Distributed Activity Recognition using Key Sensors

A. M. Jehad Sarkar, Kamrul Hasan, Young-Koo Lee,
Sungyoung Lee, Salauddin Muhammad Salim Zabir
Dept. of Computer Engineering, Kyung Hee University, Korea.
{jehad, kamrul}@oslab.khu.ac.kr, yklee@khu.ac.kr, sylee@oslab.khu.ac.kr, szabir@ieee.org

Abstract

Recent development of sensor technology gives us the opportunity to effectively monitor daily activities of individuals. As such, in this paper we present a distributed technique to recognize Activities of Daily Living (ADLs) using simple sensors. We consider a number of randomly deployed sensors in home environment augmented with home appliances (e.g., cabinet, desk, chair etc.). Our proposal consists of three major steps. At first, in a random arrangement of sensors, their triggering pattern under human actions is recorded. These records are assembled for meaningful information. This is followed by the categorization of the key sensors (i.e., most important sensors) for each activity from the acquired knowledge. Finally, we group the sensors such that activity based hierarchical clusters can be formed. The system is thus ready for activity recognition. Experiments reveal that even for a small dataset, our proposal can find out the key sensors and form clusters. Also, it is observed that our proposed mechanism yields an accuracy of determination is more than 61%. In addition, it ensures distribution of processing loads among the sensors themselves and thus minimizes the centralized processing overheads.

I. Introduction

In ubiquitous home network, recognition of daily activities (Preparing lunch, Toileting, etc.) is one of the current focuses of Researchers. Especially in health care industry it has a huge impact. As an example, monitoring daily activities can reduce the risk of elderly people or chronically ill children.

Three approaches have been tried by the researchers to detect human activity: video based, wearable sensors based, and based on sensors deployed in the environment. Video based methods have the disadvantage of breaking user's privacy, whereas wearable sensors requires user to wear sensors and accuracy depends on the position of the attachments. Therefore, activity detection methods based on sensors deployed in the environment are getting more focus. We propose a method of detecting activity of daily living by finding out the key sensors in the environment.

A key sensor can detect an activity more accurately than any other approaches. By the key sensors we mean the sensors most frequently used for an activity. These sensors can be deployed in a tricky way in the environment and by doing so a high accuracy can be achieved. However, tricky deployment of sensor and using special deployment information makes an activity detection narrow and only suited for that specific environment. So, we avoid deploying key sensors and try to find out key sensors from an environment where sensors are deployed randomly. We take a set of training data to count the frequency of activities per sensor. As we mentioned before, the sensors are assumed to be simple sensors that can attain only two ON-OFF discrete states that we name as triggered and non-triggered respectively. We consider a sensor to be a key-sensor for an activity which occurs the most while the sensor is triggered. Several key sensors can be found for an activity. At a particular time instance, key sensors of different activities can be triggered. So, we use a vote counting mechanism for deriving the highest likely activity.

Wireless Sensor Networks [1] (WSN) are used in a large set of applications like, habitat monitoring, object tracking, precision agriculture, building monitoring and military systems [1], [2], [3]. These applications need to collect and aggregate data from a large number of sensor nodes consuming less amount of energy.

Clustering is one of the widely used techniques for data aggregation in WSN. In clustering, virtual clusters are formed and the cluster-heads are selected. Sensors send the packet to a cluster-head and the cluster-head then takes the responsibility to forward the packet to the sink [4]. We use the hierarchical clustering technique for data aggregation. Formation of the clusters is based on each activity, that is, each of the clusters will represent individual activity.

Unavailability of data set that represents activities makes it difficult to progress in activity recognition research. Real world data is hard to find. Even if we can obtain this, there will be noises. Existing deployments, like Place Lab [5] generates thousands of sensor data from the environment but very few activities are instantiated. This is because users are reluctant to interpret the sensor values that reflect activity. Additionally, there are varieties of ways to do the same activity. User's movements may not always be focused. That makes it even more difficult to produce a precise model. However, to some extent it is possible to observer a sequence of sensor activations that reflects the current activity of the user. Finding the key sensors and ignoring other sensor events, can contribute to the detection of the user activity. In this paper we show how to find the key sensors and depending on their values how we can determine the user activity.

