

Real World Activity Recognition with Multiple Goals

Derek Hao Hu, Sinno Jialin Pan, Vincent Wenchen Zheng, Nathan Nan Liu, Qiang Yang

Department of Computer Science and Engineering
Hong Kong University of Science and Technology
{derekhh, sinnopan, vincentz, nliu, qyang}@cse.ust.hk

ABSTRACT

Recognizing and understanding the activities of people from sensor readings is an important task in ubiquitous computing. Activity recognition is also a particularly difficult task because of the inherent uncertainty and complexity of the data collected by the sensors. Many researchers have tackled this problem in an overly simplistic setting by assuming that users often carry out single activities one at a time or multiple activities consecutively, one after another. However, so far there has been no formal exploration on the degree in which humans perform concurrent or interleaving activities, and no thorough study on how to detect *multiple* goals in a real world scenario. In this article, we ask the fundamental questions of whether users often carry out multiple concurrent and interleaving activities or single activities in their daily life, and if so, whether such complex behavior can be detected accurately using sensors. We define several classes of complexity levels under a goal taxonomy that describe different granularities of activities, and relate the recognition accuracy with different complexity levels or granularities. We present a theoretical framework for recognizing multiple concurrent and interleaving activities, and evaluate the framework in several real-world ubiquitous computing environments.

ACM Classification Keywords

I.2.9 Robotics: Sensors

Author Keywords

activity recognition, concurrent activities, interleaving activities, conditional random field

INTRODUCTION

Activity recognition aims to recognize the actions and goals of one or more agents from a series of observations [21, 12, 9]. In ubiquitous computing, researchers have strived at discovering users' activities and goals through sensor readings. Successful and accurate activity recognition systems will help provide personalized support

for many real world applications including health monitoring, education, entertainment, etc. A typical scenario of activity recognition is in assisting the sick and disabled; e.g. [23] shows that by providing adaptive, personalized reminders of daily activities, older adults can adapt to cognitive decline better and remain in their own homes longer. Furthermore, one can find applications ranging from security-related applications and logistics support to location-based services [22, 30]. Due to its many-faceted nature, different fields may refer to activity recognition as plan recognition, goal recognition, intent recognition, behavior recognition, location estimation and location based services. In the rest of the paper, we do not explicitly differentiate between "goal recognition" and "activity recognition" since they are essentially trying to solve the same problem.

In real world situations, we intuitively find that users often pursue several activities in a concurrent and/or interleaving manner. However, how often does it occur that users pursue several goals at once in a concurrent and/or interleaving manner? As often referred to as one-track-mind, are we, as humans, often pursue one goal to the end before starting another? Furthermore, if humans often pursue several goals in a concurrent and interleaving manner, are we able to detect such complex social and behavioral patterns from sensors alone? Do these goals have any deep association with the algorithmic complexity of the solutions that are aimed at recognizing the goals? These are the scientific questions that we wish to answer in this paper.

A deep understanding of multiple-goal pursuit behavior is essential when we wish to model users' activities from observed action sequences. In daily life, a user may accomplish multiple goals within a single sequence of actions where users may achieve one goal, switch to another goal, then return to a previous goal, or do so for several sets of goals in a concurrent manner. We model multiple goals mainly by two features, namely concurrency and interleaving, where concurrency means several goals are pursued in the same time slice and interleaving means goals are pursued non-consecutively, in that one goal is paused and then resumed after a while during which time another goal is being pursued.

For example, it is common for an individual to watch TV and answer the telephone at the same time concurrently. As an example of interleaving goals, an in-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or to publish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

UbiComp '08, September 21-24, 2008, Seoul, Korea.

Copyright 2008 ACM 978-1-60558-136-1/08/09...\$5.00.

	Single actor	Two actors
Goals are compatible	Goal synergy	Goal concord
Goals are not compatible	Goal conflict	Goal competition

Table 1. Wilensky’s four-way classification of multiple goal states

dividual may be having his breakfast when the water boils, so he has to pause the activity of having breakfast and attend to the water boiling activity. Then he may return to the previous goal of having his breakfast. If these features, i.e., concurrent and interleaving goals, are taken into consideration, we can potentially improve the recognition accuracy of many real world activity recognition applications. We call this computational problem the *multiple-goal recognition* problem.

The problem of understanding multiple goals has previously been studied by researchers in many disciplines other than computer science. In psychology, it was pointed out in [17] that “unlike the sequences directed to a single main goal in a simple or technical plan, human intended action is influenced by multiple goals”. Furthermore, as early as 1983, it was argued by Wilensky[29] that the main characteristic of human planning, as compared with technical planning, is the ability to reason about the problems arising from multiple goals. He also proposed a four-way classification of multiple goal states (see Table 1). Therefore, understanding multiple goals is important not only for computer scientists, artificial intelligence or ubiquitous computing community, but also for natural sciences as a whole.

In educational psychology, different types of goals have been identified to be pursued by students when accomplishing academic tasks, such as performance goal, task goal and work avoidance goal[1] and it was reported that more than one goal may be simultaneously operative[16]. It was suggested that cognitive and self-regulatory processes depend partly on the joint and interactive effects of multiple goals more precisely than on single goals, so that it is likely that their combined impact may differ from their individual effects and several goals can interact in complementary ways to jointly regulate achievement behaviors [24, 25]. It is also very important to other researchers including cognitive science, anthropology and behavioral psychology.

In this paper, we attempt to answer the above questions from two angles. We first investigate the available benchmark data from a real world ubiquitous computing environment, whereby we show that pursuing multiple goals is commonplace in human activities. We then propose an algorithm to recognize these goals accurately entirely from sensor readings. We show that different recognition accuracies can be obtained at different levels of granularity in a taxonomic goal hierarchy.

