AlwaySocial: Social Networking in the Real World

Nir J. Peer
Department of Computer Science
A.V. Williams Building
University of Maryland
College Park, MD 20742, USA
peer@cs.umd.edu

ABSTRACT

Social networking is nowadays a popular way for people to socialize and network professionally. Currently, social networking websites provide a mostly online experience whether they are accessed from a computer or a mobile phone. This leads to a chasm between online social activities and those done in the actual world. As there is no direct way to turn an online acquaintance to a real friend, users are required to constantly synchronize between the two activities.

Social Proximity Applications (SPA) aim to introduce technology into real world networking in order to facilitate it. However, they often use proprietary and restricted profile systems and therefore do not bridge the online/offline gap. In this paper we present AlwaysSocial, a Facebook based SPA that allows users to discover other nearby users, chat with them, and retain their Facebook contact information in case they wish to become online friends. It enhances networking by making a user's public Facebook profile available to other nearby users. Since the system is proximity based, any digital interaction can be quickly turned into face-to-face interaction and vice versa.

ACM Classification: H5.2 [Information interfaces and presentation]: User Interfaces. - Graphical user interfaces.

General terms: Design, Human Factors.

Keywords: Social networking, mobile, proximity, wireless, bluetooth, awareness, presence.

INTRODUCTION

Social networking websites such as Facebook [1], MySpace [3], or LinkedIn [2] provide a venue for specifying social connections, sharing information with friends, and meeting new people. Their appeal may be attributed to several factors. They allow users to maintain contact with their friends with minimal effort; by updating their profile and posting status updates, users can stay in touch with a large group of friends instead of just their closest ones. In addition, they act as an enhanced directory service that allows users to keep a communication path with friends and acquaintances even when they relocate or change their contact information. Furthermore, by exposing one's friends, users are able to reconnect with old contacts who they have lost touch with. In other words, social networks allow users to maintain existing- and reconstitute lost- social connections. They also allow to meet or get introduced to new people depending on the particular website's etiquette. Another exciting feature of social networks, is that they sometimes allow to discover cliques and mutual connections that were not apparent. This is closely related to the Small World or six degrees of separation hypothesis.

The phenomenal success of online social networking has been a catalyst in an industry-wide shift from technology-oriented design to a more goal-oriented design. For example, photo sharing websites that used to essentially provide storage space and a sharing service have refocused around sharing photos as a form of social activity. After all, photos are used to convey a story, document an event or experience, to articulate a social connection (e.g., a group photo), retain a memory, etc. That is why social networking websites which arguably provided an initially inferior photo sharing service were so enthralling; they allowed users to tag people and places shown in photos, thus relating between the two. This refocus around *people* and their activities is bound to continue as the industry matures. As an instance, today to communicate with a person, one has often to think of an *address* such as an email address, phone number, instant messaging screen name, etc. This situation is reminiscent of the pre-DNS days, where servers and websites were accessed using their IP address. One could envision a future unified directory service or "DNS for people" that would enable one to contact *a person* using voice, video, text or any other means and through any medium of choice such as a mobile phone, a computer, etc. The address used to route the communication will become inconsequential, as it should be. Incidentally, it is worth noting that, properly implemented, having such a central database could enhance privacy, as it could hide the user's address from those trying to contact her.

Despite their popularity, social networking websites provide an inherently online experience. Users have to actively maintain them by adding friends and acquaintances, and when they get introduced to a person online, that information is not directly actionable in the sense that it only allows for mediated communication. This makes online social networking a somewhat detached activity from networking in the real world. While many social websites do provide a mobile version, it is generally just a wireless accessible, small screen, low-bandwidth version of the same website. It does not attempt to bridge the online and physical world. On the other side of the spectrum there are Mobile Social Software (MoSoSo) and Social Proximity Applications (SPA), which are rooted in the physical world, supporting social interaction between physically proximate users. These applications typically use a proprietary profile system and limit users to *only* interacting in the real world. Hence, they do not support meeting online acquaintances or friends of friends in the real world, even when they happen to be nearby. Our project is aimed at demonstrating how one popular social website, Facebook, can be integrated into real world networking.

