International Journal of Computer Information Systems and Industrial Management Applications.
ISSN 2150-7988 Volume 5 (2013) pp. 345-353

Human Activity Monitoring with Wearable Sensors and Hybrid Classifiers

Gamze Uslu[1], **H.Ibrahim Dursunoglu**[2], **Ozgur Altun**[3] **and Sebnem Baydere**[4]

Department of Computer Engineering,
Yeditepe University, Istanbul, Turkey
[1]*guslu,* [2]*hdursunoglu,* [3]*oaltun,* [4]*sbaydere@cse.yeditepe.edu.tr*

Abstract: **Activity monitoring plays a crucial role in ambient living environments for assessing changes in the normal behavioral pattern of elderly people. In this paper, we present an action description and detection mechanism for real-time activity monitoring using wearable sensors and hybrid classifiers. First a single sensor single classifier model is presented (SSSC) for the detection of simple and composite actions. Then the model is enhanced with multiple sensors and classifiers for the purpose of real-time monitoring. The enhanced Multi-Sensor Multi Classifier (MSMC) model uses two wearable TI Chronos watches with a built-in tri-axial accelerometer for data acquisition and a composition of naive Bayes, Susan Corner Detector(SCD) and Hidden Markov(HMM) classifiers for the detection of transitions between defined actions in real-time. A real testbed environment is established to asses the success of real-time monitoring. The test results have revealed that SSSC model is highly successful in controlled tests when the burden of real-time sampling is ignored whereas MSMC model is fast and accurate for real time detection of transitions between actions. The proposed models are tested against the simple actions; *walk, sit, stand, lie* as well as *walk-while-hands-in-pocket* and *walk-on-wheelchair*. The unique feature of the selected actions is that the transition between walk, sit and lie are the most likely causes of a fall event in a home environment for elderly people. The best achieved detection rates for simple actions range between 92-100 % for SSSC model whereas MSMC model is 100 % successful in real-time detection of transitions with a slightly reduced achievement for individual actions.**

Keywords: activity monitoring, hidden markov model, susan corner detection, hybrid classifier, naive bayes, chronos

I. Introduction

Assistive technology provides solutions to people with disabilities and aging population in performing tasks without being helped by another person. Even if a person is not suffering from disabilities or aging, they still can benefit from assistive technology tools and services. As a branch of assistive technology, ambient care systems are emerging. To aid everyday life of people in need, ambient care systems contain a network of objects used in people's daily routines. Ambient care systems are capable of sensing the environment through sensors and reacting to certain conditions reasoned in the network mentioned. The ultimate goal of an ambient care system is presenting the assistive technology by meeting the following criteria: Devices in the system should be embedded in the surroundings or should be wearable. The constituents of the system should be able to detect the person being serviced and his conditions, the so-called context awareness principle. The system should also be adjustable to the personal needs. It should adapt itself depending on the reactions of the person. It should understand when it is needed and consequently act as needed without the person's intrusion, namely principle of being anticipatory.

In the field of ambient care systems and more generally assistive technology, activity monitoring plays a vital role in terms of taking decisions on when to make the system respond in what way.If the action performed by a person can be identified, this reveals the information regarding what the person needs or wants, so that his needs are met by the ambient care system.The person can be reminded of taking his medication if he forgets to do so or if the detected action reveals that *he is about to fall*, he may be prevented from falling or from a more severe situation.

There are various technical challenges for the design of activity monitoring systems. Since even the same person does not perform the same activity in the same way all the time and some different actions may exhibit similar characteristics, there is a potential deterioration in the recognition accuracy. Noise in the activity signal, namely differentiating between the noise and the actual signal causes problems as well. Enhancing an activity monitoring system includes detecting abnormal activities defined in accordance with the context and providing the appropriate actuation facilities in response.

Activity monitoring can be achieved in two phases; data collection followed by data classification. Data collection process is carried out through wireless sensors, cameras, PDA's or other health care monitoring devices[1]. The devices which do not intrude into the privacy of the person to be monitored can have an advantage over the devices like cameras. Wireless sensor networks (WSN) can also improve the efficiency of data collection phase. For detection, various classification methods can be used such as least squares[2], k-nearest neighbor (k-NN)[3], hidden markov, artificial neural networks (ANN)[4] and support vector machines (SVM)[5]

In this experimental study, an indoor activity monitoring system is designed and implemented to recognize the simple actions performed by a human subject and the transitions between these actions in real time. Two models are analyzed: single sensor single classifier (SSSC) and multi sensor multi classifiers(MSMC). In both models, sensor readings from a tri-axial accelerometer built-in the TI Chronos watch is used for data acquisition.