Our algorithmic solution is to solve the multiple-goal recognition problem using a variant of the Conditional Random Field (CRF) model, which is called SCCRF

(Skip Chain Conditional Random Field) [27], where concurrent activities can be modeled as the hidden states in the CRF. However, it is not trivial to combine interleaving and concurrent activities in a unified CRF framework with acceptable speed and accuracy. Our proposed approach is to combine a skip-chain CRF for modeling interleaving activities with a correlation goal graph for modeling concurrent activities. We will evaluate our algorithm in several ubiquitous computing datasets to demonstrate the effectiveness of our algorithm. We will further show that our algorithm can achieve different levels of recognition accuracy when the activities we are supposed to be recognizing are more fine-grained or coarse-grained, i.e. fall into different complexity levels or different granularities.

RELATED WORK

Activity recognition has attracted the interest of pervasive computing community for a long time. Patterson developed methods to infer high-level behaviors from low-level sensors by building a hierarchical Dynamic Bayesian Network (DBN) and analyze the model by taking the relationships between inter time slice and intra time slice into consideration[21]. This approach was used in an application to provide cognitive assistance with transportation services[22]. MIT’s PlaceLab has also done several works in acquiring the sensor reading data in a live-in laboratory for future research, and has made such data publicly available[15, 10]. Other research areas can also be related to the activity recognition area in pervasive computing community, such as human identification from the object usage patterns, recorded by RFID sensors[9], etc.

Outside of pervasive computing, activity recognition finds its way in many other research areas of computer science as well. One can find related research in the computer vision area, which concerns tracking moving people and objects from video captures[4]. In sensor-based activity recognition research [7, 30, 14], work has been done that uses GPS, RFID and other electromagnetic sensors, as well as Wi-Fi routers and clients. In natural language understanding, human discourses are a major form of input from which one can recognize the intent of agents [2].

In artificial intelligence, there are two major approaches in solving the problem of activity recognition: logic or consistency based approaches, and probabilistic approaches. Early approaches of activity recognition [11] were based on logic and were described as a logical inference process of circumscription, where a recognizer’s knowledge is represented by a set of first-order statements called event hierarchy encoded in first-order logic.

However, logic-based approaches have limitations in distinguishing among consistent plans and have problems to handle uncertainty and noise in sensor data.

Despite having its deep roots in knowledge representation, over the years it is realized that a large portion of this field concerns statistical modeling. In sensor-based activity recognition, which is especially challenging due to the inherent noise of the input, machine learning in activity recognition has been done in a layered form, where recognition at several intermediate levels is conducted in a serialized manner. From the lowest level where the sensor data is collected, a learning-based approach concerns how to find detailed locations of agents. At the intermediate level, learning may be concerned with how to recognize individual activities from the inferred locations at the lowest level. For example, research has been done on recognizing the transportation modes of agents[13], basic action sequences from simple sensors [20] as well as from the locations of agents in an indoor environment[18, 19]. Furthermore, at the highest level of consideration, a major issue is to find out the overall goal or subgoals of an agent from activity sequences. Nevertheless, as pointed out by [12], although these different levels emphasize different aspects of human activities, their essential goals are the same.

In learning-based approaches, state-space based models are especially attractive with the underlying assumption that there exist hidden states that represent the activities and goals, and that the hidden states are evolving. State-space based models enable the inference of hidden states given the observations up to the current time. They are suitable for modeling high-level hidden concepts given the low-level observations; examples include Bui’s AHMEM [6], aggregate dynamic Bayesian networks [20], conditional random fields with its many variants [28].

PROBLEM STATEMENT

In this section, we first describe an example domain where we highlight the necessity of considering multiple goals. We then give a classification of different types of multiple goals. Finally we give an overview of our algorithm for multiple goal recognition.

Multiple Goals in the Real World

Consider a real-world ubiquitous computing environment, what will be the proportion of multiple goals, when we consider the features of concurrency and interleaving? What will be the proportion of multiple goals at different levels of goal hierarchy? What will be the relationship between recognition accuracy and goal hierarchy? All these questions should be answered with respect to a taxonomy of goals.

We use the MIT PlaceLab House_n PLIA1(“PlaceLab Intensive Activity Test Dataset 1”) for illustrating the taxonomic nature of multiple goals. This dataset was

recorded on Friday March 4, 2005 from 9 am to 12 noon with a volunteer familiar with Placelab. The actions performed were correctly recorded and labeled in this dataset. Thus, we first manually construct the goal hierarchy from MIT PLIA1 dataset. The lowest level, where the activities are extracted from the original data, includes activities such as “sweeping”, “washing-ingredients”, etc. We combined some of the relevant activities to a more generalized activity and constructed the medium level, which include some higher level activities like “preparing ingredients”, “Dealing-with-clothes”, etc. Next, we divided all the activities into 9 categories, which compose our highest level to include: cleaning indoor, yard-work, laundry, dishwashing, meal-preparation, hygiene, grooming, personal and information/leisure.

A partial description of the goal taxonomy is shown in Figure 2. The complete description of the goal taxonomy is omitted here due to space constraint.

Do we have concurrent and interleaving goals under the definition of such a goal hierarchy? The answer is yes. In a goal recognition algorithm, usually a “window size” is empirically determined so as to define the minimum time slice we consider. We illustrate a segment of activity sequence in the PLIA1 dataset under different levels of goal hierarchy to show the existence of concurrent and interleaving goals. In this illustration, the “window size” is set as one minute.

