor navigation system using nfc technology. In *ICIC'2011*, pages 11–14.

[Paul and Wan, 2009] Paul, A. S. and Wan, E. (2009). Rssi-based indoor localization and tracking using sigma-point kalman smoothers. *IEEE Journal of Signal Processing*, 3(5):860–873.

[Schuler et al., 2014] Schuler, R. P., Grandhi, S. A., Mayer, J. M., Ricken, S. T., and Jones, Q. (2014). The Doing of Doing Stuff: Understanding the Coordination of Social Group-activities. In *CHI'2014*, pages 119–128, New York, NY, USA. ACM.

[Song et al., 2014] Song, J., Sörös, G., Pece, F., Fanello, S., Izadi, S., Keskin, C., and Hilliges, O. (2014). In-air Gestures Around Unmodified Mobile Devices. In *UIST '14*, pages 319–329, New York, NY, USA. ACM.

[Stern, 2011] Stern, J. (2011). Android Beam: pictures, video and hands-on. http://www.theverge.com/2011/10/19/android-beam-pictures-video-hands-on.

[Teevan et al., 2014] Teevan, J., Morris, M., and Saponas, T. (2014). Mobile support for face-to-face social interaction. *HCIC'2014*.

[Theory, 2013] Theory, A. (2013). Chrome Racer, A Chrome Experiment. https://www.chrome.com/racer.

[Zhang and Li, 2011] Zhang, H. and Li, J. (2011). Nfc in medical applications with wireless sensors. In *ICECE'2011*, pages 718–721.

[Zou et al., 2010] Zou, J., Zhang, C., Dong, C., Fan, C., and Wen, Z. (2010). Mobile payment based on rfid-sim card. In *CIT'2010*, pages 2052–2054.