ISLAMIC UNIVERSITY OF TECHNOLOGY

Human Activity Recognition using Dynamic Time Warping

By
G. M. Mukit Hasan (104419)
Kazi MD Iftekhar (104412)
Systems & Software Lab

Supervisor
Hasan Mahmud
Assistant Professor
Department of Computer Science and Engineering

A thesis submitted to the Department of CSE in partial fulfilment of the requirements for the degree of B.Sc. Engineering in CSE
Academic Year: 2013-2014

A Subsidiary Organ of the Organization of Islamic Co-operation
Gazipur, Bangladesh
October 2014

Certificate of Research

This is to certify that the work presented in this thesis is the outcome of analysis and investigation carried out by the candidate under the supervision of Mr. Hasan Mahmud in the department of Computer Science and Engineering (CSE), IUT, Gazipur, Bangladesh. It is also declared that neither of this thesis nor any part of this thesis has been submitted anywhere else for any degree or diploma. Information derived from the published and unpublished work of others has been acknowledged in the text and a list of references is given.

Signature of the Supervisor

____

Hasan Mahmud
Assistant Professor
Department of Computer Science and Engineering (CSE)

<u>Authors</u>

___________________   ___________________

G. M. Mukit Hasan
ID: 104419

Kazi MD Iftekhar
ID: 104424

Abstract

Human activity recognition is now a well-known field of Human Computer Interaction (HCI) because its capability of providing personalized support using different applications. The recognition of human activities has become a task of high interest within the field, especially for medical, military, and security applications. Its applications include surveillance systems, patient monitoring systems, and a variety of systems that involve interactions between persons and electronic devices such as human-computer interfaces. In our work we will try to recognize human activity using wearable sensors. The works done before had the problem of using multimodal system with satisfactory results. Computation of the inputs to recognize activities are not simple. So, we designed a multi-modal system that will take accelerometer, gyroscope and ultrasonic sensor's data as input and use optimized Dynamic Time Warping algorithm to classify data to recognize activity with satisfactory success rate.

Keywords

DTW, Accelerometer, Gyroscope, Ultrasonic, Activity, Recognition

Table of Contents

Chapter 1

Introduction

Activity recognition aims to recognize the actions and goals of one or more agents from a series of observations on the agents' actions and the environmental conditions. Since the 1980s, this research field has captured the attention of several computer science communities due to its strength in providing personalized support for many different applications and its connection to many different fields of study such as medicine, human-computer interaction, or sociology.Its applications include surveillance systems, patient monitoring systems, and a variety of systems that involve interactions between persons and electronic devices such as human-computer interfaces. Automated surveillance systems in public places like airports and subway stations require detection of abnormal and suspicious activities as opposed to normal activities. For instance, an airport surveillance system must be able to automatically recognize suspicious activities like ‘a person leaving a bag’ or ‘a person placing his/her bag in a trash bin’. Recognition of human activities also enables the real-time monitoring of patients, children, and elderly persons.

We can use vision based system or sensor based system to recognize activity. But as far we researched, there are some limitations on vision based sensor system which is why we used sensor based system. The limitations are –

- **Poor Resolution:** Low resolution and low quality also affect visibility.
- **Frame Rate:** Frame rate is also a considerable factor in case of vision based system.
- **Drastic Illumination Conditions:** Different illumination conditions also affects visibility.
- **Changing Weather Conditions:** Visibility and detection depends on the weather condition in case of uncontrolled environment.
- **Occlusion:** occlusion is the effect of one object in a 3-D space blocking another object from view.

On the other hand, Sensor based system is portable and efficient to use. The determination of physical activities in everyday life suffers from suitable sensors, their position and algorithms. Distributed multi-sensor systems provide a high accuracy but not very handy and are inconvenient. Single sensor systems can achieve a sufficient recognition rate only in controlled environment and using a fixed sensor location and orientation. The concern of everyday usage is not to have an additional sensing device but an integration of this functionality into a standard device which should be easy to handle and accurate detect the everyday activities.