In our approach we do not need high-performance machine as a base station. Base station is to be used only to compare few values. Therefore, the base station could be a sensor or a mobile device or PDA and of course it could be a personal computer. And the mobility of the base station is not an issue.

Our main contribution in this paper is the mechanism of finding key-sensors corresponding to a specific activity. If properly derived, a particular set of key-sensors can help recognizing the related activity accurately. Another major contribution of our proposal is the idea of knowledge-based hierarchical formation of clusters. In addition, we propose the use of time as a secondary parameter for activity recognition. Experiments show that our proposed mechanism yields an improvement in the accuracy of determination. Also, our proposal ensures distribution of processing loads among the sensors themselves. As such, the need for heavy centralized processing is avoided.

The rest of the paper is organized as follows. Related works on activity recognition is presented in section 2. We address our Learning Algorithm in section 3 followed by our Classifier in section 4. How we can use the knowledge obtained from our algorithms to sensor network is presented in section 5. In section 6 we present our experimental results to support our claims. Section 7 concludes our paper.

II. Related works

Many research groups have been investigating how to construct smart living environments that target medical care to the individual. Intel Research group in Seattle and the University of Washington have built a prototype system that can infer a person's activities of daily living (ADLs). Sensor tags are placed on everyday objects such as a toothbrush or coffee cup. University of Rochester is building the Smart Medical Home, which is a five-room house outfitted with infrared sensors, computers, bio-sensors, and video cameras for use by research teams to work with research subjects as they test concepts and prototype products. Georgia Tech built an Aware Home as a prototype for an intelligent space. Massachusetts Institute of Technology (MIT) and TIAX are working on the PlaceLab initiative, which is a part of the House_n project. The mission of House_n is to conduct research by designing and building real living environments—"living labs"—that are used to study technology and design strategies in context. Many projects are building body networks for the collection of vital signs, such as AMON. All these systems demonstrate the excitement and need for such systems.

According to our knowledge, [5] was the first to introduce activity recognition using simple and ubiquitous sensors in home settings. The author provides context-aware experience sampling tools (ESM) [6] to user to label their own activity. Nave Bayes classifier was used recognize activities. They have showed an excellent promise, even though their mechanism suffers from low recognition accuracy.

In [7] the authors consider sensor network in office environment. The concept of Hierarchical Feature Extraction is used to detect user's activity from aggregated sensor data. Nave Bayesian inference engine is used to take input from feature extractor and gives user's activity as output.

Activities can also be detected through audio, video sensors or body attached sensors. For example, [8] uses audio video sensors for aggression detection. They first perform independent analysis of the audio and video streams to get the descriptors of a scene like: "scream", "passing train" or "articulation energy". Next, they use Dynamic Bayesian Network as a fusion mechanism that produces an aggregate aggression indication for the current scene. In [9], they show how body attached sensors can be used to recognize activities of assembly tasks. The glitch of these approaches are namely, (i) difficulties in signal analysis, (ii) people are not always comfort able wearing sensors and (iii) expensive solution.

Also, many mobility based activity detection mechanisms have been proposed. For example, [10] uses Hierarchical Hidden Markov Model (HMM), Bayesian Filter is applied in [11], Dynamic Bayesian Network is deployed in [12] etc.

III. Our model for Activity Recognition

Our model as shown in the Figure 1 consists of two main components, training and inference. We will discuss each of the modules in subsequent subsections.

