SensCare: Semi-automatic Activity Summarization System for Elderly Care

Pang Wu, Huan-Kai Peng, Jiang Zhu, and Ying Zhang

Carnegie Mellon University, Moffett Field, CA 94035, USA
{pang.wu,huankai.peng,jiang.zhu,joy.zhang}@sv.cmu.edu

Abstract. The fast growing mobile sensor technology makes sensor-based lifelogging system attractive to the remote elderly care. However, existing lifelogging systems are weak at generating meaningful activity summaries from heterogeneous sensor data which significantly limits the usability of lifelogging systems in practice. In this paper, we introduce SensCare, a semi-automatic lifelog summarization system for elderly care. From various sensor information collected from mobile phones carried by elderlies, SensCare fuses the heterogeneous sensor information and automatically segments/recognizes user's daily activities in a hierarchical way. With a few human annotations, SensCare generates summaries of data collected from activties performed by the elderly. SensCare addresses three challenges in sensor-based elderly care systems: the rarity of activity labels, the uncertainty of activity granularities, and the difficulty of multi-dimensional sensor fusion. We conduct a set of experiments with users carrying a smart phone for multiple days and evaluate the effectiveness of the automatic summary. With proper sensor configuration, the phone can continue to monitor user's activities for more than 24 hours without charging. SensCare also demonstrates that unsupervised hierarchical activity segmentation and semi-automatic summarization can be achieved with reasonably good accuracy (average F1 score 0.65) and the system is very useful for users to recall what has happened in their daily lives.

Keywords: Sensor-based Elderly Care, Structured Activity Recognition, Activity Summarization, Lifelog.

1 Introduction

In year 2020, 71 million Americans will be officially considered as elderly (65 years or older) according to the projection of US Census Bureau. This accounts for 20% of the nation's total population [17], a staggering increase from 10% in 2010. Numbers from [17] also shows that about 80% of aging adults have one chronic condition, and 50% have at least two. With such chronic conditions, most elderlies require some levels of care to assist their daily living. Statistics also shows that there are fewer young and middle-aged adults living with their parents and provide care to them. The situation is even worse in rural area where low population density and large catchment areas combine with lack of service access and reimbursement in creating barriers to community-based elder care [4].

J.Y. Zhang et al. (Eds.): MobiCASE 2011, LNICST 95, pp. 1–19, 2012.

Aging-in-place [1] has become increasingly popular in recent years as many elderly prefer to age in their own houses instead of relocating to nursing homes. Aging-in-place has the advantage that senior or elderly person can continue to live in their own surroundings without making a drastic change and maintain their valuable social networks. The success of aging-in-place depends on the telecare and other assistive technologies where elderly can access the care services remotely and promptly when needed. The fast development in mobile sensing technology and remote healthcare technologies has started to bridge the gap between the needs of age-in-place and current healthcare system where doctors and caregivers have to physically meet the patient for any checkups and diagnosis.

In this paper, we describe SensCare, a semi-automatic system that provides solutions to an important scenario in elderly care: automatic generation of meaningful activity summaries from mobile lifelogs. SensCare aims at reducing human annotation effort in activity summarization. With mobile sensors constantly logging the elderly users' daily activity, SensCare automatically recognizes the daily activity of the user and generates a meaningful hierarchical summary. An effective summary can help the elderly better understand his/her behavior so as to improve his/her well-being. It also helps doctors to diagnose causes of an elderly's medical conditions and allows remote family members to keep up with the life of the elderly. For instance, a senior suffered with insomnia carries a mobile phone with him constantly. He puts the phone under the pillow when he goes to bed. The accelerometer embedded in the phone records the motion in the night which reflects his sleeping quality. Such an automatic summary is valuable since the elderly himself cannot provide accurate quantitative answers. This summarized data, such as the average sleeping hours in the past week, wake up frequencies etc. can be accessed by the remote caregiver to evaluate the effectiveness of medicines the doctor has prescribed and by family members remotely to see how the elderly is doing.

The rest of paper is organized as follows: Section 2 presents a hypothetical before-and-after scenario to illustrate how the SensCare system would be used. We describe the system design in Section 5. This section also discusses potential limitations of the system. In Section 6, we present experimental result with data collected over 5 days and compare algorithms proposed in SensCare with other methods. In Section 7 and 8, we will go through the related works and summarize our works and guide to the future challenges of the system.

2 Scenario for Semi-automatic Activity Summarization

Many elderly hypertension patients are recommended to exercises regularly to control their high blood pressure in addition to a heart-friendly diet and taking drugs at regular basis. To qualitatively evaluate physical exercises performed by an elderly, doctors usually ask them questions during their office visits such as "Which exercises do you perform on a daily basis?" "How long does it last?" etc. In many cases, elderly patients cannot remember detailed information and can only provide rough answers like "I walk to office everyday and it takes about 20 to 30 minutes." or "Well, I went hiking with my family last weekend and had spent a whole afternoon there." Yet, "20 to 30 minutes" is too rough and it is unclear as to how many hours in the "whole afternoon" did the elderly patient actually hike . The unreliable data leads to unreliable assessment by doctors.

With SensCare (Figure 1), elderly users carry mobile devices constantly to collect data about their daily activities. The collected data is uploaded to the cloud and SensCare will automatically segment the data into a hierarchy of activities. Users can view this hierarchical segmentation in a calendar layout on their PC or tablet devices. They can annotate some activities such as "walk in the park" on the calendar. Users' annotations are also uploaded to the cloud (a backend system) for the system to propagate the annotation to all similar activities in the lifelog. After users annotate a few instances of their "useful" activities such as "walking", "hiking" and "driving", SensCare will recognize these activities and label them automatically in the future. SensCare segments the lifelog in an unsupervised manner, so users only need to annotate activities of interest to them.