Our problem statement is:

"To design a multi-modal system that will take accelerometer, gyroscope etc. sensor's data as input and use optimized Dynamic Time Warping algorithm to classify data to recognize activity with high efficiency and success rate."

We designed a multimodal system merging the sensors (Accelerometer, Gyroscope & Ultrasonic) together. The sensors reading data is taken as input and minimal path is calculated using dynamic time warping algorithm. Then we calculated the compound warp path normalizing five sensor reading data into one to classify different activities.

Our thesis report is organized in the following way:

- Introduction
- Background Study
- Proposed methodology
- Literature review/Related Work
- Conclusion and future work

The innovative aspects are:

1. Sensors are fused togather to get better accuracy,
2. Our designed system is multimodal,
3. System is sensor based activity recognition,
4. We used optimum dtw,
5. Our designed system shows the fact in initial results.

Chapter 2

Background Study

Now before we go to explain our proposed system there are some basic terminologies reader should be aware of.

Activity recognition aims to recognize the actions and goals of one or more agents from a series of observations on the agents' actions and the environmental conditions. Since the 1980s, this research field has captured the attention of several computer science communities due to its strength in providing personalized support for many different applications and its connection to many different fields of study such as medicine, human-computer interaction, or sociology.

A time series is a collection of observations made sequentially in time. For example - Stock prices, Volume of sales over time, Daily temperature readings and ECG data.

Time warping:

Given that two sequences x_1, x_2,...,x_n and y_1,y_2,...,y_m. We want to align two sequence base on a common Time-axis. Now if we have two time series, we can find out the similarity between the two time series using the following formula.

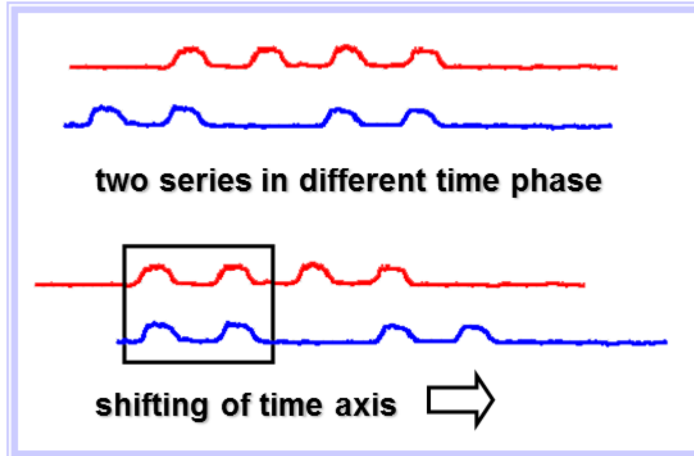
Formula of Dynamic Time Warping:

$$\gamma(i,j) = d(qi,cj) + \min\{\gamma(i-1,j-1), \gamma(i-1,j), \gamma(i,j-1)\}$$

Dynamic Time Warping:

Dynamic time warping (DTW) is a technique that finds the optimal alignment between two time series if one time series may be “warped” non-linearly by stretching or shrinking it along its time axis.

This warping between two time series can then be used to find corresponding regions between the two time series or to determine the similarity between the two time series.



Wrap path: Wrap path is found by finding out the distance of 2 similar points of the time serieses .In this case, a non-linear (elastic) alignment produces more intuitive similarity measure, allowing similar shapes to match even if they are out of phase in the Time axis.

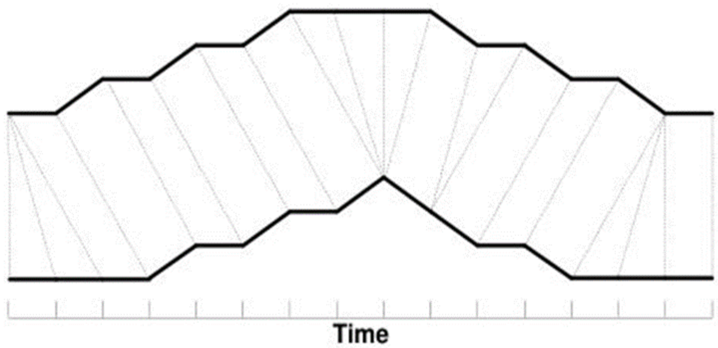


Figure 1. A warping between two time series.