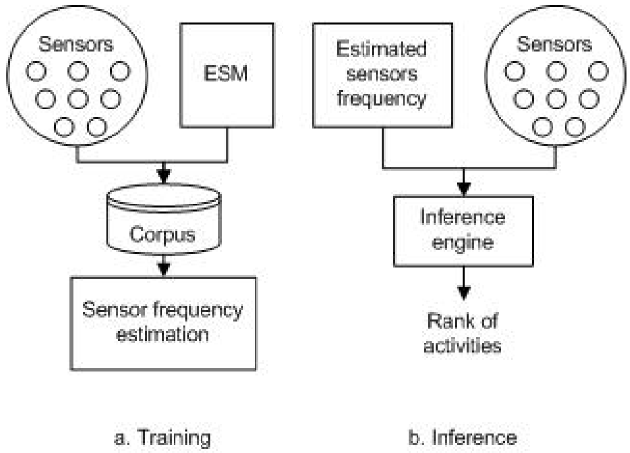

Fig. 1. a. Training (Sensor activation frequency/activity). b. Inference engine (Rank activities)

A. Training

Algorithm 1: Algorithm for creating sensors frequency

Data: Instance of activities I for training, List of sensors S in the environment, list of activities A to monitor
Result: list of estimated sensors frequencies sf (per activity) of size $m \times n$

$m = length(A)$;
$n = length(S)$;
for $a \leftarrow 1$ **to** m **do**
    for $s \leftarrow 1$ **to** n **do**
        sf$_{a,s}$ = 0;
    end
end
for $i \leftarrow 1$ **to** $length(I)$ **do**
    $aId = getActivityID(I_i)$);
    $a = getActivityIndex(aId, A)$;
    `/* Index of the given activity */;`
    for $s \leftarrow 1$ **to** n **do**
        if *isON*(I_i, S_s) **then** `/* isON = true if` s^{th} `sensor is triggered for the given instance of an activity */`
            sf$_{a,s}$ = sf$_{a,s}$ + 1;
        end
    end
end

At first, in a random arrangement of sensors, their triggering pattern under human actions is recorded through Experience Sampling Method (ESM). Which is followed by sensor frequency creation.

To create the sensor frequency we first create a histogram from the data set. We used the algorithm shown in Algorithm 1.

The algorithm first finds all the distinct activities from the given instances. It than finds all the rows that represent a activity and performs sensor wise summation. The result is put into a table (this is the output of this algorithm). This process continues for all the distinct activities. Each row of the frequency table represents each activity.

B. Activity inference

Algorithm 2 shows our proposed classifier. It takes input generated from our learning algorithm (i.e. frequency table) and the activity we are classifying. For each triggered sensor in the 'test activity', this algorithm first finds the activity from the 'frequency table' which has the maximum assessment for that sensor. It than increases the vote count for that activity. This process continues for all the triggered sensors. Finally the algorithm gives the activity as output which has the maximum vote.

Algorithm 2: Proposed classifier

Data: list of estimated sensors frequencies sf, activated sensors S, list of activities A
Result: predicted class of the given activity data, class

$[m, n] = size(sf)$;
    `/* m = number of activities, n = number of sensors */;`
for $a \leftarrow 1$ **to** m **do**
    vote$_a$ = 0;
    `/* vote`$_a$ `= vote count for activity a */;`
end
for *each sensors s in S* **do**
    $i = getSensorIndex(s)$;
    $[data, a] = max(sf(:, i))$;
        `/* a = Index of activity */;`
    $vote_a = vote_a + 1$;
end
$[data, i] = max(vote)$;
class = indexToActivityID(i, A);

IV. Distribution of Knowledge to WSN

To meet the unique requirements for knowledge distribution, we developed our clustering algorithm as an application-specific protocol architecture [13], [14]. In our case, we choose the correlation between sensors based on the prior knowledge from the classifier. We chose to use a clustering infrastructure as the basis for our algorithm.

This allows all data from nodes within the cluster to be processed locally, reducing the data set that needs to be transmitted to the end user [15].