In the single sensor-single classifier (SSSC) model, the watch is worn by the person on the left wrist for the walk action, and worn to the left thigh for the sit, stand and lie actions. The resulting sensor data obtained in the form of unsigned integers varying in the range [0,255] are converted to their 2's complement equivalents. The acceleration values in 2's complement form are classified by using naive Bayes classifier into a simple action. Naive Bayes classifier has training and prediction phases. In the training phase, the training data are exposed to normal distribution to extract unique intervals of average posterior probability. These intervals create the pattern for the specified action. Patterns for all simple actions are recorded into a database. In the prediction phase, data sample of a composite action with unknown type is detected by comparing the differentiated simple actions to the patterns in the database to produce a posterior probability value. The action of which average posterior probability value is included in one of the distinct intervals is marked as the corresponding action.

The multi sensor-multi classifier (MSMC) model includes a data collection component composed of two acceleration sensors built in TI Chronos watch worn on wrist and ankle by the human subject. It also includes a composite data classification subsystem employing naive Bayes classifier first to differentiate between activities. When naive Bayes classifier reports that an action performed by a person is ambiguous ie. may be recognized as more than one action, then the data classification component uses Hidden Markov Model (HMM) and Susan corner detector (SCD) in order to find a single answer to what the action is. Data classification has training and prediction phases. During the training phase, the system processes the samples of the actions to be recognized, generating patterns for those actions. In the prediction phase, the system processes real time data, splitting the data into chunks on the fly and inferring what action each chunk belongs to by evaluating the patterns obtained in training phase. Real-time activities are modeled as simple and composite actions. Simple actions are walk, walk while hands in pocket, stand, sit, lie and wheelchair driving. Any combination of these actions are regarded as composite actions. The system is trained with simple actions whereas the real time data processed during prediction phase are composed of composite actions. The presented work is also featured by its successful differentiation between different kinds of walk actions, namely walk, walk while hands in pocket and walk on wheelchair.

The content of the following sections is as follows: In Section II related work is reviewed. In Sections III-A and III-B, single sensor-single classifier and multi sensor-multi-classifier models are given. Section IV reveals the results of the experiments and elaborates on the future work.

II. Related Work

Human activity monitoring has started to be used in a wide area. Even though the most common tool used for monitoring activities is camera, it causes computational load as the number of people being monitored increases. Another widely used method is PIR sensors. PIR sensors are used in thermal imaging, radiometry, thermometry and biometry. Achieving coverage, video surveillance assistance and tracking exploit multiple PIR sensors. A still person can be distinguished from its background with PIR sensors. A video surveillance system with multi-modal sensor integrity has been put forward where a tracking system with multiple cameras is united with a wireless sensor network supported with PIR sensors. Apart from tracking, PIR sensors are beneficial for detection, differentiation and describing human activity. The works discussed so far are restricted as a result of the need for synchronizing the time accurately for sensor nodes and the related communication cost. There exists a study which tries to solve these issues by supporting each sensor node with two PIR sensors to synchronize the data which are sampled from PIR sensors and communication cost problem [6].

A method to extract the meaning from the raw sensor data directly on the sensors is presented by Kay Römer in [7]. A multi-modal sensor system for monitoring human activities is developed by Hung et. al. in [6]. Bao and Intille developed and evaluated algorithms to detect physical activities and used biaxial accelerometers worn simultaneously on various parts on the body for data collection in [8]. Zhu and Sheng propose a human daily activity recognition method by fusing the data from two wearable inertial sensors attached on one foot and the waist of the human subject, respectively in [9]. Lymberopoulos et. al. present an automated methodology for extracting the spatiotemporal activity model of a person using a wireless sensor network deployed inside a home in [10]. Bosch et. al. implement and evaluate physical activity monitoring and stimulation using wireless sensor networks and motion sensors in [11]. Uslu et.al [12] implemented a single sensor single classifier system for monitoring human activities.

Because data gathered from sensors may not be trustworthy some researches are carried out targeting sensor data accuracy issue. Hong et. al. address effects of sensor data uncertainty on decision making through information handling techniques such as Dempster-Shafer theory of evidence and Equally Weighted Sum operator [13]. Context-aware applications introduce high complexity due to changing context information, changing quality and uncertainty of sensor data. This complexity necessitates integrating context modelling and reasoning techniques to context aware applications which also improve maintainability and evolvability along with reducing complexity. For this reason, Bettini et. al. compare current context modelling and reasoning techniques [14]. J. Ye et. al. study situation identification as a way of coping with uncertainty of sensor data and maps noisy sensor data to patterns useful for applications. They also analyse complexity of situation identification and compare most common situation identification techniques [15].