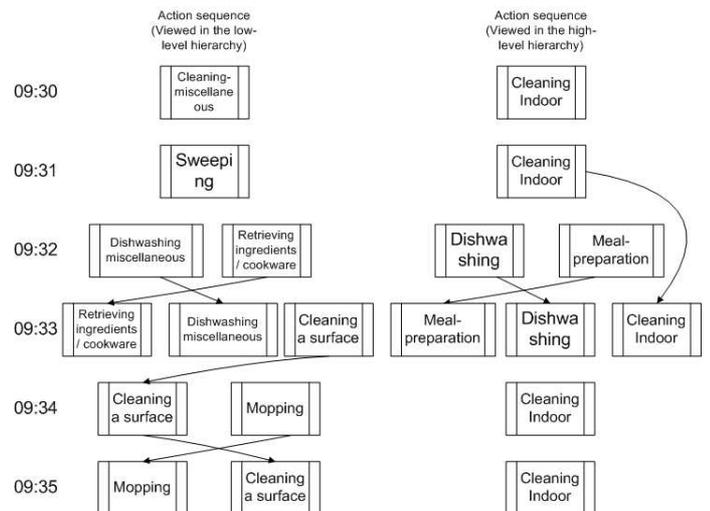


Figure 1. Illustration of a segment of action sequence, where the left sequence from top down shows a low-level multiple goal layout by time, and the sequence on the right shows that of the high level goals

In Figure 1, we extracted an activity sequence from 09:30am to 09:35am and set the “window size” as one minute. From the figure, we can see the existence of concurrent and interleaving goals, and that concurrent goals are more frequent than interleaving goals. For example, at 09:33am, three goals happen at the same time, in the low level, they are represented as “Retrieving

ingredients / cookware”, “Dishwashing-miscellaneous” and “Cleaning a surface”, while in the high level, they are represented as “Meal-preparation”, “Dishwashing” and “Cleaning Indoor”. However, both two representations show a case of the existence of concurrent goals.

For interleaving goals, in the high level representation, the goal pursued at 09:31 am was “Cleaning Indoor”, while the goal pursued at 09:33am was also “Cleaning Indoor”, which shows an example of interleaving activities. Intuitively, we may think that interleaving goals will increase when we move up in the goal hierarchy, which seems to be the case.

Another phenomenon worth mentioning is that, when we move up the goal hierarchy, i.e. when the activities become more coarse-grained, some activities may merge together to form one activity. For example, consider the action sequence in 09:34am and 09:35am, at the lowest level, the activities happened were “Cleaning a surface” and “Mopping”, while these two activities both belong to the “Cleaning Indoor” activity in the high level.

As we can see from this taxonomy and Figure 1, when we go deeper in the hierarchy, more goals will appear. Similarly, as we go deeper, the granularity of goals is also getting more refined. We can see that at the highest level of the taxonomy, there are only around a dozen goals. Thus, we expect that when we perform activity recognition at the higher levels of this taxonomy, less concurrent and more interleaving goals will be observed. We will confirm this fact through experiments later in the paper.

Furthermore, as we can observe, at the second and third levels, the goals are becoming increasingly detailed, making it more difficult to tell one goal from another based on a same set of sensors. Thus, we also expect the number of sensors to increase in order to provide various kinds of sensor reading data, as we go down the hierarchies further.

As we discussed above, in different levels of goal hierarchy, there will be different degrees or proportions of multiple goals, and the complexity levels of goals will be different as well. Furthermore, some activities may be merged to form single activities as we move to the high level. Since the observed action sequences show different behavior in terms of the number of goals when we view them at different levels of a goal hierarchy, it is therefore necessary to consider the goal recognition questions in a hierarchical manner.

Types of Multiple Goals

Many previous works on activity recognition tackled the goal-recognition problem in a single activity setting. To help understand the multiple-goal activities better, in this section, we classify the common patterns of goal or activity composition in action sequences into five types. (See Figure 3) However, the previous approaches only



Figure 2. Goal Hierarchy: Subfigure 1

tried to tackle Type 1 and Type 2 (Single Goal / Multiple and Independent Goals). As we described in last section, in real-world scenarios, it is very important to solve goal composition types 3, 4 and 5, where we can model interleaving and concurrent goals.

Note that although it is impossible for us to tell from the sensors whether two goals are in concord or competition relationship as indicated by Wilensky [17], as we do not know from the sensor readings the details of each action, we do know whether two goals are being pursued at the same time or not. Thus, as a result, we can classify goals into different types of composition as indicated in Figure 3.

In our algorithm, we will exploit skip-chain conditional random field (SCCRF) for modeling interleaving goals. It is especially attractive for our model since we can add skip edges between non-consecutive actions which are believed to be pursuing the same goal. Furthermore, CRF-based methods make no assumption on the dependence structure between different observations, thus it can model more complex relationships and collectively label all the data.

PROPOSED APPROACH

Modeling Interleaving Goals via CRF

In this and next section, we describe our approach in recognizing multiple goals from sensor readings.

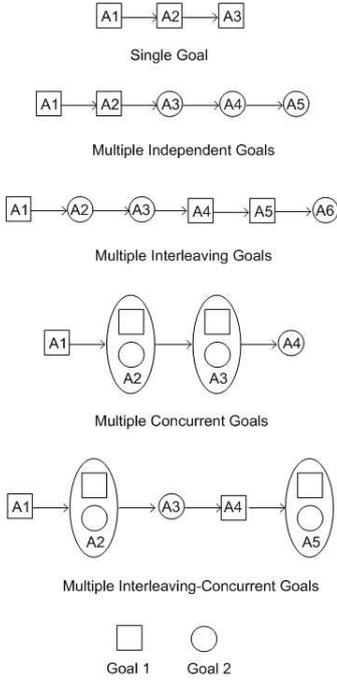


Figure 3. Ways of goal composition in activity sequences

In this section, we first focus on modeling interleaving goals. An example of interleaving goals is illustrated by the following example. As an example, a person may go to the general office to get the projector for the “seminar” goal, then goes to the printing room to pick up the printing material out for the “printing” goal. He may then go towards the seminar room. Through this example, we note that the “get projector” and “go towards seminar room” activities may have *long-distance dependencies* because the seminar goal is paused when the professor goes to the printing room.