SPAs operate on the premise that being at the same place at the same time may indicate belonging to the same milieu or having other things in common. There are countless examples—for instance, two scientists who are present the same conference are probably from a related research field, people who are invited to the same private cocktail party probably know someone in common, people who attend the same music event may be fans of the same band or share love for the same genre, people waiting at the same bus stop may be heading in the same direction or even live in the same area. It is also the case that the type of people in a certain place at a given time possibly tells something about the place such as it being a workplace or a specific social club. Therefore, instead of looking at a people's physical coordinates, there could be greater value in determining their social positioning, that is, find which other people coinhabit the same space as they do [17].

A Bluetooth-enabled mobile phone is a natural candidate for implementing an SPA. It is a quite ubiquitous *personal* device. Since Bluetooth is a close-range ($<$ 10 meters) wireless communication protocol, it can be used for mobile-to-mobile communication, to advertise user presence, and discover nearby users. As mobile data communication becomes more prevalent, social interactions in the real world could be reflected online in real time.

RELATED WORK

Mobile social applications have been the subject of extensive research. In the following we survey some notable projects that are closely related to ours. Other work, particularly in the area of Active Badges, had also explored ways of using technology to augment real-world social interaction [13, 12, 5]. Refer to [4] for a more comprehensive, albeit somewhat outdated, survey of mobile social applications.

Serendipity [6] provides a mobile profile matching and introduction service between physically proximate users. It uses periodic Bluetooth scanning to discover new Bluetooth IDs (BTIDs). These are reported to a central server that consults its database of user profiles in order to calculate a similarity score. If the score is above a certain threshold, the server alerts the matching users of the possible mutual interest. Serendipity employs proximity information combined with other parameters, such as time of day, to infer the nature of the relationship between any two proximate users. The service actively tries to introduce people to one another, but provides some user control to mitigate the disruptiveness of this approach. By using a central server, Serendipity requires an active Internet connection, thereby limiting its applicability. This technology is being commercialized by MetroSpark.com. In contrast to AlwaysSocial, Serendipity is an active introduction system. Although users are given control over the weights used to determine interestingness, it is our opinion that this can only lead to rough matches. Successful matchmaking is challenging to humans, let alone computers. There are many factors that affect whether two people would find interest in one another, and these may be hard to quantify. Therefore, we prefer an approach that puts users in charge of the matchmaking, based on public profiles.

MobiTip [17] allows users to exchange opinions or tips between proximate mobile users. It visualizes the active nearby devices to show were tips originated from. The tips are filtered based on similarity between users, on collocation as determined by Bluetooth, and on user-supplied popularity rating. Since the information is distributed among the mobile devices, optional Bluetooth hotspots can be used as local anchor points to moderate the ephemeral nature of these device-to-device encounters. The hotspots collect information from mobile users passing by, making it available to future visitors.

Social Net [18] uses short-range RF communication to track the frequency and duration of user encounters over a two-week period. Users who meet often or meet infrequently for long durations are considered potential friends or candidates for introduction. If mutual friend of them steps in, she receives a message suggesting she introduce them to one another. This leverages humans' social skills to decide whether two people should be introduced, and determine the timing and the manner this should be done. Social Net was implemented on the now defunct Cybiko devices.

In Scent, DigiDress, and later in Sensor [10, 16, 14], Nokia experimented with various iterations of an application in the area of so-called Mobile Social Software (MoSoSo). The application focused on collocated users, making it a Social Proximity Application (SPA). Each user can create a mobile profile page (Folio [14]) that can be discovered by nearby users using a compatible Bluetooth enabled phone. The system is passive in the sense that scanning for other proximate users has to be manually invoked. Another drawback of the system is that it requires users to maintain a proprietary social network and so they cannot take advantage of any existing connections with friends. During evaluation, one complaint users had was that scanning was "really slow". Also quite often no nearby users were found [16]. This was sometimes caused by the unreliability of Bluetooth scanning as explained in the FAQ in [14]. A nice feature of the system is that in order to facilitate the adoption of the application, there was a feature that enabled users to send a copy of the software to other nearby users.

Hummingbird [7] is a designated handheld device used as an *Inter-Personal Awareness Device (IPAD)*. Its purpose is to facilitate communication rather than mediate it. It does so by providing presence information to help initiate communication, which will be carried out by other means. The device promotes group awareness by "humming" whenever a member is nearby (< 100 meters). It uses an LCD display to display all nearby group members. The device was used both in a *familiar setting* (e.g., office environment) and in an *unfamiliar setting* (e.g., rock festival or academic conference). Users' satisfaction was much higher in the latter case and the authors attribute that to the fact that in such a setting there is less certainty in members' whereabouts. Because of the relatively long range, users often expressed frustration when a member shown on the Hummingbird as nearby could not be easily located.