SensCare gives caregivers more accurate information of users' daily activities and when combined with other sensor information such as the blood pressure measured by the in-home device, the doctor can come up with a better plan to improve the elderly's lifestyle in order to help his hypertension condition.

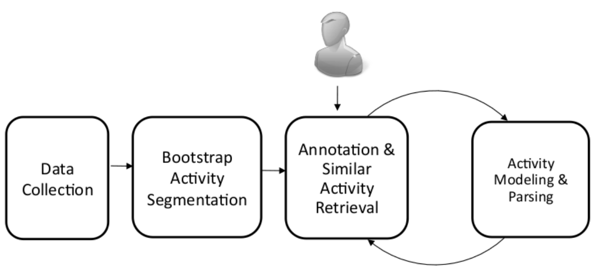


Fig. 1. The system workflow of SensCare

3 Activity Summarization through Unsupervised Hierarchical Activity Segmentation and Recognition

The purpose of activity summarization in SensCare is to automatically generate a list of activities that a user did by the end of a day when all sensors' data is uploaded. It is more useful if the summarization is hierarchical, i.e., users can zoom in a high-level activity, e.g., "played tennis from 9am to 10:30am", and see more fine-detailed activities such as "warm up, 9am to 9:05am", "tennis game: 9:10 to 9:30", "break: 9:30 to 9:40", "tennis game: 9:40 to 10:15".

Supervised activity recognition approaches can recognize a few activities specified in the labeled training data. However, labelled training data is very expensive to collect. It is also infeasible to predict what kind of activities users might be interested in the future. Supervised activity recognition is unlikely to answer questions like "What costs me most of the time this morning" or "How many hours did I spend on the road to grocery stores last month?", unless the system is trained with labeled data to address these questions.

Although it is impractical to enumerate all activities, we can detect the boundaries for different activities without any training data and then categorize similar activities, which will provide part of the semantic information. In SensCare, activity summarization is achieved through the combination unsupervised activity segmentation and activity recognition.

When all sensor data is uploaded, SensCare first quantizes each sensor readings to discrete labels called *behavior text*. This text-like symbolic representation allows the SensCare system to use some well-established algorithms from statistical natural language processing field such as Information Retrieval (IR) and Text Summarization to process the sensor data. We also apply heterogeneous sensor fusion (Section 4) to fuse multi-dimensional sensor data such as information from accelerometers and GPS into a single dimensional behavior text string for the convenience of handling multiple sensor input. The unsupervised hierarchical segmentation segments the input behavior text string into a hierarchy of shorter segments, each, hopefully corresponds to a meaningful activity at different granularities. With annotations provided by the user on some instances of the past activities, SensCare can recognize those activities similar to the labeled ones and assign meaningful labels to them, e.g., "play tennis".

In SensCare, we developed two methods for unsupervised activity segmentation: top-down segmentation by detecting activity change and the smoothed Hidden Markov Model (HMM).

3.1 Top-Down Activity Segmentation through Activity Change Detection

The underlining assumption of this segmentation algorithm is that when a user switches his/her activity at time t, there should be a significant change from behavior text string $[t-w, t-1]$ to string $[t, t+w]$. For a window size w, define the "change of activity" at time t as:

$$H(t, w) = -logS([t-w, t-1], [t, t+w-1]), \tag{1}$$

where $S(P, Q)$ (Eq. 3) measures the similarity between behavior text string P and Q.

The higher the value of $H(t, w)$, the more likely is the user to have changed his/her activity at time t. Figure 2 shows an example of H values at each data point given different window sizes for a segment of behavior text.

Notice that: (1) peaks of activity change identified by larger windows are also peaks identified by smaller windows but not vice versa; and (2) activity changes

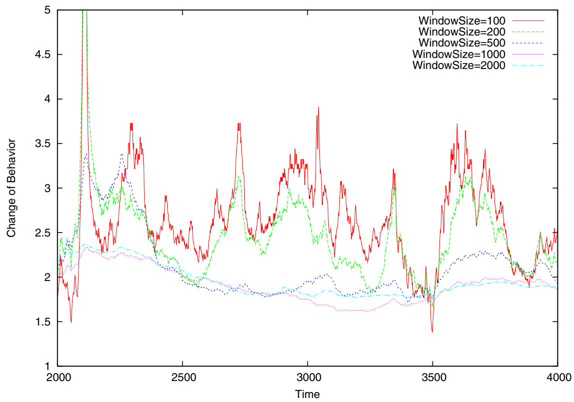


Fig. 2. Activity changes calculated by different sizes of sliding windows

over larger windows are smoother than smaller windows. Intuitively, larger window size captures changes of larger-scale activities whereas smaller window captures changes of smaller activities. Based on these finding, we first segment the lifelog data using large window sizes and then recursively segment the data using smaller windows. This results in a hierarchical segmentation of lifelogs which allows the user to efficiently browse through the lifelog and annotate activities of his/her interests.