We choose skip-chain CRF as proposed in [26] to model the interleaving goal issue for the following reasons. Skip-chain CRF has been extensively used in natural language processing (NLP), where the problem of Named Entity Recognition (NER) has similarities with the multiple-goal recognition problem to model the correlation between non-consecutive identical words in a text document.

Skip-chain CRF also has an advantage in modeling uncertainty in the real world, and allows posterior probabilities to be learned from the training data to add the skip edges.

In our model, each skip-chain CRF infers whether an individual goal is active or not, given a newly observed activity (See Figure 4). For long-distance dependencies, for each goal G_k , we first infer the action-transition probability $P(A_i|A_j, G_k)$ (G_k is shown as Goal 1 in Figure 4), which stands for the probability of the following situation: given that the goal being pursued is G_k , the

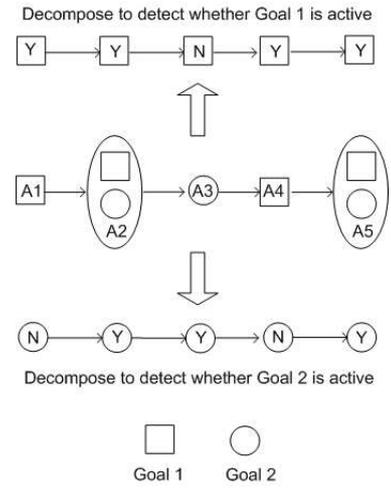


Figure 4. Activity sequence decomposed into goal sequences

last action in the process of pursuing goal G_k is A_j and the next activity being A_i . Maximum Likelihood Estimation (MLE) can be used to learn this probability. We simplify the model by assuming that the prior distribution is uniform.

In the skip-chain CRF (SCCRF) model, a second type of potential is added to represent long-distance edges. For each of the m goals under consideration, we build a corresponding SCCRf model, with the i^{th} SCCRf being used to infer whether the goal G_i is active given the set of observed activity sequences as training data. For a factor graph $G = (V, E)$, a linear-chain CRF can be set up with additional *long-distance edges* between activities A_i and A_j such that $P(A_i|A_j, G_k) > \theta$ (Refer to Figure 5 for an illustration), where θ is a parameter that can be tuned to adjust the confidence of such long-distance dependencies.

For an observation sequence \mathbf{x} , let $\mathcal{I} = \{u, v\}$ be the set of all pairs of activities for which there are skip edges connected with each other. Then the probability of a label sequence \mathbf{y} given an observation activity sequence \mathbf{x} is:

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{t=1}^n \Psi_t(y_t, y_{t-1}, \mathbf{x}) \prod_{(u,v) \in \mathcal{I}} \Psi_{uv}(y_u, y_v, \mathbf{x}). \quad (1)$$

In the above Equation 1, Ψ_t are the factors defined for linear-chain edges and Ψ_{uv} are the factors defined over the skip edges. (Also refer to Figure 5 for illustration) $Z(\mathbf{x})$ is the normalization factor. We define the potential functions Ψ_t and Ψ_{uv} in Equation 2 and Equation 3 as:

$$\Psi_t(y_t, y_{t-1}, \mathbf{x}) = \exp \left(\sum_k \lambda_{1k} f_{1k}(y_t, y_{t-1}, \mathbf{x}, t) \right) \quad (2)$$

$$\Psi_{uv}(y_u, y_v, \mathbf{x}) = \exp\left(\sum_k \lambda_{2k} f_{2k}(y_u, y_v, \mathbf{x}, u, v)\right) \quad (3)$$

λ_{1k} are the parameters of the linear-chain template and λ_{2k} are the parameters of the skip-chain template. Each of them factorize according to a set of features f_{1k} or f_{2k} .

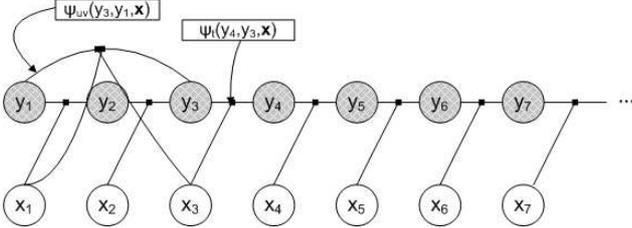


Figure 5. Illustration of the SCCRf model

Exact inference in CRFs is computationally expensive, and therefore we set a maximum number of iterations. We learn the weights λ_{1k} and λ_{2k} for the skip-chain CRF model by maximizing the log-likelihood of the training data, which requires calculating the partial derivative and optimization techniques. We omit the detailed algorithm of inference and parameter estimation of the skip-chain CRF model, but refer the readers to [27].

Modeling Concurrent Goals via Goal Graphs

We apply correlation analysis to tell when related goals may be pursued together. For example, it may be more likely that a “eating-dinner” goal is carried out when a “sitting-at-table” goal is simultaneously carried out. We wish to use the training data to build a correlation graph of goals, where two goals are related by an edge with a large positive weight in $[0, 1]$ if they have strong positive correlations. We omit the considerations of negative correlations here, which we leave for future work. To simplify our model, we only consider the probability $P(G_i|G_j)$ using our goal graph explicitly.