Jabberwocky [15] investigated how knowledge of being collocated with previously encountered strangers helped induce comfort and reduce anxiety. The project is centered around the idea of *Familiar Strangers*, a term coined by the psychologist Stanley Milgram. Unlike AlwaysSocial and many other SPAs, Jabberwockies are not meant to turn strangers into acquaintances. The authors refer to sociological and psychological evidence that shows that having strangers is in fact critical to the functioning of an urban environment. In particular, if one had to acknowledge and interact with each and every individual in a densely populated environment, this would quickly lead to overload. Therefore, the polite and expected behavior in public spaces such as elevators or when using public transportation, is often "civil inattention". An experiment conducted by Milgram and replicated by the authors has shown, however, that people do recognize certain strangers in places that they frequent regularly. This can contribute to a sense of familiarity and increased comfort. Jabberwockies allow users to determine if they are surrounded by many familiar strangers.

Cityware [11] is a recent project that aims at collecting real-world social information and integrating it into the online social network Facebook [1]. By deploying Cityware nodes throughout a city, the Bluetooth IDs of mobile devices are collected. This information is analyzed to determine which devices were present at the same time in the same place. This information is gathered about any device with Bluetooth in discoverable mode. There is no need to install any special software on the device. Facebook users who add the Cityware application to their profile can tag their device or their friends' devices. After doing so, they are able to see encounters that took place in the real-world linked to Facebook profiles. Cityware is complementary to our system in that it allows *after the fact* analysis of social encounters whereas ours brings online social information to the real-world, making it actionable information. Cityware requires the deployment of Cityware probes and so limits its applicability to certain areas of the city.

REQUIREMENTS

In designing AlwaysSocial, we set the following requirements. We believe that our focus on leveraging existing hardware and a popular social networking website could help AlwaysSocial to be immediately useful to large audiences.

- Must use an existing popular social network
- Must use standard hardware in order to reduce costs and remove obstacles to adoption
- Must be based on proximity or collocation; this is achieved by using short-range wireless communication, i.e. Bluetooth
- Must not require active Internet connection
- Must allow free communication among proximate users, so as not to discourage frequent spontaneous communication
- Must allow users to retrieve a list of people encountered after the fact

IMPLEMENTATION

AlwaysSocial is implemented as a Windows Mobile application written in C#. It communicates over Bluetooth using the InTheHand 32feet.NET library [9]. This library supports the Microsoft Bluetooth protocol stack in both the mobile and desktop versions of Windows. The Facebook Developer Toolkit [8] is a managed library that provides the access to the Facebook API.

Once started, AlwaysSocial periodically scans for nearby Bluetooth devices. It uses a Service Discovery Protocol (SDP) query to determine if a peer device supports the AlwaysSocial service. Originally, once a device had been determined as supporting AlwaysSocial, the querying device would initiate a connection request to ask for the Facebook User ID associated with that device. In our experimentation we observed that many of these connection attempts failed because the peer was busy scanning for other devices. In order to circumvent this problem, we instead advertise the Facebook User ID as part of the service descriptor. Given the user's Facebook ID, AlwaysSocial can query Facebook for the user's public profile over a mobile Internet connection, or it can connect to the peer and ask for that information directly from it. As mentioned earlier, the latter option is less reliable, but would keep the system usable when an Internet connection is not available on the querying device. Furthermore, Internet access is not strictly necessary on the peer device either, since our system caches the users' Facebook profiles. We added several features in order to improve the robustness of our system. We observed that often, when the peer device was busy scanning, our SDP query would fail and thus we may falsely assume that the peer does not support AlwaysSocial. In addition, sometimes SDP queries would discover a device but would not have its human-readable name available. Thus, we use retries to overcome intermittent failures and we periodically recheck Bluetooth devices that were found to not support AlwaysSocial, in case the service gets started later.

