The Second Life of a Sensor: Integrating Real-world Experience in Virtual Worlds using Mobile Phones

Mirco Musolesi*, Emiliano Miluzzo*, Nicholas D. Lane*, Shane B. Eisenman†,
Tanzeem Choudhury*, and Andrew T. Campbell*
*Dartmouth College, USA †Columbia University, USA

Abstract

Virtual world simulators like Second Life represent the latest and most successful frontier of online services for entertainment and business. People have virtual lives in these worlds using personal avatars. Bridging real life and these virtual worlds together is challenging, but enables new application scenarios for these systems. Existing work focuses on the representation of *objects* inside virtual worlds. We believe that the overall experience will be improved if we can reproduce not only inanimate and passive objects like buildings and beer steins but also the *subjects* in these virtual environments. In order to have a seamless integration between the physical and virtual worlds, and to maximize the scale of adoption, the system should rely on devices that people already use every day. For this reason, we believe that mobile phones are a natural and ideal conduit.

In this paper, we argue that the sensors embedded in commercial mobile phones can be used to infer real-world activities, that in turn can be reproduced in virtual settings. We discuss the challenges related to the implementation of such a system both from algorithmic and systematic points of view. We present the design of a prototype integrating a platform for inferring activities using mobile phones and Second Life. To the best of our knowledge this is the first time that sensors used on everyday mobile phones reflecting a person's activity in the physical world has been expressed in virtual worlds such as Second Life.

Categories and Subject Descriptors

C.3 [**Communication/Networking and Information Technology**]: Special-Purpose and Application-Based Systems—*Ubiquitous Computing*

General Terms

Design, Measurement, Experimentation, Human Factors

Keywords

Sensor systems, mobile phones, Second Life, virtual worlds, activity recognition

1 Introduction

Virtual world simulators represent one of the latest and most successful frontiers of online entertainment. Among them, Second Life [8] is one of the most popular Internet-based virtual worlds in terms of subscribers with over 500,000 active users. A client program called the Second Life Viewer allows its users (called *residents*) to interact with each other through mobile *avatars*. Inside the game, residents can socialize and participate in individual and group activities. An avatar's appearance may be customized across a wide range of physical attributes. Avatars may be completely invented, without any relation to the physical aspect of the human user, or can be made to closely resemble the human whom they represent. However, in the current system, there is still a divide between the virtual world and the real one. Specifically, a user may be able to recreate his appearance but he can not automatically mirror/translate his current actions and movements in his real life to the virtual life of his avatar.

Our vision is to bridge the current divide between real worlds and virtual worlds, by having real world activities (e.g., sitting, walking, running, cooking) and other aspects of the human represented in real time in the virtual world. Our aim is to go a step further than simply reproducing spaces (e.g., rooms, buildings, public spaces [7, 11]) and the objects with which humans interact. We focus instead on providing a virtual representation of humans, their surroundings as sensed by the humans themselves, and interactions with the members of their social networks (e.g., fetched from external sources such as Facebook [3]).

There are existing virtual worlds that map to the real one with great accuracy in terms of geographical locations and local features. Complex virtual representations of real geographical places exist such as the ones provided by Google Earth [4] and Microsoft Virtual Earth [9]. In Google Earth it is possible to insert 3-D representations of buildings and artifacts by means of Google Sketch. We believe that the next step will be representing real world activities in this kind of virtual setting whereby personal avatars will be able to move inside these artificial spaces following the real positions and activities of people.

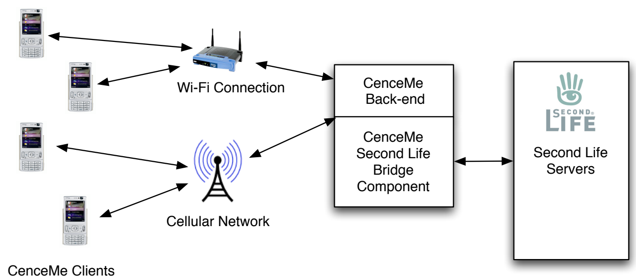

Figure 1. Architecture of the system bridging CenceMe and Second Life

Sensors have been used to support presence in virtual environments since the first experiments on the Virtual Reality Theaters [2]. In most of these systems, individuals carry special wearable devices like helmets and wristbands with embedded sensors measuring their movements. These are special-purpose external devices used to map user's actions. More recently, the integration of sensors and Second Life is discussed as a future work in [7], where the authors propose to reproduce the entire floor of a building using data collected by means of the Plug [7] platform. However, the focus of [7] is on the reproduction of a working space and not on the activities of people moving inside it (i.e, it is focused on places and objects and not on subjects). Moreover, [7] is based on a non-standard hardware platform produced in small quantities and not available to the general public.