A. Cluster Formation Algorithm

For development of our clustering algorithm we made some assumptions toward WSN.
i. Each of the sensors have unique ID,
ii. Cluster heads has the ability to reach sink node (Base Station),
iii. Cluster members can reach cluster head,
iv. Sensor can store and perform simple mathematical operations,
v. Sensor nodes are deployed in predefined fashion,
vi. Symmetric propagation channel.
Unique IDs for sensors is defined before deployment. Once IDs are defined, we deploy the sensors randomly in the environment. At this point, the sensor network enters into the training phase. During the training if a sensor triggers, it sends activation information to the base station along with time. After the training phase, we group the sensors such that the clusters are formed based on the knowledge we get from our classifier. That is, we define a cluster based on one activity or two (or more) similar activities. Example of similar activities can be preparing dinner, lunch, breakfast, etc. These activities trigger similar sensors but at different times. The nodes with highest number of activation for one activity (or one or more similar activities) are the cluster members. And the cluster head is the node with maximum number of activation among cluster members. That is, the total number of clusters is the total number (at most) of activities we are considering. For example, if we consider ten activities, total number of clusters will be at most 22 and there will be at most 10 cluster heads. It implies that after grouping all the nodes know their cluster head and the cluster members.
All nodes of a cluster will communicate through Direct-Sequence Spread Spectrum (DSSS). All the clusters have their unique spreading code. Cluster members transmit their data to cluster head using this code. This ensures the reduction of inter-cluster interference. This is known as, *transmitter-based code assignment* [16]. At this point our algorithm is similar to [15].

B. Working principle

Working principle of our clustering algorithm is simple. After deployment and grouping, if a sensor node triggers, it will send a binary value '1' to its cluster head. Cluster head will then aggregate all the sensor data and send the sum to the base station along with time. Base station collects data from all the clusters heads and determines the activity based on the maximum value it has collected from different cluster heads. It will distinguish similar activities based on time. In addition to this it will track the previous activity to increase classification accuracy. For example, if the recent activity was "preparing lunch" but the current output of the sensors indicating "preparing lunch", it will override the output with "putting away dishes". And finally deliver the activity to the user(s).
We also consider re-clustering. After a predefined time (We will define the optimal predefined time based on the activation trends of the sensors) the cluster head role will be changed. And the node with 2nd maximum number of activation for that cluster will be chosen as the cluster head for that round and so on. This ensures the proper distribution of energy in WSN and therefore, prolongs the network lifetime.

V. Evaluation

We consider the setup prepared by MIT Place Lab [17] for activity recognition. In their experiment [6], between seventy seven and eighty four sensor data collection boards equipped with reed switch sensors installed in two single-person apartments collecting data about human activity for two weeks. The sensors were installed in everyday objects such as drawers, refrigerators, containers, etc. to record opening-closing events (activation deactivation events) as the subject carried out everyday activities. The data was collected by a base station (central server) and labeled using Experience Sampling Method (ESM)[6]. We get activity name and sensor values for that activity from the given data set and construct a table. We use this table throughout our experiment. Each row of that table represents the number of times a sensor triggers for each activity. Table I shows the format of our data set. We use this as our corpus for our research.
MIT Place Lab collected activity data about both Subject one and Subject two for two weeks. Format of their data set is as follows,

TABLE I. Example data set

Activity \ Sensor	Toilet Flush	Sink Faucet	Closet	Light switch	Shower Faucet
Bathing	1	1	2	1	4
Bathing	0	2	1	1	3
Bathing	0	2	3	2	5
Toileting	1	2	1	1	0
Toileting	1	3	2	2	0
Toileting	1	1	1	1	1

ACTIVITY_LABEL, DATE, START_TIME, END_TIME

SENSOR1_ID, SENSOR2_ID, ...,

SENSOR1_OBJECT, SENSOR2_OBJECT, ...

SENSOR1_ACTIVATION_TIME, SENSOR2_ACTIVATION_TIME,...

SENSOR1_DEACTIVATION_TIME, SENSOR2_DEACTIVATION_TIME,...