From the training data, we can learn the posterior probability $P(G_i|G_j)$ and use it as the initial similarity matrix. We take the posterior probability of each pair of goals and define an $m \times m$ initial similarity matrix S as $S[i, j] = P(G_i|G_j)$.

Since the training data may be sparse, the posterior probability we get from the training data may not be so reliable. [3] proposed a method for computing the similarity matrix between vertices of different graphs. We adapted their method for modeling the similarity between vertices of the same graph. We build a directed graph $G = \langle V, E \rangle$, where the vertices V indicate different goals and $e = \langle g_a, g_b \rangle$ indicates that a goal g_a and a goal g_b have some kind of connection, so that when g_a appears, g_b is also likely to appear. The similarity matrix can be updated through iterations of $S_{k+1} = AS_k A^T + A^T S_k A$, where A denotes the adjacency matrix of the similarity graph. After creating the similarity

matrix S , we can model concurrent goals by minimizing the differences between strong correlated goals to ensure that they will appear together. As a result, our top-level inference consists of minimizing the following objective function with similarity matrix S and initially inferred probabilities $\mathbf{P}' = \{P'_1, P'_2, \dots, P'_m\}$ as well as our desired output $\mathbf{P} = \{P_1, P_2, \dots, P_m\}$.

$$\min \sum_{i,j \in \{1, \dots, m\}} (P_i - P_j)^2 S_{ij} + \mu (P_i - P'_i)^2 \quad (4)$$

In a more detailed analysis, we show that the optimization problem mentioned above can be formulated as a quadratic programming (QP) problem and solved using standard techniques in QP (a more detailed description can be found in [5]).

EXPERIMENTAL RESULTS

In this section, we try to answer the questions we pose out at the very beginning, i.e. what is the proportion of multiple goal pursuance in a real-world scenario, given a specific coarse level in the goal hierarchy? Is it necessary to model the phenomenon of concurrent and interleaving goals in the goal recognition algorithm? Can we recognize the different goals accurately using our proposed method? What is the relationship between different recognition accuracy and levels of goal hierarchy? We plan to answer these questions through our experimental results.

Do Humans Pursue Multiple Goals?

We analyze the proportion of concurrent goals by looking deeper into the MIT PLIA1 Dataset, upon which we had built the goal hierarchy. We counted the proportion of multiple goals over the number of all goals in the three different levels of goal hierarchy. Another parameter we are changing is the “window size”, which is the length of a time slice upon which we define “concurrency”. It is natural to imagine that, when the “window size” approaches zero or is very small, no concurrency will actually occur since we can always decompose human activities into segments of interleaving activities.

We show the statistics of the proportion of multiple goals in the following Figure 6. The x-axis denotes different values of “window size” and the y-axis denotes the proportion of multiple goals in the action sequence. This proportion is defined as the ratio of time slices during which there exist concurrent goals and the total number of time slices, over all activity sequences, given a specific window size.

From the above figure, we can see that the proportion of multiple goals in a real-world action sequence is relatively large; i.e. it is very important to model multiple goals in activity recognition. In addition, it can be observed from Figure 6 that the proportion of concurrent goals increases as the granularity of activities becomes less fine-grained. Since as we go up alongside with the

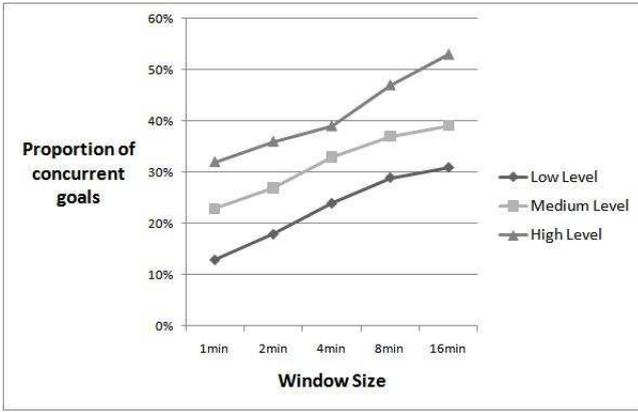


Figure 6. Statistics of multiple goal proportion

goal hierarchy, some fine-grained goals that are not concurrent at the low level will be classified into the same category and hence becomes concurrent. This concurs with our intuition.

Another interesting question is whether there will be more concurrent goals after a user has started an activity sequence for a while. Likewise, we wonder if there will be fewer concurrent goals towards the end of an activity sequence. Towards answering these questions, we gather additional statistics from the MIT PLIA1 dataset, where the results are shown in Figure 7. In this figure, we have divided the whole activity sequence in all test data into five parts, which are denoted as “1st, 2nd, 3rd, 4th, 5th”, respectively, in the figure. We then counted the average number of concurrent goals in each time phase. For this dataset, we can see that concurrent goals occur maximally in the middle of the activity sequence. This indicates that it is necessary for us to allocate more sensors for detecting multiple goals towards the middle of a long activity sequence. It would be interesting to test whether this assumption holds for other ubiquitous computing datasets in a more statistically significant manner.