3.2 Smoothed HMM Segmentation

Hidden Markov Model (HMM) has been widely used in text string segmentation and labeling. After specifying the topology of a Markov model, an HMM can be trained on unlabeled sequences of text through the Baum-Welch algorithm [3]. which estimates model parameters such as the probability of emitting a symbol from a certain state and the transition probability from one state to another. A trained HMM can then be used to "parse" a sequence of text and estimate the most likely "states" sequence (e.g., "S1 S1 S1 S2 S2 S1 S1 S3 ...") that generates the observed text. "state" is then used as the label for each observed symbols or in our case, the underlying activity for the observed sensor readings. When the state labels changes, we consider the underlying activity has changed and segment the data to reflect this change. For example, segmenting the data as [S1 S1 S2] [S2 S2] [S1 S1] [S3 ....

In our implementation, each state in the HMM emits single behavior text symbols. This leads to a problem where HMM segments the input data into too many activities. To smooth out these noise, we apply a sliding window of size $2w$ over the recognized state sequence. At time t, we use the dominant activity symbols within the window $[t - w, t + w]$ as the smoothed activity symbol for time t and segment the sequence to activities over the smoothed activity/state symbols.

3.3 Activity Recognition

The purpose of activity recognition is to assign semantic meanings to the segmented activities. In SensCare users can annotate some instances of activities after the automatic segmentation. For unlabeled activities, SensCare will search through the labeled activities and assign the label of the most "similar" labeled activity to the unlabeled ones. As all sensor data has been converted to a single dimension behavior text, similarity between two activities can be calculated by the distance of their corresponding behavior text strings.

Inspired by the BLEU metric [20] where averaged n-gram precision is used to measure the similarity between a machine translation hypothesis and human generated reference translations, we use *averaged n-gram precision* to estimate the similarity between two lifelog segments.

Assuming that P and Q are two activity language sentences of the same length l. P is the sequence of P_1, P_2, ..., P_L and Q is the sequence of Q_1, Q_2, ..., Q_L. Denote the *similarity* between P and Q as $S(P, Q)$. Define the n-gram precision between P and Q as $\text{Prec}_n(P, Q) =$

$$\frac{\sum_{\tilde{p} \in \{\text{All } n\text{-gram types in } P\}} \min(freq(\tilde{p}, P), freq(\tilde{p}, Q))}{\sum_{\tilde{p} \in \{\text{All } n\text{-gram types in } P\}} freq(\tilde{p}, P)}, \tag{2}$$

and the similarity between P and Q is defined as:

$$S(P, Q) = \frac{1}{N} \sum_{n=1}^{N} \text{Prec}_n(P, Q) \tag{3}$$

$\text{Prec}_n(P, Q)$ calculates the percentage of n-grams in P that can also be found in Q and $S(P, Q)$ averages the precision over 1-gram, 2-gram and up to N-gram. In our experiments, we empirically set $N = 5$.

4 Heterogeneous Sensor Fusion

Different types of sensors capture different aspects of users' activity. Combining information from multiple sensors helps to disambiguate activities that are similar to each other on certain aspects. Activities such as *"Driving"* and *"Eating"* are very similar to each other if we only look at their accelerometer readings. If we include the location/speed information, these two activities can be well distinguished.

Different sensors have different data representation and semantic meanings. Most sensor fusion approaches first classify single sensor readings and then combine classification results from multiple sensors to derive high-level meanings. Other approaches [15] directly concatenate features from different sensors and train activity classifier using the combined feature vectors. Both approaches work for certain applications but lack flexibility. For Multiple Sensor Fusion techniques, the problem typically has a "tracking" nature: given a set of interested quality and input sensors, they fuse the sensors to better model the interested quality. Thus, these approaches are inherently application specific, depending on the mapping relationship between the input sensors and the interested quality.

Based on the behavior text representation, we propose to fuse heterogeneous sensor information in a more principled way where sensor readings are concatenated as a new unit to reduce the conditional entropy of the whole data set for a certain task. In this paper, we use GPS and accelerometer readings as example to illustrate our sensor fusion approach.

After quantizing the GPS coordinates to behavior text, we calculate the mutual information between each accelerometer label and GPS labels. If they are not correlated, i.e, with low mutual information value, we concatenate the two together into a new label, otherwise we drop the GPS label when the mutual information is high. The intuition is that when two types of sensor highly correlate to each other such as the "running" motion always occur in the gym, then there is no need to use both information to represent this activity, whereas "sitting" motion occur with many locations: living room, study room, cafe etc., we need to combine these two information to distinguish different ways of "sitting" which can be "sit on coach and watch TV in the living room" or "sit in front of the laptop and surf the web" or "sit in a cafe and chat with friends."

Mutual information is defined as:

$$I(A;G) = H(A) - H(A|G) \tag{4}$$

where $H(A)$ is the marginal entropy of label A in accelerometer text, $H(A|G)$ is the conditional entropy of label A and G. The conceptual process is illustrated in following figure:

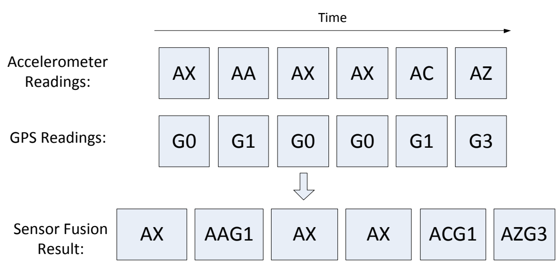


Fig. 3. Conceptual sensor fusion process

5 SensCare System

SensCare is based on a non-intrusive mobile lifelogger system designed using a blackboard architecture [22]. SensCare consists of two groups of components: the *blackboard* which acts as a centralized context data storage and message router, and *satellite applications* that provides or consumes context data, or perform system reconfiguration (Figure 4).

As shown in figure 4, the conceptual blackboard consists with three major components: event controller, raw data storage and activity data storage.