Starting from this technological scenario, the contribution of this paper can be summarized as follows:

- We discuss the systems challenges and the design issues in integrating real world people-centric sensor systems and virtual worlds;
- We present a system that is able to reproduce human activities and not only inanimate objects in a virtual world for the first time;
- We discuss the design of a preliminary prototype based on the integration of a system for activity inference based on mobile phones and Second Life.

In Section 2 we present an overview of the main challenges in implementing systems supporting real activity inference algorithms for virtual worlds based on data extracted using mobile phones. In Section 3 we discuss our ongoing work on the CenceMe platform [1] that is the basis for the practical implementation of the proposed system, and present preliminary results that are at the root of the proposed architecture. Section 4 concludes the paper, summarizing the contribution of our work.

2 System Design Challenges

In this section, we discuss the key issues in detecting real-world activities by means of commercial mobile phones and seamlessly integrating this information into virtual worlds environments.

2.1 Activity Recognition and Visualization

Activity inference, mapping and visualization are key aspects of the proposed system. Activity inference consists of the extraction of high-level activities (e.g., sitting, running, walking or talking) from time series of raw data sampled from sensors (e.g., accelerometer, microphone). There is significant interest in the research community in designing high accuracy activity recognition algorithms [6, 12]. Many user-specified visualization policies for inferred activity may exist. Some users might want to map their physical activities in the real world into different virtual ones. For example, a user may want to specify that the action of running corresponds to flying in the virtual world. The other interesting aspect is information accuracy. Some users may prefer to discard the information if the inference is not sufficiently accurate given predefined quantitative thresholds. Alternatively, a default activity (e.g., standing) can be preset, which will be visualized if the system is not able to perform activity recognition with a sufficient accuracy. This process can be based on the definition of specific classifier thresholds on the Receiver Operating Characteristic (ROC) curves [13]. Generally, the accuracy of these measurements is related to the limitations of currently available embedded sensors. We believe activity recognition accuracy will only improve in the future when more powerful sensors become available for use with more effective inference algorithms.

We also consider the possibility of information flow from virtual worlds to the phones. Of particular interest is researching the best methods for reporting and representing in the real world changes and interactions that occur in the virtual world. An example of a very simple interaction is the opening of an internet chat window if avatars start a conversation in Second Life. More complex interaction dynamics can be envisaged and the improvements of the user interfaces of the phones are likely to open new research opportunities.

2.2 Intermittent Connectivity

Cell phones may experience intermittent connectivity due to limits in network coverage. Virtual avatars should be set to a meaningful state (such as the latest known state with a question mark appearing above the avatar) during intervals when no new updates are received. A more complex solution may involve the use of prediction techniques to determine the current state and the location given the sequence of the previous states. Delay tolerant mechanisms can also be adopted to temporarily store user status locally on the phone in case of disconnection. The time series of state data can later be input to the virtual world. Research is needed to determine how to best (if at all) represent such a delayed burst of physical state information in the virtual world. We believe that strict real-time action rendering may be not necessary for most users. However, this is an interesting aspect that should be analyzed in detail for different activities and classes of users by means of experiments and surveys.

2.3 Scalability

Given the potentially high number of users, designing for scalability is important. To reduce the number of status updates flowing into the Second Life visualization, the status/activity of the virtual avatar is updated only when the inferred status/activity of the human user changes. This, in turn, is dependent on how often the activity classifier is executed. By leveraging historical activity traces, we propose to set the classifier execution frequency proportional to the degree of variability of the user's past behavior. A sensor data push model from the mobile phone to the server is used. We believe the mobile phone is best positioned to manage the trade-off between limited local storage, data upload opportunity, and user expectations of application responsiveness.

Therefore, our default approach is to run activity recognition algorithms on the mobile device to reduce communication costs and also to reduce the computational burden on the server. However, we recognize that some classification algorithms may be too computationally intensive for handheld devices and propose to run a portion (possibly all) of the classifier on a back end server in these cases. This may be particularly appropriate during the training phase of a classification model, or when data sourced from multiple users support feature vectors input to the classifier. Low-fidelity classifiers requiring fewer resources can be used on mobile phones in absence of connectivity or to reduce transmission costs.

2.4 Data Sharing and Group Activities

The focus of existing work on sensor-based human activity inference is on the analysis and processing of information collected by devices carried by or attached to a single individual [6, 12]. This inference process exploits only the information extracted by the local sensors. We believe that the system should leverage the sharing of raw or processed data among different people in physical proximity by means of short range radio communication. This mechanism may also allow for a more accurate inference of group activities such as talking or moving together, improving their rendering in the virtual space [14]. Moreover, by means of this mechanism, individuals will be able to update not only their status, but also those of their neighbors.