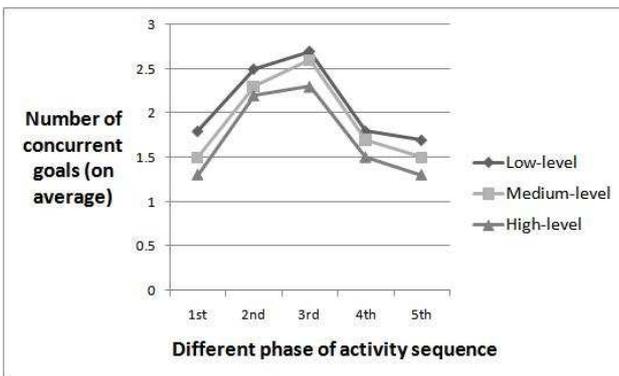


Figure 7. Average number of concurrent goals in different phases of activity sequence

How Accurately Can We Predict Multiple Goals?

In this section, we will present the experimental results on the performance of our algorithm. We will test our algorithm through two different datasets, where the first dataset is collected from an indoor Wi-Fi environment in a university department building, and the second dataset comes from MIT Placelab.

To set up the experiments, we need to compare our algorithm with a commonly used baseline. As we have reviewed in Section 2, many previous goal recognition approaches follow a single-goal recognition strategy. Thus, we have chosen to use Naive Bayes algorithm as our baseline algorithm, which only focuses on a single goal setting. We would like to observe whether this baseline will reduce the recognition accuracy significantly when we consider single goals only.

Naive Bayesian classifiers are shown to perform very well in many machine learning applications [8]. For classification, this algorithm computes the posterior probability $P(G_j|x)$ of sample x belonging to class G_j according to the Bayes’ rule:

$$P(G_j|x) = \frac{P(x|G_j)P(G_j)}{P(x)},$$

where

$$P(x|G_j) = \prod_{m=1}^{|S|} P(S_m = v_{m,k}|G_j)$$

is the product of $|S|$ likelihoods. $|S|$ is the number of sensors readings, where each collection of sensor instance data is represented as a vector x , and $v_{m,k}$ is a possible value of the sensor S_m . Then, a sample x is predicted to a goal j^* that has the highest value $P(G_{j^*}|x)$. If the sensor readings are continuous valued, as in the case of the Wi-Fi data, we will use a probability density function for each of the factors in the product.

For both domains, we will use cross validation to get the recognition accuracy and the recognition accuracy is defined as the percentage of correctly recognized goals over the number of all goals. Also, we test different parameters of our approach to show that the accuracy is reliable under different parameter settings.

The first domain is from [7] where the observations are obtained directly from sensor data and the activities correspond to that of a professor walking in a university office area. In this data set, nine goals of a professor’s activities are recorded, 850 single-goal traces, 750 two-goal traces and 300 three-goal traces are collected so that the dataset can evaluate both multiple-goal recognition and single-goal recognition. We used three-fold cross validation for training and testing.

Experimental results are shown in Table 2. In this dataset, there is only one flat hierarchical level. Thus we only show the comparison of the algorithms at one level. The comparison result is as shown in the second column of Table 2, where the bottom figure shows the

accuracy of Naive Bayes algorithm in the 3-fold cross validation. The remaining figures show the results from our algorithm under different parameter settings. As we can see, on average, our algorithm performs 13% to 30% better than the single goal approach.

Algorithm	Accuracy (Variance)
$\theta = 0.5, \mu = 0.4$	88.2%(1.5)
$\theta = 0.5, \mu = 0.5$	87.3%(1.7)
$\theta = 0.5, \mu = 0.6$	88.4%(1.6)
$\theta = 0.6, \mu = 0.4$	91.3%(1.8)
$\theta = 0.6, \mu = 0.5$	92.0%(2.0)
$\theta = 0.6, \mu = 0.6$	91.8%(2.3)
$\theta = 0.7, \mu = 0.4$	94.0%(2.7)
$\theta = 0.7, \mu = 0.5$	94.8%(2.4)
$\theta = 0.7, \mu = 0.6$	94.2%(2.7)
$\theta = 0.8, \mu = 0.4$	93.7%(2.9)
$\theta = 0.8, \mu = 0.5$	93.2%(2.5)
$\theta = 0.8, \mu = 0.6$	92.9%(2.6)
Naive Bayes	64.8%(1.5)

Table 2. Recognition accuracy: WiFi dataset

For the Wi-Fi dataset, we have only a flat hierarchy and thus cannot answer the question of the relationship between the recognition accuracy and the granularity of recognizing goals. This is not the case with the second dataset, which is the MIT PlaceLab PLIA1 dataset. In this dataset, we are using location information to predict what activity the user is currently pursuing. We choose different levels in the goal hierarchy and predict the corresponding goals in the low level, medium level and the high level, where the result of each level is shown in the second, third and fourth columns of Table 3, respectively.

In Table 3, the lowest level means the level that describes activities such as “Hand-washing-dished”, “Washing-ingredients” while the medium level means the level that describes activities like “Preparing-ingredients”, “Preparing-food” etc. The highest level describes activities like “Meal-preparation” and “Hygiene”. Detailed descriptions of levels of different goal hierarchy can be found in Figure 2. Note that we had only included one subfigure, the other two subfigures are omitted due to space constraints.