Sazonov et. al. present a shoe sensor because distributing multiple sensors on the body can be too obtrusive. Their method can operate without feature extraction to classify postures and typical activities with respect to heel acceleration and plantar pressure [16]. Chen et. al. employ domain knowledge, ontology and semantic reasoning in activity recognition [17]. People suffering from aging problems or stroke need continuous physical therapy so activity monitoring systems offer help for physiatrists. Chiang et. al. propose an activity monitoring procedure to assess movements of patients to see whether they abide what therapy necessitates or not. Their system utilizes WSN based body motion sensors containing accelerometer and gyroscope along with fuzzy algorithm to differentiate between static postures and dynamic motions [18]. Ward et. al. propose performance metrics for continuous activity recognition [19]. Czabke et. al. carry out real time activity monitoring with data acquired from a tri-axial accelerometer and processed on a microcontroller. Their solution does not require a specific positioning of sensor and they classify actions with no training data [20]. To the best of our knowledge, current studies in the literature do not particularly focus on capturing transitions between actions on a real-time sequence. Rather they focus on detecting predefined specific actions.

III. Methodology

We have established an experimental setup to analyze the proposed SSSC and MSMC models. In both models TI Chronos watch which contains a 3D accelerometer is used for the data collection in training and prediction phases. The watch communicates with the PC through an access point (AP) over its RF interface operating SimpliciTI protocol stack. Using multiple Chronos' with a single AP brings synchronization problem which causes data loss. For the AP to receive data from multiple Chronos' in MSMC model, we reprogrammed it. Sensor readings are transmitted from Chronos to AP after all Chronos' establish a connection with the AP. A master-slave communication protocol is developed between AP and Chronos watches. As a result of this scheme, while one Chronos transmits to AP, others do not send any data. In addition, the received data are buffered. Since Chronos sends twenty acceleration vectors per transmit, the size of the buffer which is the size of a chunk extracted from real time activity data includes twenty vectors.

The classification module of SSSC uses only naive Bayes classifier whereas MSMC uses naive Bayes classifier as the initial classifier but upon conflicting detection results, SCD and HMM are used to resolve the conflict in real-time. The classifiers naive Bayes and SCD evaluate standard deviation and mean of training data under normal distribution to generate patterns for actions. The system is designed to monitor daily activities of a human subject by recording the number of repeated actions and transitions between actions. Training of HMM proceeds with calculating the probability of moving to the next state by using the repeated actions and transition counts recorded in the database. In the prediction phase, naive Bayes classifier processes the real time data to evaluate the posterior probability value to be compared with the posterior probability of other actions. If the posterior probability of the real time data corresponds to more than one action, SCD compares the distance value of the real time data with other actions. With the condition that SCD is unable to classify, HMM assesses transitions, observations and last defined action. In the following sections, the algorithms used in both models are explained in detail.

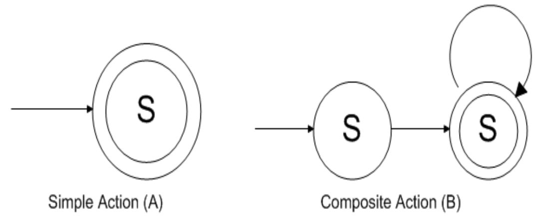


Figure. 1: Formal Representation of Simple and Composite Actions

Notations used in Algorithms 1-8 are as follows:
N : Number of simple actions
K : Number of samples taken for each action
S : Number of acceleration vectors in corresponding sample
C : Number of repetitions for an action
card(H) : Number of elements in set H
R : Set of chunks taken from real time data
AC_i : Simple action *i*, with i ϵ 1, 2, ..., N
T_{tr} : Training sample
μ_l : Mean
σ_l : Standard deviation
A_l : Array of acceleration vectors along axis *l*, *l* ϵ $\{$X, Y, Z$\}$
$C_z[.]$: Test data along Z axis
$T_z[.]$: Training data along Z axis
A : Array of acceleration vectors
$postProb[.]$: Array of posterior probability values
npdf : Normal probability density function

A. Single sensor-single classifier approach

In this approach, continuous activity monitoring is achieved through detecting successive simple actions in a collected data samples. The collected data sample from a single Chronos watch is composed of lines each of which contain a vector of X,Y and Z axis acceleration values. Simple actions, which can not be split into other actions, are stored in database and actions contained in collected data are classified into these actions by utilizing naive Bayes classifier. The defined simple actions are ***walk***, ***sit***, ***stand*** and ***lie***, whose content are implied by their names. The collected data sample holds a composite action since it contains multiple simple actions. Simple actions which are performed sequentially produce composite actions. The sample composite actions experimented are ***sit-after-lie*** and ***walk-after-sit***. The formal definitions of simple and composite actions can be seen as finite state automata as given in Figure 1. The state S represents a simple action in the automata. At the end of the process, the collected data sample is divided into chunks which are labeled as one of the actions recorded in the database. The size of each chunk is determined dynamically and the size value is used to figure out how long the action related to that chunk lasts. This is where the innovation of the proposed approach is. Figure 2 illustrates a picture of the test environment. In this study, all simple and composite action data samples are obtained in a controlled test environment.