The main screen of AlwaySocial shows the current status (see Figure 1). The owner's profile picture and name appear at the top of the screen. Beneath it appear the names and pictures of all the nearby peer users that were found during scanning. Each such nearby peer is called a *proximate*. Whenever automatic scanning is initiated, a binoculars icon appears at the top-right corner of the screen. As new proximates are found, they are added and shown in the bottom container. At any time, the user can select any of the proximates to initiate a chat session with them. The chat window supports multi-chat (see Figure 2). By clicking on any

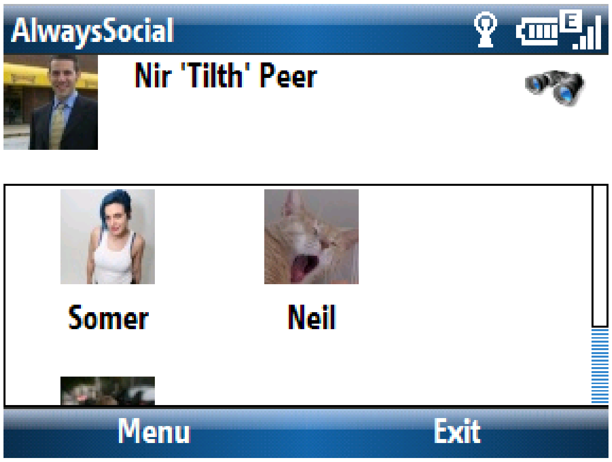

Figure 1: Scanning for proximate users. Three proximates have already been found—Somer, Neil, and Bontgomery (which can be seen by scrolling down)

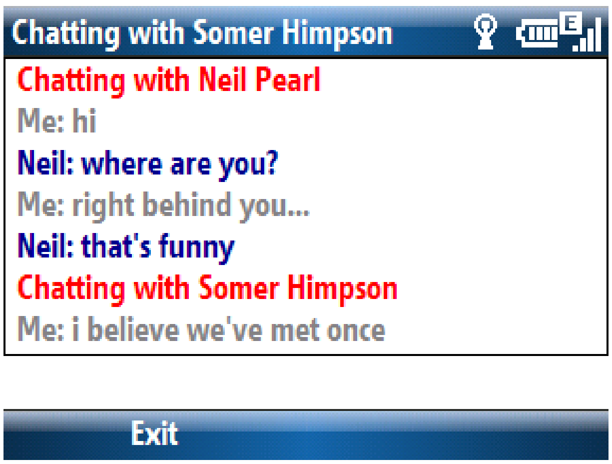

Figure 2: Chatting with two proximate users.

chat message received from another proximate, a chat session with that specific user will ensue.

Since we only had one Windows smartphone at our disposal, in order to test our system with multiple proximates, we simulated several mobile AlwaysSocial users using a desktop version of AlwaysSocial installed on Bluetooth-enabled laptops (see Figure 3). The desktop version was realized by porting AlwaysSocial to a desktop Windows application, thereby obtaining a functionally equivalent version. The AlwaysSocial desktop interface is shown in Figure 4.

EVALUATION

We envision two main uses for AlwaysSocial. In the first, a user is relatively stationary and looking for proximates. This is often the case when socializing or networking. The other use is when the user is moving around, for example when passing by an exhibition.

In order to evaluate AlwaysSocial we deployed four clients running on 1 smartphone and 3 laptops. These machines, as well as six others that were Bluetooth discoverable but did not run AlwaysSocial are shown in Table 1.

Figure 3: Bluetooth-enabled laptops simulating smartphone AlwaysSocial users.

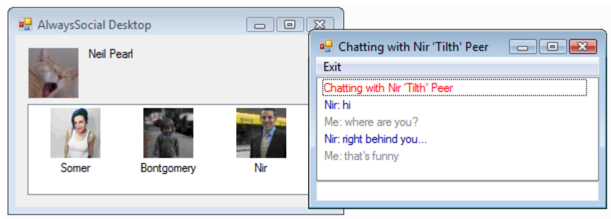

Figure 4: Chat on the desktop version of AlwaySocial.

We tested the system many times and then manually analyzed detailed log files for two specific runs. Our latpop clients were kept fixed and the mobile phone was used while walking around an office room. Table 2 shows some statistics for one of our representative runs. The scanning time until the first AlwaysSocial client is found is normally in the order of a few seconds, however there is a large variability. Naturally, if more non-AlwaysSocial Bluetooth machines are queried before, it will delay the discovery of the first one. Also, if a machine was not found due to failure of the SDP query, it will take at least an additional scanning cycle to discover it. These SDP failures are quite common, and so the total scanning time until all clients are discovered is in the order of a few minutes.