2.1 Why DTW Algorithm:

We can use feature extraction method to recognize human activities instead of DTW algorithm. In case of feature extraction method we have to use different types of classifiers at the same time such as k-nearest neighbor, quadratic classifier, support vector machine etc. Most likely to implement some probability functions to achieve good result like HMM ,Conditional Random Field etc., whereas using DTW use for these probability models get minimum. But using DTW we can just compare time series distance to classify human activities which more accurate. Now it's also time efficient because algorithm's time complexity can be improved to O (N) from O (N^2).

Reference Data: Labeled sensor data that will match with test data to obtain

Test Data: Test data is obtained using sensor reading from activities to match with reference data to recognize activities.

Cost Matrix: Cost matrix is a matrix used to compare to time series and specify warping path.

2.2 Mechanism of DTW:

The dynamic time warping problem is stated as follows: Given two time series X, and Y, of lengths |X| and |Y|

$X=X_1+X_2+X_3+....+X_{|X|}$

$Y=Y_1+Y_2+Y_3+...+Y_{|Y|}$

From there we can construct a path:

$W=W_1+W_2+W_3+....+W_k$

Where max $(|X|,|Y|)\leq k<|X|+|Y|$

Where K is the length of the warp path and the k_{th} element of the warp path is W (i,j).

Where i is an index from time series X, and j is an index from time series Y.

To use this problem formulation to map it to find the warp path there are some rules to be followed.They are :

The warp path must start at the beginning of each time series at $w_1 = (1, 1)$ and finish at the end of both time series at W_k= (|X|, |Y|). This ensures that every index of both time series is used in the warp path.

There is also a constraint on the warp path that forces i and j to be monotonically increasing in the warp path, which is why the lines representing the warp path in Figure 1 do not overlap.

Every index of each time series must be used.

To find the optimum warp path[18], there is a formula:

The optimal warp path is the warp path is the minimum-distance warp path, where the distance of a warp path W is

$$Dist(W) = \sum_{k=1}^{k=K} Dist(w_{ki}, w_{kj})$$

Dist(W) is the distance (typically Euclidean distance) of warp path W, and Dist(w_{ki} , w_{kj}) is the distance between the two data point indexes (one from X and one from Y) in the k_{th} element of the warp path.

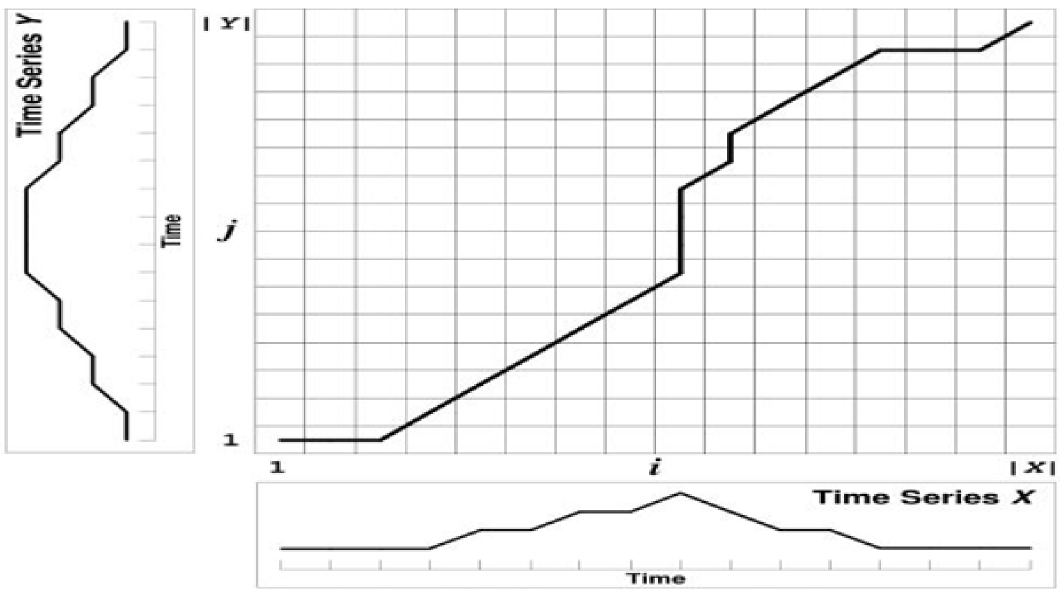


Figure 2: Dynamic Time Warping

2.3 Optimum warp path calculation:

The 2 time series are[18]