Training phase of the Naive Bayes classifier consists of per-

Figure. 2: Walk and Sit-after-Lie Actions

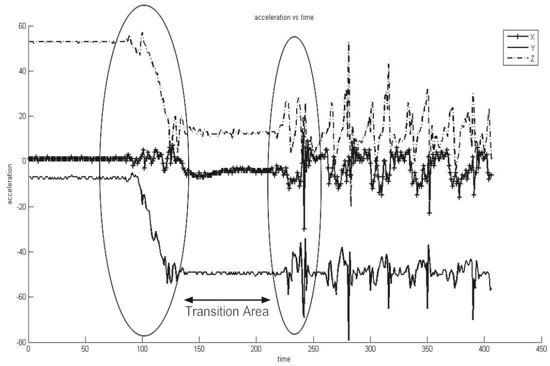


Figure. 3: Walk-after-Sit Data Sample

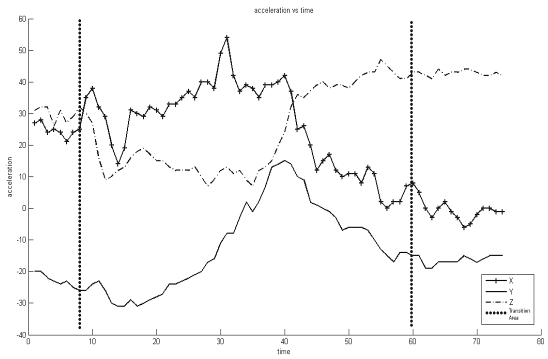


Figure. 4: Sit-after-Lie Data Sample

forming the same action many times, acquiring distinct intervals for every simple action, finally storing these intervals in the database. In our experiments, every action is repeated 25 times. The intervals contain average posterior probability values. Once the training phase is complete, the prediction process takes place. The algorithm which generates the distinct intervals in training phase is also used in the prediction phase to generate an average posterior probability value. In the prediction phase, collected data is processed by extracting a chunk from it in each iteration. *n* being the current iteration number, C_n being chunk at iteration n, c_i being *i*th line in the collected data, iteration number ranging from one to number of lines in collected data, C_n = *[*c_1,c_2,c_3,...,c_n*]* shows the structure of a chunk extracted from collected data at any iteration. Chunk at iteration number one is exposed to the same interval generation scheme in training phase and depending on the resulting average posterior probability value is in which interval, the chunk is classified into the corresponding action. As long as the following iterations generate the same action classified as the first iteration, the iterative procedure continues, otherwise iteration terminates. The chunk at the instance of termination is the ultimate structure representing a simple action classified in collected data and the iteration number at the instance of termination is the size of that chunk. After the first chunk is classified, collected data is truncated so that it starts with the vectors right after the first chunk and chunk classification continues until there are no vectors left in collected data.

Algorithm 1 Average posterior probability calculation

$A \leftarrow C_z[.] \bigcup T_z[.]$
$(\mu, \sigma^2) \leftarrow normalDistribution(A)$
$postProb[.] \leftarrow npdf(T_z[.], \mu, \sigma^2)$
$avg \leftarrow average(postProb[.])$

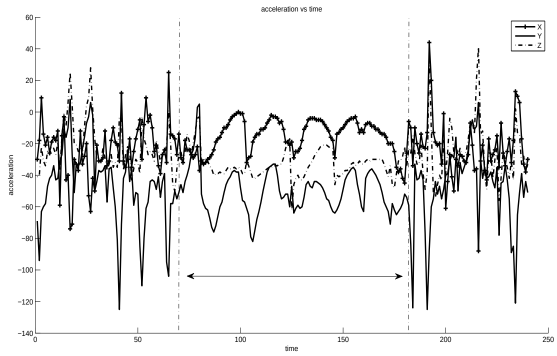
Average posterior probability of training sample T or a chunk C from real time sample is calculated as in the Algorithm 1 While naive Bayes classifier can be implemented as assigning the action yielding greatest posterior probability as the action detected, this work follows the approach that generating unique intervals for every action out of posterior probability values and regards these unique intervals as the differentiating parameter.

There are a number of reasons why normal distribution is chosen in this work during posterior probability generation process: First, the normal distribution provides simplicity since practically there are many cases where a population who does not fit normal distribution is successfully processed under the normal distribution. Second, as the size of the population increases, the probability distribution becomes more similar to the normal form. Particularly the second point is a strong reason because the length of the training data can reach the order of thousands.