Date is in the mm/dd/yyyy format, time is in the hh:mm::ss format. Here is an example of one activity,
Toileting,4/1/2003,11:52:1,11:58:50
100,137
Toilet Flush,Freezer
11:55:43,11:56:2
16:35:49,11:56:13
For our experiment we use Visual C++ 2005 Express Edition to process this data such that we get output like,
ACTIVITY_LABEL START_TIME END_TIME SENSOR1_ID SENSOR2_ID ...
For our convenience we replaced the sensor ids with 1, 2, 3 ... We have seen that seventy six sensors were present for both Subjects in their data set. For rest of our experiment we use Matlab version 7.0.1 with Service Pack 1.
[17] recorded twenty two activities for Subject one and twenty four activities Subject two. However, we only work with thirteen activities of Subject one and ten activities for Subject two. We removed activities with less than four examples. We use "leave one day out" strategy to separate the training and testing dataset. In this strategy, one day was used for testing and remaining days were used for training.
Table II and III shows the accuracy per activity for Subject one and Subject two respectively. We achieved 61.51% and 55.11% of accuracy as shown in the Tables.

TABLE II. Output of the classifier from Subject One

Activity	Correctly Classified	Miss Classified	Accuracy
Going out to work	0	12	0
Toileting	56	29	65.882
Bathing	6	12	33.333
Grooming	33	4	89.189
Dressing	19	5	79.167
Preparing breakfast	8	6	57.143
Preparing lunch	12	5	70.588
Preparing dinner	7	1	87.5
Preparing a snack	5	9	35.714
Preparing a beverage	7	8	46.667
Washing dishes	3	4	42.857
Cleaning	0	8	0
Doing laundry	15	4	78.947
	171	107	61.511

Here it is to be noted that, we use time shown in Table IV to distinguish similar activities like lunch, dinner, etc.

VI. Conclusion and Future work

In this paper we present an algorithm for recognizing activity of daily living using wireless sensor network. We consider that the sensors are randomly deployed in home environment. We show how our proposed classifier can be used to find the key sensors per activity. Also show how hierarchical clusters can be formed based on these sensors to detect high level context. Our proposed mechanism yields an improvement in the accuracy of determination by more than 61%. In addition, our proposal ensures distribution of processing loads among the sensors themselves and thus minimizes the centralized processing overheads. In addition to this we will collect data. As an extention of our work we are considering a multi-user environment. Also, the same idea can be employed to other activity recognition areas such as, employees at office, patients at hospital etc.

TABLE III. Output of the classifier from Subject Two

Activity	Correctly Classified	Miss Classified	Accuracy
Toileting	34	6	85
Taking medication	1	13	7.1429
Preparing breakfast	17	1	94.444
Preparing lunch	9	11	45
Preparing dinner	8	6	57.143
Preparing a snack	4	12	25
Washing dishes	11	10	52.381
Watching TV	7	8	46.667
Listening to music	6	12	33.333
	97	79	55.114

TABLE IV. Time as a secondary parameter

Activity	Start	End
Preparing Breakfast	05:00:00	09:00:00
Preparing Lunch	11:00:01	14:00:00
Preparing Dinner	18:00:01	21:00:00
Preparing Snacks	09:00:01	11:00:00
Preparing Snacks	21:00:01	04:00:00
Preparing Beverage	02:00:01	18:00:00

VII. Acknowledgment

This research was supported by the MKE (Ministry of Knowledge Economy), Korea, under the ITRC(Information Technology Research Center) support program supervised by the IITA(Institute of Information Technology Advancement) (IITA-2008-C1090-0801-0002) and by the MIC(Ministry of Information and Communication), Korea, Under the ITFSIP(IT Foreign Specialist Inviting Program) supervised by the IITA, C1012-0801-0003. Also, this work is financially supported by the Ministry of Education and Human Resources Development (MOE), the Ministry of Commerce, Industry and Energy (MOCIE) and the Ministry of Labor (MOLAB) through the fostering project of the Lab of Excellency.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Comput. Netw.*, vol. 38, no. 4, pp. 393–422, 2002.