In this case, the points are:

(0,0),(1,0),(2,0),(3,1),(4,1) ...

Here, the points are:

(0,0),(1,0), (2,1),(3,1), (4,2)...

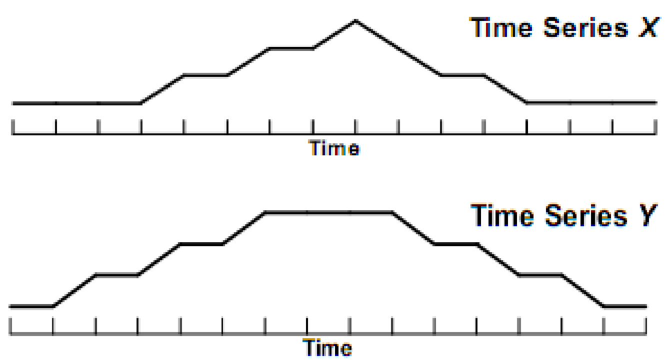


Figure 3: Two separate time series

If the warp path passes through a cell D (i, j) in the cost matrix, it means that the i_{th} point in time series X is warped to the j_{th} point in time series Y.

So, from the time series we first map (0, 0) and (0, 0) points of the both series to (1, 1) point of the graph having warp path.

Then, for the second point we take (1,0), (1,0) for the both input and previously taken input was (1,1) so, the minimum of these three is(1,0) ,which is added with (1,1).So we get the new value (2,1).

For the next value we have (2, 0) and (2, 1) and the previous point of the warp path is (2, 1).So the minimum of these three is (2, 0). By adding (1, 1) we get the third point (3, 1).

For the next value we have (3, 0) and (3, 1) and the previous point of the warp path is (3, 1).So the minimum of these three is (3, 0). By adding (1, 1) we get the third point (4, 1).

In this way we can create the warp path from (1, 1) to (|X|, |Y|), where |X|, |Y| are the length of those time series.

2.4 Cost matrix:

After the entire matrix[18] is filled, a warp path must be found from D (1, 1) to D (|X|, |Y|). The warp path is actually calculated in reverse order starting at D (|X|, |Y|).

A greedy search is performed that evaluates cells to the left, down, and diagonally to the bottom-left.

Whichever of these three adjacent cells has the smallest value is added to the beginning of the warp path found so far, and the search continues from that cell.

The search stops when D (1, 1) is reached.

2.5 Speeding up DTW(Minimum Comparison):

The methods used make DTW faster[18] fall into three categories:

i. **Constraints –** Limit the number of cells that are evaluated in the cost matrix.
ii. **Data Abstraction –** Perform DTW on a reduced representation of the data.
iii. **Indexing –** Use lower bounding functions to reduce the number of times DTW must be run during time series classification or clustering.

Chapter 3

Related Works

DTW running time is costly so to get better performance some compromisation are needed.

Salvador et. al. [18] tried to reduce the complexity of DTW algorithm from $O(N^2)$ to O(N) in both time and space. They called the optimized algorithm as Fast DTW.

- Fast DTW is a recursive implementation.
- They did it using three key operations:
  - **Coarsening–** Shrink a time series into a smaller time series that represents the same curve as accurately as possible with fewer data points.
  - **Projection–** Find a minimum-distance warp path at a lower resolution, and use that warp path as an initial guess for a higher resolution's minimum-distance warp path.
  - **Refinement–** Refine the warp path projected from a lower resolution through local adjustments of the warp path.