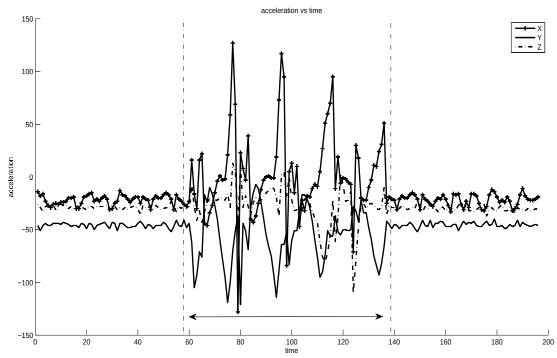
Acceleration vs time relationship for the composite action ***walk-after-sit*** are depicted in Figure 3. Here, time is represented virtually by the number of the acceleration vectors collected during the action. Higher number of vectors resembles an action of longer duration. With the transmission frequency of 33 packets per second, approximately 2.5 s corresponds to 45 vectors. During this approximate calculation, lost packets caused by Chronos are ignored. Figure 3 illustrates the collected data sample for the ***walk-after-sit*** action. In these plots, the action occurs as the combination of simple actions sit, stand and walk respectively. Since the aim is detecting walk after sit rather than the sequence of sit, stand and walk, during the tests, the section of the signal showing the stand action is ignored by filtering that segment, regarding stand as the transition. The data is obtained by appending walk samples to the end of sit samples. The training data related to the simple walk action are collected by wearing sensor on the left wrist whereas the composite action is performed with the sensor on the left thigh. The transition signal samples for the ***sit-after-lie*** action are also illustrated in Figure 4.

(a) Wheelchair Test (b) Stand Test

Figure. 5: Sensor placement on human body



(a) Walking test data sample for arm sensor



(b) Walking test data sample for foot sensor

Figure. 6: Walking real time test data samples

1) Training

Training data are exposed to normal distribution to extract mean and standard deviation for the Z axis. These values are used to calculate average posterior probability which form the pattern of an action. A pattern is created for every action and all patterns are inserted to a database to be used in the prediction phase later.

2) Prediction

A collected data sample of an action whose type is unknown is compared to all of the action patterns in the database, producing a posterior probability value. In posterior probability calculation, normal probability density function values found for the Z axis are used. To evaluate a normal probability density function value, the acceleration data obtained from the collected data sample, the mean and standard deviation values related to the specified axis are used.

B. Multi sensor-multi classifier approach

MSMC model focuses on the detection of transitions. Detecting transition between activities can not always be done with SSSC model. In the cases where SSSC model is insufficient to generate a unique class for an action, using multiple sensors help generating such a unique class by enlarging the feature space. Thus the use of multiple sensors decreases the probability of an action overlapping with the unique features of more than one action. Also, hybrid classifiers have the potential to improve the detection success further.

Algorithm 2 Main classification module

for $i <= N$ **do**
  $vectorCount \leftarrow 0$
  for $j <= K$ **do**
    for $m <= S$ **do**
      $vectorCount \leftarrow vectorCount + 1$
      A[vectorCount]$\leftarrow T_{tr}$[m]
    end for
  end for
end for

Actions are classified according to the location of Chronos watch yielding the signal of that action. One Chronos is worn on right ankle whereas the other one is worn on right wrist as in Figure 5(a) and 5(b). Plots of walking and wheelchair driving are shown in Figure 6(a), Figure 6(b), Figure 7(a) and Figure 7(b). The sensor on right ankle forms the foot oriented detection subsystem and the sensor on the right wrist forms hand oriented detection subsystem. System distinguishes the Chronos watches according to first bit of dataset. Signals emitted from both Chronos' are combined after separate recognition.

The main classification module is shown in Algorithm 2. This algorithm computes the mean μ and standard deviation σ of the training data along X, Y and Z axes.

In the prediction phase of naive Bayes classifier as shown in Algorithm 4, a chunk from real time data is compared to the unique intervals generated for each action. If a single class is obtained for that chunk, detection becomes complete, otherwise classification proceeds with HMM or SCD.

Training phase of naive Bayes classifier as given in Algorithm 3 calculates the posterior probability value, namely pattern P_j for each sample taken for every action through mean μ and standard deviation σ of the acceleration data along X and Z axes since these are the dominant axes for the actions within the scope of this work. Range[i,j] indicates

Algorithm 3 Training phase of naive Bayes classifier

for $i = 1 \rightarrow N$ **do**
  for $j = 1 \rightarrow K$ **do**
    $P_j \leftarrow f(\mu_{i(X,Z)}, \sigma_{i(X,Z)})$;
    $Range[i, j] \leftarrow P_j$;
  end for
end for

Algorithm 4 Prediction phase of naive Bayes classifier

> **for all** $j \epsilon R$ **do**
> $countNonMatching \leftarrow 0$
> $countMatching \leftarrow 0$
>   **for** $i = 1 \rightarrow N$ **do**
> $P_j \leftarrow f(\mu_{i(X,Z)}, \sigma_{i(X,Z)})$
>   **if** (**then**$P_j \epsilon Range[i]$)
> $countMatching \leftarrow countMatching + 1$
>   **else**
> $countNonMatching \leftarrow count + 1$
>   **end if**
>   **end for**
>   **if** $countNonMatching = N$ **then**
>   call HMM scheme
>   **else**
>   **if** $countMatching > 1$ **then**
>   call SCD scheme.
>   **else**
>   Classify segment j as action i
>   **end if**
>   **end if**
> **end for**

the range of posterior probability values which belong to the action *i*, j indicating the sample index related to the action *i*. SCD algorithm maps data to a nucleus whose center is equal to μ_Y and radius is equal to σ_Y. Algorithm 5 and 6 explain how SCD scheme fulfils classification in training and prediction phases.