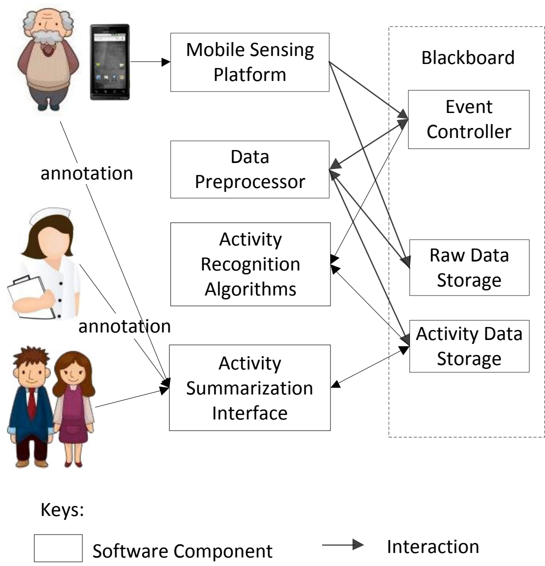


Fig. 4. System architecture

The *raw data storage* stores the unprocessed context data received from sensing platform. The *activity data storage* saves intermediate data generated by activity recognition applications or user's annotation results. The third component, *event controller*, acts as a message router and controller of the system. It receives events from one application then dispatches it to another if there is an application subscribing to that message.

Components outside the blackboard falls into two categories: context/event provider and consumer. A typical context provider is a mobile sensing platform, which is carried by the user and acts as the raw data source of the system. Some of the components are in both classes, for instance, the activity summarization interface will both consume the preprocessed data and provide user annotation to activity data storage. The data preprocessor and the activity recognition algorithm work together in an activity recognition loop and provide initial segmentation for users' annotation. Activity indexing and similar activity retrieval applications are helpers to reduce the human effort in activity summarization (annotation) process. After all, activity data will be displayed on the summarization interface to key parties in healthcare network, including the elderly, his family members and caregivers. The following sections will provide more details of these key components.

5.1 Smart Phones as Mobile Sensing Platform

There are two typical approaches of traditional sensor-based monitoring system: "Smart Home" [21] and wearable sensing platforms.

Smart Homes have sensors mounted on walls and ceilings or have them built in furnitures and appliances. With a rich set of sensors, this approach can collect a wide varieties of lifelog data for further analysis. However, deploying such a system is expensive.

Wearable sensing platforms [11] attach low-cost-light-weight sensors such as EEG and ECG to the human body to measure users' body information like brainwave and heart-rate. However, most of these devices are invasive and uncomfortable to wear [16].

For elderly healthcare, we believe three aspects of the mobile sensing platform are important to the success of sensor-based elderly care systems:

- Mobility. The sensing platform will be embedded in devices that users can naturally carry at daily basis. The device should be lightweight and provide a user friendly interface to configure. The power should last at least one day without charging with sensors activated.
- Sensing capability. The sensing platform should have sufficient types of sensors to capture different aspects of a patient's activity.Many mobile sensing platforms in the market has only one or two types of sensors (e.g., accelerometer only FitBit). As shown by our own work and by others(e.g., [14] and [12]) using multiple types of sensors usually leads to much higher activity recognition accuracy.
- User acceptance. The sensing device should not significantly disturb users' normal life. Comfortableness, appearance and privacy are important for users to accept the technology. For most elderly users, wearing an uncomfortable device over 8 hours per day will make them rejecting the technology. Similarly, wearing several wired sensors around the body or carrying a sensing platform with camera might lead to embarrassment in social events.

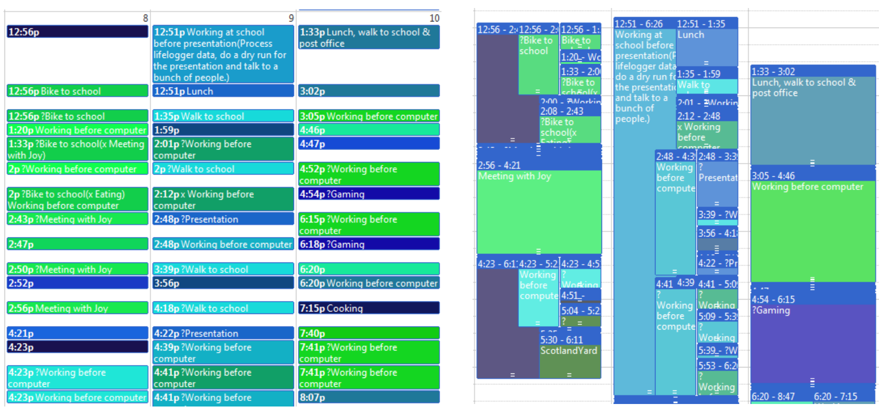
Based on these criteria, we choose smart phones as the mobile sensing platform for SensCare. Smart phones are affordable to most users. They do not need any special deployment or configuration to get their embedded sensors work. Smart phones have built-in network connections for data transmission and they usually have programming interfaces for developer to write new applications. Though it would be ideal to extend the sensing capability with other wearable sensors such as EEG or ECG and transmit the data to smart phones via bluetooth.

We developed the mobile sensing client on an Android phone. The mobile sensing client records the following information: 1) 3-axis accelerometer for motion, 2) Magnetometer for the azimuth value of phone's heading direction. 3) GPS coordinates for outdoor locations, 4) Microphone recordings for sound (we only use sound information for ground truth annotation in this paper), 5) Rotation matrix of the phone derived from the accelerometer reading and the g-value[1], 6) Ambient light , 7) Temperature sensor for environment temperature, and 8) WiFi signal strength for indoor location. All these sensory data are recorded with their sampling time-stamps. The data is saved in CSV format and is transmitted to the raw context data storage on the server side using the HTTP protocol via wireless connections.