2.5 External Sensing Devices

In addition to sourcing data from phones, the activity recognition process may be augmented by means of external devices connected to the phones through the Bluetooth radio. These devices may be equipped with additional sensors in order to improve the accuracy of activity classifiers running on the phone, or to extend the set of the actions that can be inferred. Having a small size sensing module external to the phone (e.g., attached to the ankle) allows for more accurate activity inference (e.g., running) compared with using phone-embedded sensors which may be ill-positioned (e.g., in the pocket).

For this purpose, we developed a badge-sized device, the BlueCel, integrating a set of various sensors including accelerometer, light, temperature and Galvanic Skin Response (GSR) sensors. The GSR sensor can be used to infer emotions like stress level. In our vision, the system should not rely on these devices, but on off-the-shelf platforms (i.e., the sensors integrated in the mobile phones). At the same time, it should be able to use them in a seamless and opportunistic manner when available.

2.6 Context Reflection

Virtual worlds are an excellent medium to represent environmental data. There is a growing interest in displaying sensor data using common and popular media such as Web-based portals (e.g., the SensorMap project [10], the BikeNet system [1]). People-centric sensing is an ideal way of collecting data for these systems in real-time. Data can be collected and displayed in virtual worlds such as Second Life in a powerful and captivating manner by changing the appearance of an environment according to the sensed data. For example, high humidity may trigger the visualization of rain or misty conditions. A pre-processing of the raw environmental data may be performed in the back-end. Users may also actively participate in the data collection process through the acquisition of multimedia content such as recordings of sounds and voices, photos and videos to be integrated in the virtual worlds. While these aspects are very promising, they are out of the scope of this work.

2.7 Privacy and Social Implications

To manage their exposure, users should be able to disconnect from the virtual worlds at any time or be able to interrupt the sampling of their activities. The actual positions and the actions of users in the virtual world may be disclosed according to pre-configured user policies (e.g., only to one's listed buddies imported from other websites like Facebook [3]). The problem of location privacy is even more important in this class of systems where personal information is exposed to a potentially large set of people. Additional privacy issues are related to the collection of environmental data by means of people carrying devices and, above all, data collected about them. In [5] the authors discuss important privacy and security challenges related to people-centric sensing.

3 System Prototype Implementation

In this section, we discuss the overall architecture of the system for the integration of activity information into virtual worlds such as Second Life and Map Services such as Google Earth. We first describe ongoing work related to the CenceMe [1] platform, which acts as the core service for the retrieval and the analysis of the sensed data in our architecture.

Bridging CenceMe and Virtual Worlds. We give a brief overview of the CenceMe platform in order to understand the key issues that underpin the proposed architecture. For more detailed information on CenceMe refer to [1]. CenceMe is a general personal sensing system that enables members of social networks to share their sensing presence extracted by means of mobile phones with their buddies. CenceMe is able to provide information about the personal status of its users and their surroundings. Similarly, by mining the sensed data CenceMe can extract patterns and features of importance in a user's life routines, such as the current status in terms of activity (e.g, as sitting, walking, standing, dancing, meeting friends) and physical or logical location (e.g, at the gym, coffee shop, at work or other significant places for the individuals).

We have integrated the CenceMe back-end and the Second Life servers by means of a connector component, developed for the Second Life Linux client, that receives inputs from the CenceMe back-end. The logical architecture of the system is shown in Figure 1. The Second Life Linux client

```
induceAction(list outputMapping) {
  ...
  if (body == MAPPING_D) {
    llStartAnimation(PHY_ACTIVITY_D);
    state = PHY_ACTIVITY_D;
  }
  else if (body == MAPPING_G) {
    llStartAnimation(PHY_ACTIVITY_G);
    state = PHY_ACTIVITY_G;
  }
  ...
  else {
    llStartAnimation(DEFAULT_ACTIVITY);
    state = DEFAULT_ACTIVITY;
  }
}
default {
  timer() {
    http_request_id =
    llHTTPRequest(cme_backend),parameter_list,
    default_request);
  }
  http_response(key request_id, integer status,
  list metadata, string body) {
    outputMapping = parseResponse(body);
    induceAction(outputMapping);
  }
}
```

Figure 2. Code of the LSL script triggering the mapping of the activities

Figure 3. Second Life display of near-real-time inferred activities injected from CenceMe. Three states set by the CenceMe infrastructure are translated according to a user-defined mapping: sitting → yoga-floating (left), standing → standing (center), and running → flying (right).