Error of Fast DTW is at most 19.2% .To reduce the complexity the error rate has increased.Casale et al. used accelerometer as sensor and feature selection technique to recognize activity with success 92%

In this case accelerometer value was calculated as:

$$A_m = \sqrt{A_x}^2 + {A_y}^2 + {A_z}^2$$

Features were extracted from window of 52 samples 1 second of accelerometer data and 50% overlapping between the windows.

Still, in order to complete the set of features some were added like: mean value, standard deviation, skewness, kurtosis, correlation between each pairwise of accelerometer axis.

Chapter 4

Proposed Approach

Our propose approach is given below step by step:

4.1 System Understanding:

We have followed supervised learning as we try to recognize the activities. So, there would be 2 kinds of data:

I. Trained data or referenced data
II. Testing data

Referenced data would be stored in the database. We need proper equipment to acquire the testing data along with controlled environment. After some preprocessing like filtering, averaging etc. we can get the usable data which can be used as input for the DTW algorithm. With the help of the algorithm we can get the minimum warp path distance.

In this case, we used 3 sensor data: 3 axis accelerometer, gyroscope and ultrasonic sensor data. Then we used some mechanism to convert 5 values (3 of them for 3-axis accelerometer) to 1 compound value that ranges from 0 to 1. Then with the help of the normalized value we can detect activity with better accuracy.

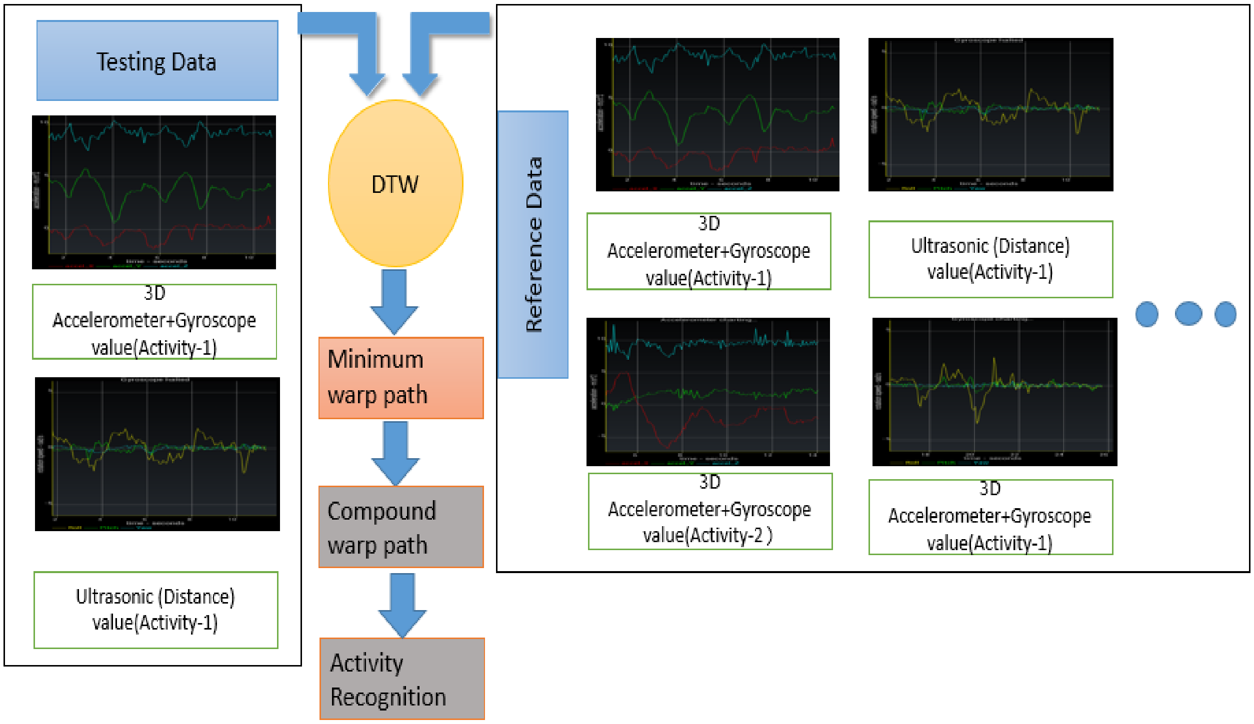


Figure 4: System Understanding

4.2 Architecture:

Basic idea of the system architecture is something like this:

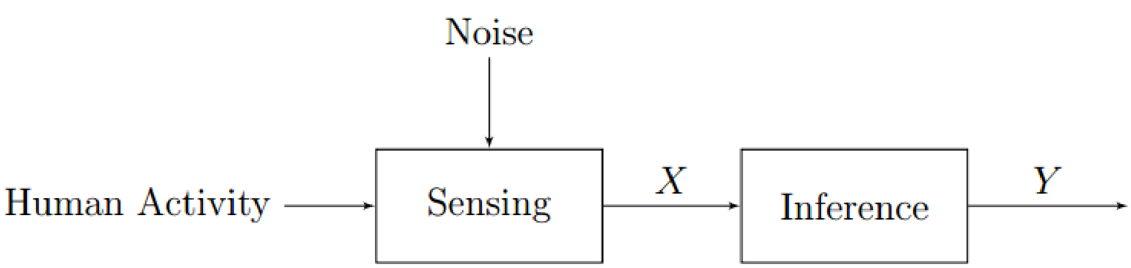


Figure 5: Basic Architecture

As we can see from the figure, we first take input data from sensor reading from user's activity. Those value contains noise, so 'X' is the observed sensor data and 'Y' is the recognized activity.

This figure can be described greatly with another figure:

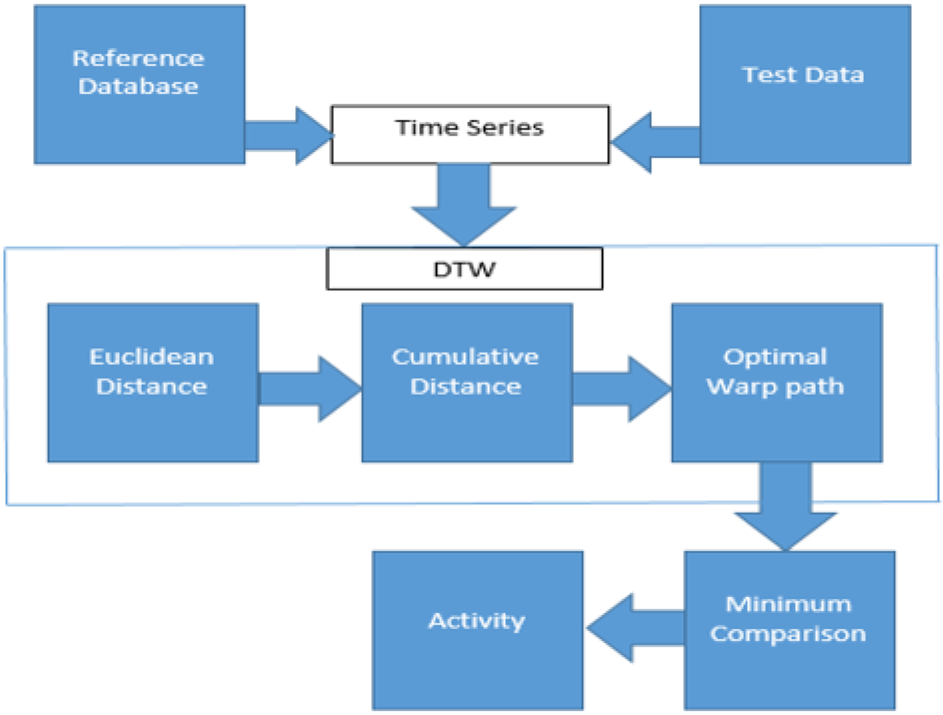


Figure 6: Architecture

Here we can see that there are mainly 4 steps to our proposed architecture. They are:

I. Data collection and preprocessing.
II. Optimum warp path calculation.
III. Compound calculation.
IV. Activity calculation.