Table 1: Bluetooth clients active during the experiment (the lower part of their Bluetooth address is anonymized).

Address	Name	AlwaysSocial?
0018DBXXXXXX	BT500	-
0020E0XXXXXX	LINEN	✓
001A45XXXXXX	Motorola H700	-
001060XXXXXX	PADDHP	-
00197EXXXXXX	PALLADIUM	✓
0014A4XXXXXX	SILK	-
001060XXXXXX	SOBO	✓
0010C6XXXXXX	TAISHAN	-
0012D2XXXXXX	T-Mobile56	✓
001F5BXXXXXX	unryu	-

Table 2: Scanning statistics for a representative run. $\#_{before}$ is the number of Bluetooth clients discovered before any AlwaysSocial user was found. It took t_{first} to find the first and t_{all} to find the rest.

Client	$\#_{before}$	t_{first}	t_{all}
Smartphone	2	11 sec	47 sec
Laptop T	4	17 sec	120 sec
Laptop D	5	30 sec	137 sec
Laptop I	7	133 sec	265 sec

With these relatively long acquisition times, for AlwaysSocial to acquire all proximates it should be used either in small rooms where the peer Bluetooth clients would remain in range even if the user is moving fast, or in scenarios where the user is almost stationary. In any case, in slow to moderate walking speed, the user should be able to locate at least some proximates.

FUTURE WORK

We would like to implement AlwaysSocial on Symbian Series 60 which is another popular Smartphone OS. In addition, the forthcoming Google Android, promised to be an open mobile platform, is also a likely candidate. In order to increase the application's appeal, the Google OpenSocial API may be used in order to support several social networking websites. However, currently its RESTful Data APIs still have not been released.

We also would like to implement a handshake functionality which will easily mark two people as online friends. This can be implemented using current hardware in multiple ways. One possibility is to send a special handshake message over bluetooth. Another is to use IrDA (infra-red) communication, requiring both handsets to face one another. Yet another alternative, if the phone is camera-enabled, is to use image recognition to scan a barcode displayed on the peer mobile phone.

We would need to evaluate AlwaysSocial with a broad set of users. It will be interesting to see in what ways it gets used. Users' privacy concerns will also be of great interest. We may have to complement the Facebook profile system with additional limited public information to be used in conjunction with AlwaysSocial, in case the current public information is considered too revealing.

DISCUSSION AND CONCLUSIONS

Social Proximity Applications (SPAs) are useful to connect people with their friends (when they are close-by but obstructed from view) and to facilitate social interaction with strangers. Using personal profiles, users can express their identity, capture their current mood, or share some of their thoughts and ideas with other people nearby [16]. Doing so can help create a sense of community, solidarity, or form a basis for social interaction with new people [15].

We believe that while many social applications transcend physical distance by connecting friends wherever they are, there is also a need for facilitating interaction between collocated strangers. It makes sense for proximate individuals to interact directly without any mediation, but environmental conditions, social appropriateness, or psychological inhibitions could all be an obstacle for engaging in a conversation with a stranger. Using an SPA, an individual can find nearby people, determine interest based on their public profile, find topics for conversation, and then interact directly or through the SPA. In essence, the information captured in the public profile can promote interaction by making a stranger somewhat familiar. In addition, it provides potential ice breakers, and allows individuals to better evaluate if they are interested in interacting with one another to begin with. The ability to immediately transition into rich unmediated interaction makes this type of introduction very compelling.

In any social networking, privacy can become an issue. Users should be able to turn the service off, for example if they're concerned about being trackable. In addition they should be allowed to maintain their desired level of anonymity by controlling what kind of information gets published on their profile. For instance, one person may publish their picture whereas another may use an avatar instead. Human social behavior is intricate, and so sometimes people would want to be discoverable to some but not to others. But by being in the same physical surrounding, they risk being spotted out even without the aid of an SPA, therefore this may not be a serious issue.

