OmniSense: A Collaborative Sensing Framework for User Context Recognition Using Mobile Phones

Heng-Tze Cheng, Senaka Buthpitiya, Feng-Tso Sun, Martin Griss
Carnegie Mellon Silicon Valley
{hengtze.cheng, senaka.buthpitiya, lucas.sun, martin.griss}@sv.cmu.edu

1. INTRODUCTION

Context information, including a user's locations and activities, is indispensable for context-aware applications such as targeted advertising and disaster response. Inferring user context from sensor data is intrinsically challenging due to the semantic gap between low-level signals and high-level human activities. When implemented on mobile phones, more challenges on resource limitations are present. While most existing work focuses on context recognition using a single mobile phone [1], collaboration among multiple phones has received little attention, and the recognition accuracy is susceptible to phone position and ambient changes. Simply putting a phone in one's pocket can render the microphone muffled and the camera useless. Furthermore, naïve statistical learning methods used in prior work are insufficient to model the relationship between locations and activities.

In light of the existing challenges and inspired by the idea of collaborative sensing for social event recording in VUPoint [2], we propose *OmniSense*, a new collaborative mobile sensing framework that combines correlated sensor data from phones in proximity for robust context recognition. The contributions are threefold: (1) Using multiple phones implies concurrent sensing of the same environment from different positions, thus increasing the probability of getting more useful sensor data and becoming less susceptible to ambient changes. (2) Since OmniSense combines sensed data from different phones, phones with different sensing capability can complement each other. For example, a phone without GPS can benefit from the position information provided by nearby phones. (3) In addition to the multi-phone framework, we also design a coupled hidden Markov model for context recognition. We use the model to learn the temporal dependency of user locations and activities, and adjust the multimodal feature weighting most suitable for the context type.

2. SYSTEM ARCHITECTURE

OmniSense consists of two stages: (1) *Group Sensing Stage* that groups nearby phones and extract features from sensor data, and (2) *Context Recognition Stage* that infer the user's context based on the sensor data and the pre-trained model. The sensing application is implemented on Nokia N95 phones using Python.

2.1 Group Sensing Stage

The first step of collaborative sensing is proximity detection. First, a phone performs a Bluetooth scan and transmits a list of Bluetooth addresses of discovered devices to the server, which clusters two devices in the same group if both of them appear in each other's list. Since the range of Bluetooth is roughly 10 meters, it is suitable for forming a group in a typical-size room. The reason why we do not use Bluetooth pairing is the concern of intrusiveness. If a mobile phone prompts a user to pair with another devices every time a new device is found, it would be highly intrusive and thus undesirable.

However, if a huge amount of phones are using the service simultaneously, it is impractical to search through all the phones and make grouping decisions. To address the scalability problem, we design a hierarchical clustering scheme. First, the server clusters phones according to their latest available GPS readings.

HotMobile '10, Feb. 22-23, 2010, Annapolis, M.D.

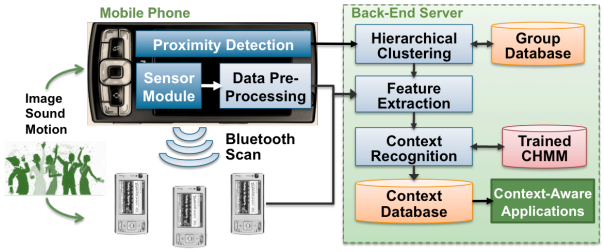

Figure 1: The system architecture of OmniSense.

Then, the group is further partitioned into subgroups using WiFi APs seen by the phones. Finally, Bluetooth addresses seen by the phones are used for fine-grained grouping. After grouping, we extract features from the sensor data of microphones, cameras, and accelerometers. For audio, we extract short-time energy, zero-crossing rate and Mel-frequency cepstral coefficients. For images and accelerometer readings, we adopt the features used in [1].

2.2 Context Recognition Stage

It has been shown that human trajectories exhibit a high degree of temporal and spatial regularity. Therefore, we propose the use of a coupled hidden Markov model (CHMM) [3] to jointly model activity and location sequences. We use the Viterbi algorithm to train different Gaussian mixture models (GMMs) to fit different types of context. For the ith context stream (e.g. location stream or activity stream) at time t, among all the possible states $s_{i,t}$ (each corresponds to a specific location or activity), the optimal hidden state $s_{i,t}^{*}$ maximum likelihood is formulated as

$$s_{i,t}^{*} = \arg\max_{s_{i,t}} (\alpha \log \sum_{k=1}^{K} w_{i,k} N(\mathbf{O}_{i,t}, \mathbf{m}_{i,k}, \mathbf{C}_{i,k}) + \beta \log P_{trans}) \quad (1)$$

where $P_{trans} = P(s_{i,t} | s_{1,t-1}, s_{2,t-1}, ..., s_{k,t-1})$. The intuition behind eq. (1) is that a user's activity can be predicted using both the sensed data (e.g. sound and movement) and the user's previous activity or location. In eq. (1), α and β represent the weighting on observation probability and transition probability, respectively, which are empirically determined. $\mathbf{O}_{i,t}$ in eq. (2) denotes the observed vector of the i^{th} stream at time t. For the i^{th} stream, $\mathbf{m}_{i,k}$, $\mathbf{C}_{i,k}$, and $w_{i,k}$ are the mean matrix, covariance matrix, and weighting of the k^{th} Gaussian mixture, respectively. K is the number of mixtures used in the GMM. The resulting $s_{i,t}^{*}$ is the mutual context of the group. In experiments, we optimize the parameters empirically via a separate validation set.

3. APPLICATIONS

We are developing a context-based group advertising system based on the mutual context and interests of a social group rather than individuals in traditional approach. OmniSense also enables ubiquitous security surveillance and disaster response since audio-visual data sensed from different phones are integrated for analysis. As smartphones become pervasive, we believe more applications can be built on this multi-phone framework.

4. REFERENCES

[1] M. Azizyan, I. Constandache, R. R. Choudhury. SurroundSense: Mobile Phone Localization via Ambience Fingerprinting. In *Proc. Int'l Conf. Mobile Computing and Networking,* pp. 261–272, 2009.

[2] X. Bao, R. R. Choudhury. VUPoints: Collaborative Sensing and Video Recording through Mobile Phones. In *Proc. Workshop on Networking, Systems, Applications on Mobile Handhelds*, pp. 7–12, 2009.

[3] A. V. Nefian, et al. A coupled HMM for audio-visual speech recognition. In Proc. *ICASSP*, pp. 2013–2016, 2002.