The chunks from real time data which can not be defined with naive Bayes classifier and SCD are exposed to HMM scheme. HMM considers two events ***last action*** and ***next action***. Last action is the last detected action whereas next action is the current action to be detected. If the last action is not defined, as the last action HMM assigns the action with greatest observation probability, in our case sit whose observation probability is shown in Table 1.

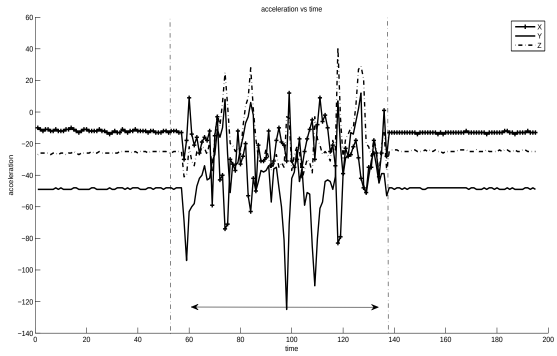
Having an action as the last action is important since HMM needs to find the transition probability from last action to next action. After assigning last action, HMM estimates the next action using (1).

$$L(n|l,o) = P(n|l).P(n|o) \qquad (1)$$

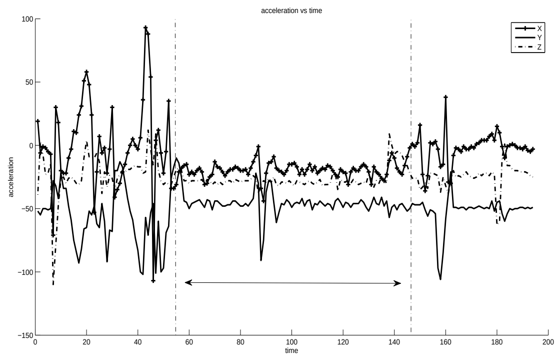
Interpretation of (1) is as follows: $L(n|l,o)$ indicates probability that the next action is n, given that l is the last action and o is the observation probability of last action. $P(n|l)$ shows the probability of obtaining n as the last action given that last action is l. Finally, $P(n|o)$ is the probability of obtaining n as the last action given that o is the observation probability of last action. HMM checks transition probabilities from last action to every action. Transition probabilities between actions are depicted in Table 2 where transition column indicates actions between which the transition occurs

Table 1: Observation probabilities of simple actions

Action Name	Observation Probability
Walk	0.35
Sit	0.45
Stand	0.2



(a) Wheel chair test data sample for arm sensor



(b) Wheel chair test data sample for foot sensor

Figure. 7: Wheel chair real time test data samples

such that Walk-Sit means the transition from walk to sit. In the case of potential next actions that generate equal transition probabilities, the action which yields greatest observation probability is selected to be the next action. HMM training and prediction phases are presented in Algorithm 7 and 8. In these algorithms, transProb[.] and obsProb[.] show transition probability matrix and observation probability vector for simple actions. Also, transition probability of moving from action i to action j is shown as transProb(i,j) and maximum of the transition probabilities from action i to all other actions is designated as max(transProb(i),Y), Y being the set of actions generating the maximum transition probability. Finally, maximum of the observation probabilities of the actions is indicated by max(obsProb,Y), Y being the action having the maximum observation probability.

Algorithm 5 Training phase of SCD

> **for** $i = 1 \rightarrow N$ **do**
> $vectorCount \leftarrow 0$
>   **for** $j = 1 \rightarrow K$ **do**
>   **for** $m = 1 \rightarrow S$ **do**
>   distance=([X,Y,Z]-[0,center,0])/radius
>   vectorCount++
>   Range[i,vectorCount]=distance
>   **end for**
>   **end for**
> **end for**

Algorithm 6 Prediction phase of SCD

```
actionFound ← 0
for i = 1 → N do
    distance=([X,Y,Z]-[0,center,0])/radius
    if distance ∉ Range[i] then
        SCD fails to classify the segment
    else
        actionFound ← 1
        Mark action as the action corresponding to i
        Terminate SCD
    end if
end for
if actionFound = 0 then
    SCD fails to classify this segment
    Call HMM scheme.
end if
```

Algorithm 7 Training phase of HMM

```
transProb[.] ← 0
obsProb[.] ← 0
for i = 1 → N do
    for t = 1 → C do
        for j = 1 → N do
            action=AC_i∪ AC_j
            (transAction_1,transAction_2)=nBayes(action)
            transProb(transAction_1,transAction_2)++
            obsProb(transAction_2)++
        end for
    end for
end for
```

IV. Results and Conclusion

This experimental study covers the design and implementation of a real time indoor human activity monitoring system by addressing the two phases of activity monitoring, namely data acquisition and classification. Two models are used in the experiments; single sensor-single classifier and multi sensors-multi classifiers models.