[1] We used Motorola Droids which do not have gyroscope sensors. The orientation of the phone is estimated by the Android SDK based on the accelerometer readings and the gravity g-value.

5.2 Interactive Activity Summarization Interface

After the data is processed on the server side, SensCare displays all segmented/recognized activities on the web using a personal calendar system. The user can browse through the calendar, select and annotate automatically identified activities, or create/identify a new activity when the automatic activity segmentation fails to do so. Users can also view the hierarchical structure of their daily activities (Figure 5), which will provide them more choices on granularity for activity annotation as well. A color code is assigned to each activity so the user can easily distinguish between different activities. To better visualize activities by their semantic meanings, we use the same color for similar activities on the calendar. Once the user annotates one activity on the calendar, the description of the activity, e.g., "playing tennis", will be propagated such that all similar activities will all be labeled as "play tennis."



(a)Timeline View

(b) Hierarchical Structured View

Fig. 5. Personal calendar based timeline under different views

6 Experiments and Evaluation

6.1 Experiments Setup

The sensing client runs on a *Motorola Droid*. Table 1 shows sensors used and their sampling rates. Although video and audio data should help, we decided not to use them in SensCare for three reasons. First of all, the media stream will create a large data storage and transmission overhead, which greatly reduces the battery life. Another reason is privacy concern, especially for recording video. In addition, we realized that capturing video on Android requires the application stays on foreground, which will prevent the user using other functionalities of the phone.

We collected 36 hours of real life data in 5 continue weekdays from two graduate students to verify and analyze the system[2]. The data collection process lasts

[2] We are planning to collect data from elderly participants to study more realistic data for elderly care.

Table 1. Sensors on the mobile client and their sampling rate

Sensor	Sampling Rate
Accelerometer, magnetometer	
Ambient light, temperature	20Hz (every 50 ms)
Microphone	8KHz
Camera	
GPS, WiFi	every 2 minutes

about 7 hours each day. To simulate the lifestyle of an elderly, users are asked not to perform strenuous physical activities. Table 2 is a summary of activities done by the user.

User carried two phones during the data collection stage (Figure 6). One was tied to the user's right arm and another phone was used in the normal way: most of the time the phone was in user's pocket and from time to time the user took it out to make phone calls or check emails.

By the end of each day, the user will use the web interface to annotate his activities during the day in three settings:

Fig. 6. Two phone positions in experiments

- Based on the unsupervised segmentation results, label each identified activity. User will based on his memory and the coloring of the activity to assign the "meaning" of the identified activity.
- Without looking at the unsupervised segmentation, the user listens to the recorded audio and creates from scratch his daily activity summary on the calendar. This segmentation/annotation is used as the ground truth in our experiments.

The goal of the experiments is to evaluate 1) whether the automatic activity segmentation matches the ground truth, and 2) whether the similar activity coloring scheme and the automatic activity recognition through similar activity label propagation helps the user to recall what has happened before.

Table 2. Activity instance count and their average time

Activity	**Instance Count**	**Avg. Time per Instance (minutes)**
Walk	7	50.29
Working on Computer	15	66.07
Cooking	4	19.75
Eating	9	20.78
Washing Dishes	2	4.5
Cycling	2	21.5
Video Gaming	2	47.5
Presentation	2	29
Having Class	2	79
Meeting	2	69.5
Talking to Somebody	5	15
Driving	3	8.67
Printing Paperwork	1	44

6.2 Evaluation Metric

We evaluate the accuracy of the activity segmentation and recognition by calculating the F-Score of user's annotation with the ground truth.

Each identified activity in system's annotation A is a triple of $< s, t, l >$ where s is the starting time of the activity, e is the ending time and l is the label of the activity such as "walking". Similarly, we can represent each activity in the ground truth G also as a triple of $< s, t, l >$.

For each activity A in the system's annotation, if there exists a ground truth activity G such that $A_l = G_l$, i.e., the two activities have the same label, and $A_s = G_s \pm \Delta$ $A_e = G_e \pm \Delta$ where two activities have roughly the same starting time and ending time within the allowed margin Δ, then we consider a matches the ground truth activity g and is a true positive (Figure 7). With the precision and recall value calculated for each activity type, we can estimate the harmonic mean and report the F1 score. High F1 scores indicate the system's segmentation/label matches the ground truth.

6.3 Impact of Phone Position in Activity Recognition

We first compare whether the position of the phone has any impact on the activity recognition. Figure 8 shows that system performs better when the mobile phone is attached to users right arm. But for some specific events, like cycling and walking, right arm setting performs worse than pocket setting. It makes sense because arms are relatively stable while riding the bike yet legs are moving more frequent and regularly. In the case of walking, the motion pattern of arms is not as unique as the motion pattern of legs. For events that are closely related to hand movements, like cooking, the right arm setting performances better.