4.3 Data Collection & Preprocessing:

- Two types of data need to be collected
  - Reference Data
  - Test Data
- Data collection process:

- Position of the sensors are fixed: attached with user's thigh.
- Tri-axial acceleration from the accelerometer (total acceleration) and the estimated body acceleration.
- Tri-axial Angular velocity from the gyroscope.
- Distance from thigh to ground from ultrasonic sensor.
- Total 4 activities are considered: sitting, walking, running, laying.
- For each activity 5 instances are taken for each user. First 3 instances are used as training data & last 2 instances are used for testing.
- These time domain signals were captured at a constant rate of 50 Hz.
- Then they were filtered using a median filter and a 3rd order low pass Butterworth filter with a corner frequency of 20 Hz to remove noise.[20]
- Similarly, the acceleration signal was then separated into body and gravity acceleration signals using another low pass Butterworth filter with a corner frequency of 0.3 Hz. [20]

4.4 Warp Path Calculation:

Warp path is calculated by taking 2 time series of same sensor data for each axis and try to plot them in the graph. The DTW algorithm will return the warp path distance. Then we stored them in tables like this:

	Sitting	Walking	Running	Laying
Sitting	211.188, 9.054, 7.296, 30.069, 49.851	2187.405, 39.054, 49.131, 102.201, 55.009	228.399, 19.537, 23.232, 106.178, 99.923	211.456, 88.255, 185.063, 51.811, 146.197
Walking	2187.405, 39.054, 49.131, 102.201, 55.009	1676.760, 7.975, 5.375, 24.862, 32.977	1968.016, 30.470, 26.039, 35.803, 61.936	1998.297, 49.317, 145.557, 73.153, 134.386
Running	228.399, 19.537, 23.232, 106.178, 99.923	1968.016, 30.470, 26.039, 35.803, 61.936	36.939, 9.747, 21.152, 35.508, 42.128	44.287, 79.468, 165.532, 88.650, 132.37

Laying	211.456, 88.255, 185.063, 51.811, 146.197	1998.297, 49.317, 145.557, 73.153, 135.386	44.287, 79.468, 165.532, 88.650, 132.37	23.678, 41.454, 27.509, 28.919, 125.007

Table 1 : User 1 instance 1 verses user 1 instance 2

	Sitting	Walking	Running	Laying
Sitting	14.831, 9.738, 58.980, 157.029, 128.049	316.284, 316.241, 302.053, 252.477, 137.343	40.0,13.762, 93.088, 159.299, 148.090	117.913, 12.185, 451.836, 2432.415, 1301.193
Walking	316.284, 316.241, 302.053, 252.477, 137.343	0.0, 0.0, 68.464, 227.563, 133.036	316.284, 315.599, 301.711, 175.806, 163.204	337.220, 315.796, 522.433, 2599.258, 1388.779
Running	40.0, 13.762, 93.088, 159.299, 148.090	316.284, 315.599, 301.711, 175.806, 163.204	10.577, 1.131, 69.383, 131.345, 101.738	117.913, 1.633, 453.277, 2615.699, 1181.687
Laying	117.913, 12.185, 451.836, 2432.415, 1301.193	337.220, 315.796, 522.433, 2599.258, 1388.779	117.913, 1.633, 453.277, 2615.699, 1181.687	32.994, 0.603, 279.332, 403.515, 1006.020

Table 2: User 2 instance 1 verses user 2 instance 2

	Sitting	Walking	Running	Laying
Sitting	14.831, 9.738, 58.980, 157.029, 128.049	316.284, 316.241, 302.053, 252.477, 137.343	40.0,13.762, 93.088, 159.299, 148.090	117.913, 12.185, 451.836, 2432.415, 1301.193
Walking	316.284, 316.241, 302.053, 252.477, 137.343	0.0, 0.0, 68.464, 227.563, 133.036	316.284, 315.599, 301.711, 175.806, 163.204	337.220, 315.796, 522.433, 2599.258, 1388.779
Running	40.0, 13.762, 93.088, 159.299, 148.090	316.284, 315.599, 301.711, 175.806, 163.204	10.577, 1.131, 69.383, 131.345, 101.738	117.913, 1.633, 453.277, 2615.699, 1181.687
Laying	117.913, 12.185, 451.836, 2432.415, 1301.193	337.220, 315.796, 522.433, 2599.258, 1388.779	117.913, 1.633, 453.277, 2615.699, 1181.687	32.994, 0.603, 279.332, 403.515, 1006.020