Table 2: Transition probabilities of simple actions

Action Name	Transition Probability
Walk-Walk	0.8
Walk-Sit	0.05
Walk-Stand	0.15
Sit-Walk	0.2
Sit-Sit	0.6
Sit-Stand	0.2
Stand-Walk	0.3
Stand-Sit	0.2
Stand-Stand	0.5

In single sensor-single classifier approach, Naive Bayes classifier is implemented for the data classification subsystem. Unique intervals of average posterior probability of the training data in 2's complement form are calculated with the normal distribution in the training phase. In the prediction phase, the real time sample is partitioned into chunks at the points where chunks show the simple actions and the chunk size shows the representative duration of classified action by means of calculating average posterior probability for each

Algorithm 8 Prediction phase of HMM

```
if lastAction ∉ defined then
    lastAction ← sit
end if
max(transProb(i),Y)
if card(Y) > 1 then
    max(obsProb,Y)
    Segment is classified as Y
else
    a∊ Y
    Segment is classified as a
end if
```

chunk. The tests are repeated several times. On average following success rates are achieved: walk 92%, sit 100%, lie 88% and stand 96% as tabulated in Table 3. Based on these simple actions, various numbers of tests are performed for the detection of ***walk-after-sit*** and ***sit-after-lie*** composite actions. The model classified the actions successfully when the transition signals that normally occur between actions in real-life samples are ignored.

Table 3: Detection success rates for simple actions (SSSC)

Action Name	Detection Success Rate
Walk	92%
Sit	100%
Lie	88%
Stand	96%

In multi sensor-multi classifier approach, a hybrid classifier is used. naive Bayes, HMM and SCD are used to detect transitions between activities in real time. In the training stage of Naive Bayes classifier, data are processed to calculate mean and standard deviation under the normal probability distribution to extract a posterior probability. In addition to this, every action and transition between the actions is observed to calculate transition probability and observation probability under joint probability distribution in the training of HMM. In the prediction phase, the posterior probability of the real time data is calculated. If it overlaps with any other action, the data are processed by SCD to find distance to center of circle. If the distance is mapped to any action, it is marked as detected action. Otherwise, HMM classifier finds the next action using transition and observation probabilities, obtained in training stage. The tests are repeated for several times and it has been observed that the model is fast and 100% accurate in detecting the transition signals. However, because of the real time processing delay a minor reduction in the detection of individual actions are observed.

Table 4: Detection success rates for simple actions(MSMC)

Action Name	Detection Success Rate
Walk	94%
Walk while hands in pocket	96%
Sit	94%
Stand	94%
Wheelchair driving	98%

Detection success rates achieved with MSMC model are 94% for walk, 96% for walking while hands in pocket, 94% for sit, 94% for stand, and 98% for wheelchair driving, as given in Table 4. The results reveal that the proposed filtering mechanism can successfully distinguish actions. As a future work, the model will be integrated into an abnormal activity detection scenario, such as fall and bump.

References

[1] Y. Lin, E. Becker, K. Park, Z. Le, and F. Makedon, "Decision making in assistive environments using multimodal observations," in *PETRA 09 Proceedings of the 2nd International Conference on PErvsive Technologies Related to Assistive Environments*. ACM, 2009, pp. 1–8.

[2] D. W. Marquardt, "An Algorithm for Least-Squares Estimation of Nonlinear Parameters," *SIAM Journal on Applied Mathematics*, vol. 11, no. 2, pp. 431–441, 1963.

[3] P. Cunningham and S. J. Delany, "k-Nearest Neighbour Classifiers," *Multiple Classifier Systems*, pp. 1–17 (UCD–CSI–2007–4), University College Dublin, 2007.

[4] J. Hopfield, "Artificial neural networks," *Circuits and Devices Magazine, IEEE*, vol. 4, no. 5, pp. 3 –10, sep 1988.

[5] S. R. Gunn, "Support Vector Machines for Classification and Regression," *The Analyst*, vol. 135, no. 2, pp. 230–267, Citeseer, 1998.

[6] Hung, Tahir, Farrell, McLoone, and McCarthy, "Wireless Sensor Networks for Activity Monitoring using Multi-sensor Multi-modal Node Architecture," *Proceedings of Signals and Systems Conference (ISSC 2009), IET Irish*, pp. 1–6, 2009.

[7] K. Römer, "Discovery of Frequent Distributed Event Patterns in Sensor Networks," in *European Conference on Wireless Sensor Networks EWSN*, vol. 67322, no. 5005. Springer-Verlag, 2008, pp. 106–124.

[8] L. Bao and S. S. Intille, "Activity Recognition from User-Annotated Acceleration Data," *Proceedings of Pervasive Computing, Second International Conference, PERVASIVE 2004*, vol. 3001, pp. 1–17, Springer, 2004.