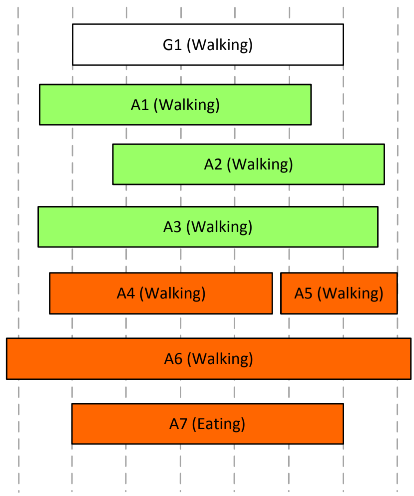


Fig. 7. Ground truth labeled as G_1 with three true annotations(A_1-A_3) and four false annotations(A_4-A_7)

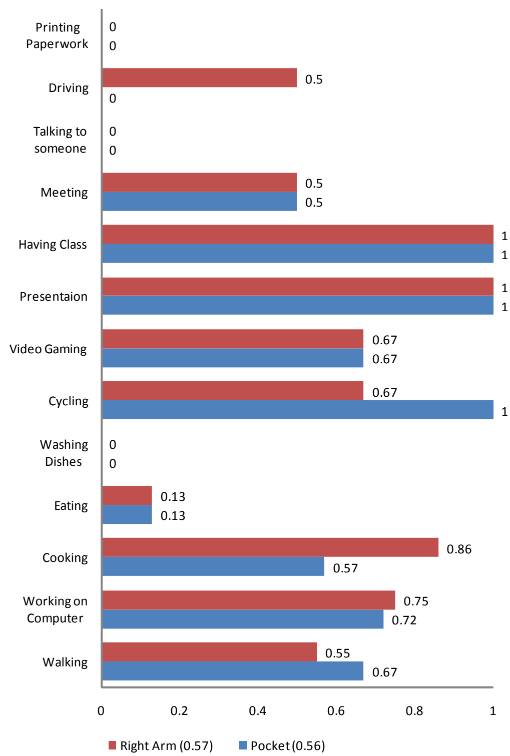


Fig. 8. F1 score of user annotations on pocket and right arm datasets, produced by only using accelerometer data. For pocket dataset, the overall F-Score is 0.57, while the score of right arm dataset is 0.56.

6.4 Hierarchical Activity Recognition vs. Smoothed HMM

We also compare the performance between the hierarchical activity segmentation with the smoothed single-layer HMM. From the best performance of HMM, we set the number of states in HMM to 10 and set the smoothing window size to 800 (around 6 minutes) empirically. Averaged over all activity types, HMM performs worse than the hierarchical segmentation approach (Figure 9). In particular, HMM performs badly on high-level activities such like *"Having Class"*, *"Meeting"*, *"Working on Computer"* and *"Presentation"*. These activities are usually composed by multiple low-level activities and have multiple motion patterns. HMM doesn't have the capability to merge these similar pattern to a higher level activity.

We found that HMM's inability to provide detailed activity structure information also hinders the user to annotate lower level activities. For example, from HMM's segmentation the user can only annotate *"Working Before Computer"* but cannot label sub-activities like *"Writing Paper"* *"Data Preprocessing"* and *"Activity Annotation"* (Figure 10).

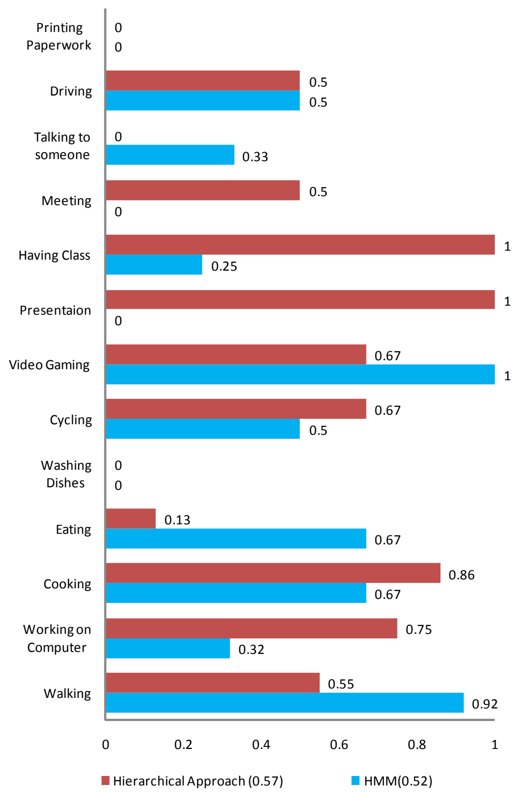


Fig. 9. The F-Score of user annotation on Hierarchical activity segmentation result vs. HMM, on right arm dataset, motion only

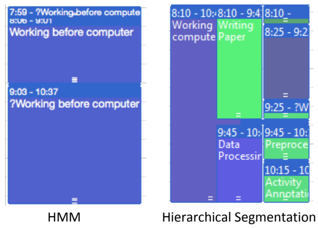


Fig. 10. Single level activity segmentation might lost information regards of lower level events

6.5 Power Consumption

To achieve ubiquitous sensing, selected sensors on the smart phone need to work continuously. With all sensors active, the sensing application can drain the battery very quickly. We estimate how long the mobile sensing platform can continuously monitor the user.

We exam the battery life under three sensing configuration sets. The first one keeps all sensors on; the second one keeps all sensors on except the camera. In the third setting, both camera and microphone are turned off.

Experiments show that the phone can only work for 3 hours without charging if all sensors were turned on. With the camera turned off, it can work for 8-10 hours, which is long enough for daily activity monitoring. The smart phone can work for more than 24 hours if microphone is turned off.