is available as open-source software [8] as an alpha version release. CenceMe can be used to interconnect virtual worlds and social networking systems such as Facebook that have become extremely popular. Information can be fetched from social network sites and displayed in virtual environments (e.g., presence information about the buddies/groups can be automatically reflected in Second Life). Virtual social events can be also organized between members of the same social networking community. Instead of using basic instant messaging, members may see what their friends and colleagues are doing (playing sports, walking, etc.) and their current geographical position by means of a three-dimensional visualization provided by a virtual environment rendering engine like the one used by Second Life. This may replace the current simple but rather cryptic idle/away/online status icons and text messages.

There is currently no publicly available API to create and move characters in map services using Google Earth or Microsoft Live Search Maps. A proposal for merging Google Earth and Second Life exists [11] as far as inanimate objects are concerned. We plan to integrate CenceMe and these tools as soon as an API is made available.

Design and Preliminary Evaluation of the Prototype. We design a preliminary prototype of a component bridging Second Life and CenceMe to better understand the limitations of our systems design. The integration is based on the exploitation of virtual CenceMe objects that the Second Life avatar can wear or carry. This object alters the avatar animation to reflect the actual activity state of a human user carrying a cell phone that is running the CenceMe sensing application. The CenceMe object is implemented as a standard Second Life object that we program to communicate with the CenceMe infrastructure via XML encoded requests and responses exchanged via HTTP. Second Life CenceMe object behavior is written in LSL (Linden Script Language [8]). The interactions with the CenceMe infrastructure use the same CenceMe external application extension APIs as other application plug-ins (i.e., Facebook, Pidgin) that have been constructed (as described in [1]). Objects acquire a new state by polling the CenceMe infrastructure at a rate that can be set by users (within ranges defined by CenceMe administrators). The avatar can disconnect from CenceMe by taking off the object.

The interface between the CenceMe system and Second Life largely uses off-the-shelf technologies. Java servlets hosted within a Tomcat server offer a set of APIs made available via XML-RPC for retrieving information about user activities from a MySQL database. The communication paradigm between the Second Life and CenceMe platforms that we adopt is pull-based, i.e., periodic requests are issued by a script running in the Second Life platform to the CenceMe server. Our implementation uses this API to both retrieve inferred activities about CenceMe users as well as a specification that defines how these inferences are to be represented. A library of potential Second Life activities is maintained (e.g., walking, teleporting, yoga-floating) from which the user may choose to map specific real-world activities. These relationships are imported and applied to all incoming data. Real actions can be mapped to different actions in the virtual world (e.g., running mapped to flying).

Each virtual object within Second Life retrieves data independently from the CenceMe server. Each object is implemented using the LSL language. LSL can be used to specify object behavior within the Second Life environment. The code is compiled by Second Life servers into bytecode with a maximum memory usage limit of 16KB. The bytecode is executed by virtual machines running on these servers. LSL is an event oriented language and it is suited to implement state machine style solutions. In our case, it is used to implement reactions to changes in real world activities by triggering modifications of the activities displayed in the virtual world.

In Figure 2, we show a fragment of the LSL implementation of the objects. The flow of execution is as follows:

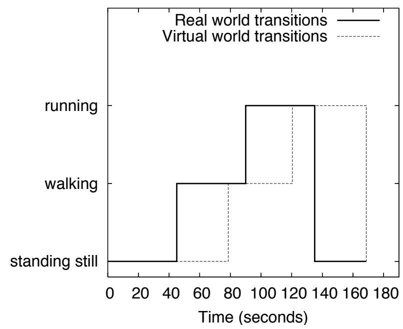

Figure 4. Delay Measurements

once the object is attached to the avatar, a periodic timer is initiated. In our experiment, the period of the timer was 0.5 seconds. The body of the timer causes the HTTP request to be made (see the `timer()` body in Figure 2). An XML-RPC based request for both the latest inferred action and the user defined mapping from CenceMe inferred activities to Second Life actions is sent. An event handler for the returning response (see the `http_response()` block) first calls `parseResponse()` which processes the incoming response, returning a list of potentially invoked avatar actions. Then, this output is provided to a second function, `induceAction()` which takes these responses and determines which avatar action should be displayed.