[9] C. Zhu and W. Sheng, "Human Daily Activity Recognition in Robot-assisted Living Using Multi-sensor Fusion," *Proceedings of Robotics and Automation, 2009, ICRA '09, IEEE International Conference*, pp. 2154–2159, 2009.

[10] D. Lymberopoulos, A. Bamis, and A. Savvides, "Extracting spatiotemporal human activity patterns in assisted living using a home sensor network," *Proceedings of the 1st ACM international conference on PErvasive Technologies Related to Assistive Environments PETRA 08*, p. 1, 2008.

[11] S. Bosch, M. Marin-Perianu, R. Marin-Perianu, P. Havinga, and H. Hermens, "Keep on moving! activity monitoring and stimulation using wireless sensor networks," *Proceedings of EuroSSC'09, the 4th European conference on Smart sensing and context*, pp. 11–23, ACM, 2009.

[12] G. Uslu, O. Altun, and S. Baydere, "A Bayesian Approach for Indoor Human Activity Monitoring," *Proceedings of IEEE Int. Conference on Hybrid Intelligence Systems, HIS 2011*, pp. 324–327, 2011.

[13] X. Hong, C. Nugent, M. Mulvenna, S. McClean, B. Scotney, and S. Devlin, "Evidential fusion of sensor data for activity recognition in smart homes," *Pervasive and Mobile Computing*, vol. 5, no. 3, pp. 236 – 252, Elsevier Science Publishers B. V., 2009.

[14] C. Bettini, O. Brdiczka, K. Henricksen, J. Indulska, D. Nicklas, A. Ranganathan, and D. Riboni, "A survey of context modelling and reasoning techniques," *Pervasive Mob. Comput.*, vol. 6, no. 2, pp. 161–180, Elsevier Science Publishers B. V., Apr. 2010.

[15] J. Ye, S. Dobson, and S. McKeever, "Situation identification techniques in pervasive computing: A review," *Pervasive and Mobile Computing*, vol. 8, no. 1, pp. 36 – 66, Elsevier Science Publishers B. V., 2012.

[16] E. Sazonov, G. Fulk, J. Hill, Y. Schutz, and R. Browning, "Monitoring of posture allocations and activities by a shoe-based wearable sensor," *Biomedical Engineering, IEEE Transactions on*, vol. 58, no. 4, pp. 983 –990, april 2011.

[17] L. Chen, C. Nugent, and H. Wang, "A knowledge-driven approach to activity recognition in smart homes," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 24, no. 6, pp. 961 –974, june 2012.

[18] S.-Y. Chiang, Y.-C. Kan, Y.-C. Tu, and H.-C. Lin, "Activity recognition by fuzzy logic system in wireless sensor network for physical therapy," in *Intelligent Decision Technologies*, ser. Smart Innovation, Systems and Technologies, J. Watada, T. Watanabe, G. Phillips-Wren, R. J. Howlett, and L. C. Jain, Eds., Springer Berlin Heidelberg, 2012, vol. 16, pp. 191–200.

[19] J. A. Ward, P. Lukowicz, and H. W. Gellersen, "Performance metrics for activity recognition," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 1, pp. 6:1–6:23, Jan. 2011.

[20] A. Czabke, S. Marsch, and T. Lueth, "Accelerometer based real-time activity analysis on a microcontroller," in *Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2011 5th IEEE International Conference on*, may 2011, pp. 40 –46.

Author Biographies

Gamze Uslu received her BSc degree in computer engineering from Yeditepe University, Istanbul, Turkey in 2011. She is a research assistant and currently pursuing the MSc degree in computer engineering at Yeditepe University. Her research interests include activity monitoring, machine learning and body sensor networks.

Halil Ibrahim Dursunoglu received the B.Sc. degree in Computer Engineering from Yeditepe University, Istanbul, Turkey, in 2012. He works as manager in an international company. His research interests include wireless sensor networks and activity monitoring.

Ozgur Altun received the B.Sc. degree in Computer Engineering from Kocaeli University, Turkey, in 2010. He is currently pursuing the M.Sc. degree in Satellite Communication and Remote Sensing at Istanbul Technical University. He works as a research assistant at Yeditepe University. His research interests include wireless sensor networks, activity monitoring, wireless and satellite communication.

Sebnem Baydere is a full Professor and the Chair of the Department of Computer Engineering at Yeditepe University, Istanbul. She received her BSc and MSc degrees in Computer Engineering from Middle East Technical University (METU), Ankara, in 1984 and 1987 respectively. She received her PhD degree in Computer Science from University College London (UCL), UK, in 1990. Her current research interests are in the area of wireless sensor and ad hoc networks, wireless multimedia networks, network interoperability, context aware systems and distributed systems. Dr Baydere coordinated several national and international research projects and published papers in the area of wireless sensor networks and distributed systems. She served as technical committee member for several conferences and peer-reviewed journal articles on Sensor Networks. She also served numerous times as evaluator for European and National research projects.