6.6 Sensor Fusion

As shown in Figure 9, unsupervised activity segmentation using accelerometer sensor only fails to identify activities such as *"Washing dishes"*, *"Driving"* and *"Talking to somebody"*. By analyzing the original lifelogs, we noticed that these activities don't differ significantly motion-wise from their preceding and following activities. For instance, *"Driving"* is usually followed by *"Having dinner"* whose motion signatures are similar to *"Driving"* . Using motion-only information, the system can not separate these activities from each other and these activities are recognized as one super-activity such as "driving, then having dinner". This is common for events that last less than 15 minutes.

We experiment with sensor fusion of combining motion sensors with location information. Instead of simply concatenate the two sensor into one, we use the mutual information criterion to fuse two sensors only when needed (Section 4). Figure 11 compares results of using motion only information and "motion plus location" fusion for activity segmentation. In general, sensor fusion significantly improves the recall for location related activities such as "driving" and "eating". In motion only segmentation, these two activities are usually concatenated into one. After introducing the GPS location information, the system is now able to distinguish them. Similarly, with motion-only information, "meeting" and "talking" are usually recognized as *"woking on computer"*. Combined with location information, they can now be identified correctly.

However, we do notice sensor fusion causes lower activity segmentation when location information should not be used to identify an activity. In particular, for activities that occur at different locations such as "working on computer" and "cycling" , there is no need to distinguish "working on computer at room 101" vs. "working on computer at room 102", or "cycling on Castro street" vs "cycling on Liberty Ave.". But for the overall performance, the sensor fusion result is superior than motion only activity segmentation (Figure 11).

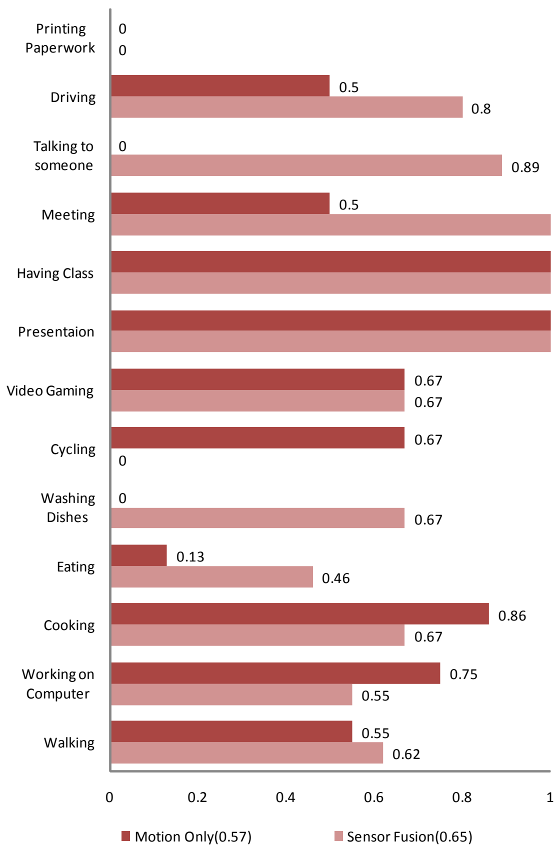


Fig. 11. A comparison of motion only and sensor fusion activity annotation results, on right arm dataset

7 Related Work

There are many works try to recognize user activity from accelerometer data [5,13]. The most successful and exhaustive work in this area is made by

Bao et al. [2]. In their experiments, subjects wore 5 biaxial accelerometers on different body positions as they performed a variety of activities like walking, sitting, standing still, watching TV, running, bicycling, eating, reading etc. Data collected by the accelerometers was used to train a set of classifiers, which included decision trees (C4.5), decision tables and nearest-neighbor algorithm found in the Weka Machine Learning Toolkit [23]. Decision tree classifiers showed the best performance, recognizing activities with an overall accuracy of 84%. However, most of the investigated algorithms are supervised and would be hard to be applied to applications like *SensCare*, since collecting enough trainning data is difficult under such a real world application senario, and it's impractical to train models for all possible activities in human's life.

The sequential and temporal characteristic of activity makes dynamic models such as the Hidden Markov Model (HMM) widely used in activity recognition. To overcome some of the shortages of HMM, like performance degrades when the range of activities become more complex or long-term temporal dependency exists between activities that makes Markov assumption difficult to deal with [6], variations of HMM were developed such as Layered HMM [19], Switching Hidden Semi-Markov Model (HSMM) [6] and Hierarchical Hidden Markov Model (HHMM) [7] for sensor-based activity detection. Although building HMM family models don't need labeled training data, however, it requires to predefine the number of state as an parameter for model training. In activity recognition's perspective, for HMM based systems, the number of activities needs to be a prior knowledge, which might only be true under certain application senarios. On the other hand, the EM algorithm employed by HMM for parameter estimation is a performance bottleneck for model trainning. Our experience with Hierachical HMM has indicated that it is still hard to scale up to efficiently handle the data size as what SensCare needs to process.

There are several works try to model activities using language-based techniques. Such techniques include suffix-tree [10], probabilistic context-free grammar [9,18], and a decision-tree-based [8] method. While all these techniques are hierarchical, they are either supervised or require a lot of human intervention.

8 Conclusion

In this paper, we introduce SensCare, an activity summarization system using smart phones to provide summarization for elder healthcare systems. We evaluate the feasibility of using smart phones as sensing platform regarding of user acceptance, privacy, power consumption and the impact of phone position on activity detection.

An unsupervised hierarchical activity segmentation algorithm was used in the system to overcome challenges like detecting activity with rare labeled data and unknown activity granularities. We compare the performance of selected algorithm with a single-layer Hidden Markov Model. Result show that the structural activity segmentation significantly improves the annotation quality.