[2] A. Cerpa, J. Elson, M. Hamilton, J. Zhao, D. Estrin, and L. Girod, "Habitat monitoring: application driver for wireless communications technology," in *SIGCOMM LA '01: Workshop on Data communication in Latin America and the Caribbean*. New York, NY, USA: ACM, 2001, pp. 20–41.

[3] G. J. Pottie and W. J. Kaiser, "Wireless integrated network sensors," *Commun. ACM*, vol. 43, no. 5, pp. 51–58, 2000.

[4] X. Renyi and W. Guozheng, "A survey on routing in wireless sensor networks," *Progress in Natural Science*, vol. 17, no. 3, pp. 261–269, March 2007.

[5] S. S. Intille, K. Larson, E. M. Tapia, J. Beaudin, P. Kaushik, J. Nawyn, and R. Rockinson, "Using a live-in laboratory for ubiquitous computing research," in *Pervasive*, 2006, pp. 349–365.

[6] E. M. Tapia, S. S. Intille, and K. Larson, "Activity recognition in the home using simple and ubiquitous sensors," in *Pervasive*, ser. Lecture Notes in Computer Science, A. Ferscha and F. Mattern, Eds., vol. 3001. Springer, 2004, pp. 158–175.

[7] C. R. Wren and E. M. Tapia, "Toward scalable activity recognition for sensor networks," in *LoCA*, 2006, pp. 168–185.

[8] W. Zajdel, J. D. Krijnders, T. Andringa, and D. M. Gavrila, "Cassandra: Audio-video sensor fusion for aggression detection," in *IEEE Int. Conf. on Advanced Video and Signal based Surveillance (AVSS*, 2007.

[9] "Activity recognition of assembly tasks using body-worn microphones and accelerometers," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 10, pp. 1553–1567, 2006, student Member-Jamie A. Ward and Member-Paul Lukowicz and Senior Member-Gerhard Troster and Member-Thad E. Starner.

[10] S. Lühr, H. H. Bui, S. Venkatesh, and G. A. W. West, "Recognition of human activity through hierarchical stochastic learning," in *PERCOM '03: Proceedings of the First IEEE International Conference on Pervasive Computing and Communications*. Washington, DC, USA: IEEE Computer Society, 2003, p. 416.

[11] D. J. Patterson, L. Liao, D. Fox, and H. A. Kautz, "Inferring high-level behavior from low-level sensors," in *Ubicomp*, 2003, pp. 73–89.

[12] D. J. Patterson, D. Fox, H. Kautz, and M. Philipose, "Fine-grained activity recognition by aggregating abstract object usage," in *ISWC '05: Proceedings of the Ninth IEEE International Symposium on Wearable Computers*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 44–51.

[13] W. B. Heinzelman, "Application-specific protocol architectures for wireless networks," Ph.D. dissertation, 2000, supervisor-Anantha P. Chandrakasan and Supervisor-Hari Balakrishnan.

[14] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *HICSS '00: Proceedings of the 33rd Hawaii International Conference on System Sciences-Volume 8*. Washington, DC, USA: IEEE Computer Society, 2000, p. 8020.

[15] A. P. Chandrakasan, A. C. Smith, W. B. Heinzelman, and W. B. Heinzelman, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on Wireless Communications*, vol. 1, pp. 660–670, 2002.

[16] L. Hu, "Distributed code assignments for cdma packet radio network," *IEEE/ACM Trans. Netw.*, vol. 1, no. 6, pp. 668–677, 1993.

[17] E. M. Tapia, S. S. Intille, and K. Larson, "Mit activity recognition data," 2004. [Online]. Available: http://courses.media.mit.edu/2004fall/mas622j/04.projects/home/