As described above, we use a clothing metaphor to provide an intuitive way for people to configure and control the mapping of their actions. In Figure 3, we show a number of screenshots of an avatar that carries a CenceMe virtual object (in this case, a pager - see inset). In the figures, we show three states set by the CenceMe infrastructure and translated according to a user-defined mapping: sitting → yoga-floating (left), standing → standing (center), and running → flying (right). Details about the activity classifiers can be found in [1]. Alternative CenceMe clothing and accessories include significant places [1] t-shirts that display the logo graphic representing a location of significance in the user's life (i.e., the home, gym, pizzeria, etc.), and virtual mood rings that display the inferred emotional state of the individual. We also present some preliminary performance results of the integrated system. In Figure 4, we show the average delay experienced between performing the activity in the real world and displaying the inferred activities in Second Life. The experiment consists of a sequence of activities performed by users in repeated sessions at different times of the day (at noon, 4pm, and 8pm ET). We average the data over 15 sessions. The sequence of activities is standing still → walking → running → standing still. The transitions between states are reported as step curves. We observe an average delay of about 30 seconds in reporting the activity changes. This delay is due to the activity recognition process on the phones, the transmission of the inferred activity to the back-end, its retrieval from CenceMe database, and its display in Second Life. The delay is mostly caused by the periodic inference process running on the phones and the uploading of this information to the CenceMe back-end. The delay due to the transmission from the CenceMe back-end to the Second Life servers, and the display of the activity information in Second Life is negligible.

We are augmenting our current implementation to: investigate controlling movement in the geographical space; define appropriate privacy policies; and develop mechanisms to address periods when the avatar becomes disconnected from the physical user, or when inferred user state accuracy falls below an acceptable bound.

4 Conclusion

In this paper, we have proposed to bridge real and virtual worlds using commercial mobile phones equipped with standard sensors. We have discussed a number of key issues that need to be tackled to realize our vision. We presented a preliminary implementation of this architecture, demonstrating a live system integration of CenceMe and Second Life. To the best of our knowledge, this is the first time that sensors used on everyday mobile phones reflecting a person's activity in the physical world has been expressed in Second Life.

Acknowledgment

This work is supported in part by Intel Corp., Nokia, NSF NCS-0631289, ARO W911NF-04-1-0311, and the Institute for Security Technology Studies (ISTS) at Dartmouth College. ISTS support is provided by the U.S. Department of Homeland Security under award 2006-CS-001-000001, and by award 0NANB6D6130 from the U.S. Department of Commerce. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of any supporting agency.

5 References

[1] A. T. Campbell, S. B. Eisenman, N. D. Lane, E. Miluzzo, R. Peterson, H. Lu, X. Zheng, M. Musolesi, K. Fodor, and G.-S. Ahn. The rise of people-centric sensing. *IEEE Internet Computing Special Issue on Mesh Networks*, June/July 2008.

[2] C. Cruz-Neira, D. J. Sandin, and T. A. DeFanti. Surround-screen projection-based virtual reality: the design and implementation of the CAVE. In *Proceedings of SIGGRAPH '93*, pages 135–142, New York, NY, USA, 1993. ACM Press.

[3] Facebook. Facebook Website. http://www.facebook.com.

[4] Google. Google Sketch. http://sketchup.google.com/.

[5] P. C. Johnson, A. Kapadia, D. Kotz, and N. Triandopoulos. People-centric urban sensing: Security challenges for the new paradigm. Technical Report TR2007-586, Dartmouth College, Department of Computer Science, February 2007.

[6] J. Lester, T. Choudhury, and G. Borriello. A practical approach to recognizing physical activities. In *Proceedings of PERVASIVE'06*, May 2006.

[7] J. Lifton, M. Feldmeier, Y. Ono, C. Lewis, and J. A. Paradiso. A platform for ubiquitous sensor deployment in occupational and domestic environments. In *Proceedings of IPSN'07*, April 2007.

[8] Linden Research Inc. Second Life. http://secondlife.com/.

[9] Microsoft. Microsoft Virtual Earth. http://www.microsoft.com/virtualearth/default.aspx.

[10] S. Nath, J. Liu, and F. Zhao. SensorMap for Wide-Area Sensor Webs. *Computer*, 40(7):90–93, 2007.

[11] W. Roush. Second Earth. *Technology Review*, July/August 2007.

[12] S. Wang, W. Pentney, A.-M. Popescu, T. Choudhury, and M. Philipose. Common Sense Based Joint Training of Human Activity Recognizer. In *Proceedings of IJCAI'07*, January 2007.

[13] I. H. Witten and E. Frank. *Data Mining*. Morgan Kaufmann Publishers, 2005.

[14] D. Wyatt, T. Choudhury, and J. Blimes. Conversation Detection and Speaker Segmentation in Privacy Sensitive Situated Speech Data. In *Proceedings of Interspeech 2007*, August 2007.