From Table 3, we could see that we can indeed achieve recognition accuracy of acceptable performance in a real-world multiple activity recognition scenario. Also, by setting different levels of granularity or choosing different levels in the goal hierarchy, one can achieve different levels of recognition accuracy. Our experimental results confirm our earlier intuition that coarse-grained activity recognition tends to achieve higher accuracy than fine-grained ones. We will discuss this phenomenon in the next section. Some possible and intuitive explanations are that, when viewing these sensor readings as data instances in a high-dimensional space, we are trying to find the decision boundaries between different goals. The decision boundaries between low-level goals

Algorithm	Low-level	Medium-level	High-level
$\theta = 0.5, \mu = 0.4$	80%(3.7)	82%(4.1)	89%(3.9)
$\theta = 0.5, \mu = 0.5$	79%(3.8)	83%(3.9)	91%(4.5)
$\theta = 0.5, \mu = 0.6$	80%(3.2)	82%(4.0)	91%(3.7)
$\theta = 0.6, \mu = 0.4$	80%(3.6)	84%(4.1)	92%(2.8)
$\theta = 0.6, \mu = 0.5$	83%(3.9)	85%(4.0)	94%(3.5)
$\theta = 0.6, \mu = 0.6$	87%(3.8)	84%(3.6)	93%(4.2)
$\theta = 0.7, \mu = 0.4$	84%(3.6)	85%(2.6)	96%(2.7)
$\theta = 0.7, \mu = 0.5$	86%(3.1)	85%(3.3)	94%(2.8)
$\theta = 0.7, \mu = 0.6$	85%(2.9)	85%(3.0)	95%(2.9)
$\theta = 0.8, \mu = 0.4$	83%(3.3)	88%(3.1)	93%(3.0)
$\theta = 0.8, \mu = 0.5$	82%(3.6)	87%(3.2)	93%(3.0)
$\theta = 0.8, \mu = 0.6$	83%(3.4)	88%(2.8)	93%(3.3)
Naive Bayes	41.0%(1.2)	47.2%(1.5)	51.2%(1.3)

Table 3. Recognition accuracy in different granularity levels of MIT PlaceLab dataset in different levels of the goal hierarchy

may be difficult to find due to the fact that these data themselves have much noise and it is difficult to draw the decision boundaries. However, when we move up towards the higher levels, the confusion between different goals appear to be smaller and the decision boundaries are therefore much clearer than the case at the lower level.

Another question we wish to investigate is the sensitivity of different sensors. In the above experiment on MIT PLIA1 Dataset, we were using the location sensor for activity recognition, and it can be seen that the recognition accuracy will be different in different levels. PLIA1 dataset also provides other sensors apart from the “Wireless static” sensor, which measures the movement of an object to which it is attached while sending data to receivers through wireless channels. It also includes sensors such as the wired humidity sensor, wired pressure sensor, wired light sensor, etc.

Was the difference in the observed recognition accuracy caused by the intrinsic nature of goal recognition problem or by the sensitivity of various kinds of sensors in this test environment? We present some preliminary results that might answer this question. Instead of using the location data to estimate user activities, we further used humidity, pressure and light sensor readings for the same task and also tested them in different levels of goal taxonomies. The results are shown in Table 4. Note that, due to time constraint, we need to simplify this experiment to the case where the parameter setting is chosen as $\theta = 0.7, \mu = 0.5$.

Sensor data	Low-level	Medium-level	High-level
Wired humidity	71%(1.8)	75%(2.4)	78%(2.2)
Wired pressure	68%(2.3)	71%(1.9)	70%(2.5)
Wired light	69%(1.9)	73%(3.0)	75%(2.8)

Table 4. Recognition accuracy using different kinds of sensors

As can be seen from Table 4, the same phenomenon

in changing accuracy can be observed even when we use other kinds of sensors. This lends support to our claim that the recognition accuracy is more related to the granularity of goals, which corresponds to the level in the taxonomy of goals to which we designate for the activity recognition problem.

CONCLUSIONS AND FUTURE WORK

In this paper, we studied the problem of goal recognition in real world ubiquitous computing environments. We posed two fundamental questions at the beginning of the paper: (1) how often do users pursue concurrent and interleaving activities? and (2) is it possible to accurately predict this behavior using sensors in a real world activity recognition framework? By analyzing publicly available datasets based on goal hierarchies, we see that more than 30% of the observation sequence have multiple concurrent and/or interleaving activities. Hence, we have demonstrated the importance to analyze multiple goal settings in the problem of activity recognition.

In addition, after demonstrating the importance of multiple activity recognition, we further developed an algorithm that can accurately recognize concurrent and interleaving activities. This algorithm is based on a CRF model to reflect certain assumptions in interleaving activities and exploited the positive correlations between goals for modeling the concurrent activities.

However, another major problem in activity recognition is to determine the granularity of activities we are about to recognize. Given the close-world assumption that is usually assumed in plan recognition or activity recognition, when we try to recognize goals based on different levels of goal hierarchy, different recognition accuracy will be achieved. We empirically tested the recognition accuracy, using our multiple activity recognition algorithm, to evaluate whether the algorithm can achieve different levels of recognition accuracy. The experimental results supported our claim.

We point out several future research directions in the multiple activity recognition framework.

First, can we do better when there are more than one user in the action sequence? This question was first mentioned in [15], where a new MIT House_n dataset: PLCouple was made available that included two users: a man and a woman. Considering Table 1, where for multiple users, they may either cooperate to achieve a goal or compete for a goal. Understanding this scenario is very interesting in real life.

Secondly, can we construct a goal hierarchy automatically from an observation sequence? In a real world setting, the number of activities individuals perform might be very large that it is impossible to construct a goal hierarchy by hand. If we could automatically construct, or derive some logical relationships between high level

goals and low level goals, the resulting output, although it may not be a complete hierarchy, will provide useful information in constructing the final goal hierarchy.

Thirdly, in our experiments, we've shown that different recognition accuracies can be achieved under different levels of granularities. Apart from empirically set the granularities of goal recognition, can we automatically choose the "best" level of granularity, by a few labeled sensor readings? This problem, if solved, can help aid the application where a certain probability of recognition accuracy needs to be achieved. Thus it can tell the user to what levels of granularity it can support, given the hard constraint that the recognition accuracy must achieve a certain probability.