In terms of the wireless communication technology being used, we had somewhat mixed success with Bluetooth. The benefits of using this technology is that it is readily available on many handsets and that is short-range thus supports interaction based on proximity in a same-room scale. However, as discussed previously, using current Bluetooth technology, scanning is slow and discovery is quite unreliable. Another considered option, that of using WiFi was ruled out since it is currently not as common and its range is too large for this application. In addition, it generally relies on having fixed access points and so will not work where WiFi is not deployed. Limiting the wireless range makes it more probable that direct interaction will be possible, and it helps keep the number of discovered users manageable. Naturally when people socialize or network they are not looking to meet every person, but rather develop good rapport with some interesting people.

To conclude, from an informal survey that we conducted, people seemed excited when AlwaysSocial was described or demonstrated to them. While socializing is rightfully perceived as a human activity, we believe it can be enhanced through the use of an application such as AlwaysSocial. By being able to quickly shift from the virtual to the real world, it reduces any potential aversion to the use of technology for such a purpose. We believe that a great advantage of AlwaysSocial over ordinary social networking websites is that it provides profile information where and when it can be acted upon through unmediated interaction.

REFERENCES

1. Facebook. http://www.facebook.com, May 2008.

2. Linkedin. http://www.facebook.com, May 2008.

3. Myspace. http://www.myspace.com, May 2008.

4. Timo Arnall. Mobile social software applications. http://www.elasticspace.com/2004/06/mobile-social-software, May 2008.

5. Richard Borovoy, Michelle McDonald, Fred Martin, and Mitchel Resnick. Things that blink: Computationally augmented name tags. *IBM Systems Journal*, 35(3&4):488–495, 1996.

6. Nathan Eagle and Alex Pentland. Social serendipity: Mobilizing social software. *IEEE Pervasive Computing*, 4(2):28–34, 2005.

7. Lars Erik Holmquist, Jennica Falk, and Joakim Wigstrm. Supporting group collaboration with interpersonal awareness devices. *Personal and Ubiquitous Computing*, 3(1-2):13–21, March 1999.

8. Clarity Consulting Inc. Facebook developer toolkit. http://www.codeplex.com/FacebookToolkit, May 2008.

9. InTheHand. 32feet.net library. http://www.codeplex.com/32feet/, May 2008.

10. Younghee Jung, Jan Blom, and Per Persson. Scent field trial: understanding emerging social interaction. In *MobileHCI '06: Proceedings of the 8th conference on Human-computer interaction with mobile devices and services*, pages 69–76, New York, NY, USA, 2006. ACM.

11. Vassilis Kostakos and Eamonn O'Neill. Cityware: Urban computing to bridge online and real-world social networks. In Marcus Foth, editor, *Handbook of Research on Urban Informatics: The Practice and Promise of the Real-Time City*, chapter XIV. Information Science Reference, IGI Global, 2008.

12. Mathew Laibowitz, Jonathan Gips, Ryan Aylward, Alex Pentland, and Joseph A. Paradiso. A sensor network for social dynamics. In *IPSN '06: Proceedings of the fifth international conference on Information processing in sensor networks*, pages 483–491, New York, NY, USA, 2006. ACM.

13. Joseph F. McCarthy, David W. McDonald, Suzanne Soroczak, David H. Nguyen, and Al M. Rashid. Augmenting the social space of an academic conference. In *CSCW '04: Proceedings of the 2004 ACM conference on Computer supported cooperative work*, pages 39–48, New York, NY, USA, 2004. ACM.

14. Nokia. Sensor. http://www.nokia.com.sg/A4504217, May 2008.

15. Eric Paulos, Elizabeth Goodman, and Elizabeth Goodman. The familiar stranger: anxiety, comfort, and play in public places. In *CHI '04: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 223–230, New York, NY, USA, 2004. ACM.

16. Per Persson, Jan Blom, and Younghee Jung. Digidress: A field trial of an expressive social proximity application. In *UbiComp 2005: Ubiquitous Computing*, volume Volume 3660/2005, pages 195–212. Springer Berlin / Heidelberg, 2005.

17. Åsa Rudström, Martin Svensson, Rickard Cöster, and Kristina Höök. Mobitip: Using bluetooth as a mediator of social context. In *Proceedings of Ubicomp 2004 Adjunct Proceedings*, Nottingham, UK, 2004.

18. Michael Terry, Elizabeth D. Mynatt, Kathy Ryall, and Darren Leigh. Social net: using patterns of physical proximity over time to infer shared interests. In *CHI '02: CHI '02 extended abstracts on Human factors in computing systems*, pages 816–817, New York, NY, USA, 2002. ACM.