A heterogeneous sensor fusion technique is used to improve the activity segmentation. Experiments over five days of real life dataset indicate that activity segmentation based on sensor fusion is much better than using motion infomration only.

For future work, we will investigate a more principled method for sensor fusion to determine which sensor types should be fused to best describe an activity. We will use all sensor information collected in our future experiments and work with our partners in elderly care to test the system with elderly users.

Acknowledgement. This research was supported in part by CyLab at Carnegie Mellon under grants DAAD19-02-1-0389 and W911NF-09-1-0273 from the Army Research Office and the awards from Google Research and Cisco Research.

References

1. Aging in place. Journal of Housing for the Elderly, `http://www.seniorresource.com/ageinpl.htm`
2. Bao, L., Intille, S.S.: Activity recognition from user-annotated acceleration data, pp. 1–17. Springer (2004)
3. Baum, L.E., Petrie, T., Soules, G., Weiss, N.: A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. The Annals of Mathematical Statistics 41(1), 164–171 (1970)
4. Buckwalter, K.C., Davis, L.L.: Elder caregiving in rural communities (2001)
5. Bussmann, J.B.J., Martens, W.L.J., Tulen, J.H.M., Schasfoort, F.C., van den Berg-Emons, H.J.G., Stam, H.J.: Measuring daily behavior using ambulatory accelerometry: The Activity Monitor. In: Behavior Research Methods, Instruments, & Computers, pp. 349–356 (August 2001)
6. Duong, T.V., Bui, H.H., Phung, D.Q., Venkatesh, S.: Activity recognition and abnormality detection with the switching hidden semi-markov model. In: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), vol. 01, pp. 838–845. IEEE Computer Society, Washington, DC (2005)
7. Fine, S., Singer, Y., Tishby, N.: The hierarchical hidden markov model: Analysis and applications. Machine Learning 32(1), 41–62 (1998)
8. Ghasemzadeh, H., Barnes, J., Guenterberg, E., Jafari, R.: A phonological expression for physical movement monitoring in body sensor networks. In: 5th IEEE International Conference on Mobile Ad Hoc and Sensor Systems, pp. 58–68 (September 2008)
9. Guerra-Filho, G., Fermuller, C., Aloimonos, Y.: Discovering a language for human activity. In: Proceedings of the AAAI 2005 Fall Symposium on Anticipatory Cognitive Embodied Systems, Washington, DC (2005)
10. Hamid, R., Maddi, S., Bobick, A., Essa, I.: Structure from statistics-unsupervised activity analysis using suffix trees. In: International Conference on Computer Vision (2007)
11. Harvard Sensor Networks Lab: Codeblue project: Wireless sensor networks for medical care, `http://fiji.eecs.harvard.edu/CodeBlue`
12. Huynh, T., Fritz, M., Schiele, B.: Discovery of activity patterns using topic models. In: Proceedings of the 10th International Conference on Ubiquitous Computing, UbiComp 2008, p. 10 (2008)

13. Lee, S.W., Mase, K.: Activity and location recognition using wearable sensors. IEEE Pervasive Computing 1(3), 24–32 (2002)
14. Logan, B., Healey, J., Philipose, M., Tapia, E.M., Intille, S.S.: A Long-Term Evaluation of Sensing Modalities for Activity Recognition. In: Krumm, J., Abowd, G.D., Seneviratne, A., Strang, T. (eds.) UbiComp 2007. LNCS, vol. 4717, pp. 483–500. Springer, Heidelberg (2007)
15. Maurer, U., Smailagic, A., Siewiorek, D., Deisher, M.: Activity Recognition and Monitoring Using Multiple Sensors on Different Body Positions. In: International Workshop on Wearable and Implantable Body Sensor Networks (BSN 2006), pp. 113–116 (2006)
16. Mundt, C., Montgomery, K., Udoh, U., Barker, V., Thonier, G., Tellier, A., Ricks, R., Darling, R., Cagle, Y., Cabrol, N., Ruoss, S., Swain, J., Hines, J., Kovacs, G.: A multiparameter wearable physiologic monitoring system for space and terrestrial applications. IEEE Transactions on Information Technology in Biomedicine 9(3), 382–391 (2005)
17. National Center for Chronic Disease Prevention and Health Promotion: Improving and Extending Quality of Life Among Older Americans: At A Glance 2010 (2010), `http://www.cdc.gov/chronicdisease/resources/publications/AAG/aging.htm`
18. Ogale, A., Karapurkar, A., Aloimonos, Y.: View-invariant modeling and recognition of human actions using grammars. In: Workshop on Dynamical Vision at ICCV, pp. 115–126 (2005)
19. Oliver, N., Horvitz, E., Garg, A.: Layered representations for human activity recognition. In: Proceedings of the 4th IEEE International Conference on Multimodal Interfaces, ICMI 2002, pp. 3–8. IEEE Computer Society, Washington, DC (2002)
20. Papineni, K., Roukos, S., Ward, T., Zhu, W.: Bleu: a method for automatic evaluation of machine translation. Technical Report RC22176(W0109-022), IBM Research Division, Thomas J. Watson Research Center (2001)
21. Virone, G., Wood, A., Selavo, L., Cao, Q., Fang, L., Doan, T., He, Z., Stoleru, R., Lin, S., Stankovic, J.A.: A an advanced wireless sensor network for health monitoring (2006)
22. Winograd, T.: Architectures for context. Human-Computer Interaction 16, 401–419 (2001)
23. Witten, I., Frank, E.: Data mining: Practical machine learning tools and techniques with java implementations. The Morgan Kaufmann Series in Data Management Systems (October 1999)