ACKNOWLEDGMENTS

We thank the support from NEC (China) Co. Ltd. We also want to thank the reviewers for providing detailed comments and advice for making it better in the final version.

REFERENCES

1. Elliot A.J. and Harackiewicz. Approach and avoidance achievement goals and intrinsic motivation: A mediational analysis. *Journal of Personality and Social Psychology*, 70(3):461–475, 1996.
2. Nate Blaylock and James F. Allen. Corpus-based, statistical goal recognition. In *IJCAI*, pages 1303–1308, 2003.
3. Vincent D. Blondel, Anahí Gajardo, Maureen Heymans, Pierre Senellart, and Paul Van Dooren. A measure of similarity between graph vertices: Applications to synonym extraction and web searching. *SIAM Rev.*, 46(4):647–666, 2004.
4. Oren Boiman and Michael Irani. Detecting irregularities in images and in video. *International Journal of Computer Vision*, 74(1):17–31, 2007.
5. Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.
6. Hung Hai Bui. A general model for online probabilistic plan recognition. In *AAAI*, pages 1309–1318, 2003.
7. Xiaoyong Chai and Qiang Yang. Multiple-goal recognition from low-level signals. In *AAAI*, pages 3–8, 2005.
8. P. Domingos and M. Pazzani. On the optimality of the simple bayesian classifier under zero-one loss. *Machine Learning*, 29:103–130, 1997.
9. Mark R. Hodges and Martha E. Pollack. An 'object-use fingerprint': The use of electronic sensors for human identification. In *UbiComp*, pages 289–303, 2007.

10. Stephen S. Intille, Kent Larson, Emmanuel Munguia Tapia, Jennifer Beaudin, Pallavi Kaushik, Jason Nawyn, and Randy Rockinson. Using a live-in laboratory for ubiquitous computing research. In *Pervasive*, pages 349–365, 2006.
11. Henry Kautz. *A Formal Theory of Plan Recognition*. PhD thesis, University of Rochester, 1987.
12. Lin Liao. *Location-Based Activity Recognition*. PhD thesis, University of Washington, 2006.
13. Lin Liao, Dieter Fox, and Henry A. Kautz. Learning and inferring transportation routines. In *AAAI*, pages 348–353, 2004.
14. Lin Liao, Dieter Fox, and Henry A. Kautz. Extracting places and activities from gps traces using hierarchical conditional random fields. *IJRR*, pages 119–134, 2007.
15. Beth Logan, Jennifer Healey, Matthai Philipose, Emmanuel Munguia Tapia, and Stephen S. Intille. A long-term evaluation of sensing modalities for activity recognition. In *Ubicomp*, pages 483–500, 2007.
16. J.L. Meece and Holt K. A pattern analysis of student’s achievement goals. *Journal of Educational Psychology*, 85:582–590, 1993.
17. Keith Oatley. *Best Laid Schemes: The Psychology of Emotions*. Cambridge University Press, 1992.
18. Jeffrey Junfeng Pan, Qiang Yang, and Sinno Jialin Pan. Online co-localization in indoor wireless networks by dimension reduction. In *AAAI*, pages 1102–1107, 2007.
19. Sinno Jialin Pan, James T. Kwok, Qiang Yang, and Jeffrey Junfeng Pan. Adaptive localization in a dynamic wifi environment through multi-view learning. In *AAAI*, pages 1108–1113, 2007.
20. Donald J. Patterson, Dieter Fox, Henry A. Kautz, and Matthai Philipose. Fine-grained activity recognition by aggregating abstract object usage. In *ISWC*, pages 44–51, 2005.
21. Donald J. Patterson, Lin Liao, Dieter Fox, and Henry A. Kautz. Inferring high-level behavior from low-level sensors. In *Ubicomp*, pages 73–89, 2003.
22. Donald J. Patterson, Lin Liao, Krzysztof Gajos, Michael Collier, Nik Livic, Katherine Olson, Shiaokai Wang, Dieter Fox, and Henry A. Kautz. Opportunity knocks: A system to provide cognitive assistance with transportation services. In *Ubicomp*, pages 433–450, 2004.
23. Martha E. Pollack, Laura E. Brown, Dirk Colbry, Colleen E. McCarthy, Cheryl Orosz, Bart Peintner, Sailesh Ramakrishnan, and Ioannis Tsamardinos. Autominder: an intelligent cognitive orthotic system for people with memory impairment. *Robotics and Autonomous Systems*, 44(3-4):273–282, 2003.
24. Wentzel K. R. Social and academic goals at school: Motivation and achievement in context. *Advances in Motivation and Achievement*, 1991.
25. Wentzel K. R. Motivation and achievement in adolescence: a multiple goal perspective. *Student Perceptions in the Classroom: Causes and Consequences*, 1992.
26. Charles Sutton and Andrew McCallum. Collective segmentation and labeling of distant entities in information extraction. *University of Massachusetts TR 04-49*, 2004.
27. Charles A. Sutton, Khashayar Rohanimanesh, and Andrew McCallum. Dynamic conditional random fields: Factorized probabilistic models for labeling and segmenting sequence data. *JMLR*, 8:693–723, 2007.
28. Douglas L. Vail, Manuela M. Veloso, and John D. Lafferty. Conditional random fields for activity recognition. In *AAMAS*, pages 1331–1338, 2007.
29. Robert Wilensky. *Planning and Understanding: A Computational Approach to Human Reasoning*. Addison-Wesley Publishing Co., 1983.
30. Jie Yin, Qiang Yang, and Jeffrey Junfeng Pan. Sensor-based abnormal human-activity detection. *TKDE*, 2, 2008.