Table 3 : User 3 instance 1 verses user 3 instance 2

	Sitting	Walking	Running	Laying
Sitting	118.283, 123.977, 87.065, 211.098, 138.291	402.765, 332.311, 550.262, 2639.990, 1435.762	218.282, 127.273, 488.759, 2657.139, 1228.595	146.016, 125.981, 131.607, 218.689, 132.578
Walking	402.765, 332.311, 550.262, 2639.990, 1435.762	239.941, 26.711, 53.773, 78.959, 143.074	2003.456, 90.329, 478.956, 2601.827, 1202.119	2105.910, 89.151, 85.664, 171.576, 193.321
Running	218.282, 127.273, 488.759, 2657.139, 1228.595	2003.456, 90.329, 478.956, 2601.827, 1202.119	56.222, 18.428, 84.018, 109.061, 104.817	156.321, 117.335, 110.725, 182.930, 181.198
Laying	146.016, 125.981, 131.607, 218.689, 132.578	2105.910, 89.151, 85.664, 171.576, 193.321	156.321, 117.335, 110.725, 182.930, 181.198	123.775, 39.921, 77.609, 117.408, 113.453

Table 4 : User 1 instance 1 verses user 2 instance 2

	Sitting	Walking	Running	Laying
Sitting	137.991, 14.036, 16.378, 14.239, 76.001	203.752, 134.952, 139.244, 452.101, 133.743	235.919, 19.029, 27.979, 33.116, 188.018	141.618, 134.953, 238.675, 115.202, 176.676
Walking	203.752, 134.952, 139.244, 452.101, 133.743	199.382, 18.634, 13.548, 41.181, 124.679	1961.663, 20.061, 21.772, 70.844, 188.251	1997.205, 98.633, 96.578, 80.099, 160.121
Running	235.919, 19.029, 27.979, 33.116, 188.018	1961.663, 20.061, 21.772, 70.844, 188.251	18.262, 16.108, 6.263, 30.224, 156.924	61.329, 126.989, 121.671, 99.863, 172.725
Laying	141.618, 134.953, 238.675, 115.202, 176.676	1997.205, 98.633, 96.578, 80.099, 160.121	61.329, 126.989, 121.671, 99.863, 172.725	32.252, 49.317, 93.729, 71.497, 131.835

Table 5 : User 1 instance 1 verses user 3 instance 2

	Sitting	Walking	Running	Laying
Sitting	146.267, 265.471, 248.884, 156.516, 133.585	379.099, 382.729, 637.262, 2742.596, 1437.718	246.266, 268,191, 550.329, 2670.960, 1234.678	168.149, 267.114, 275.458, 219.823, 181.932
Walking	379.099, 382.729, 637.262, 2742.596, 1437.718	31.759, 4.834, 5.885, 8.224, 154.574	53.863, 14.117, 456.346, 2173.857, 1226.145	84.361, 14.468, 14.146, 426.916, 262.957

Running	246.266, 268,191, 550.329, 2670.960, 1234.678	53.863, 14.117, 456.346, 2173.857, 1226.145	50.749, 9.214, 103.494, 134.041, 126.408	161.539, 108.033, 106.184, 134.888, 137.853
Laying	168.149, 267.114, 275.458, 219.823, 181.932	84.361, 14.468, 14.146, 426.916, 262.957	161.539, 108.033, 106.184, 134.888, 137.853	25.567, 14.467, 14.097, 118.935, 59.823

Table 6 : User 2 instance 1 verses user 3 instance 2

So, for each activity comparison 5 values become the output and there are 4*4=16 values. So, in total 80 values are there. But in this way comparison becomes complex. To simplify the term 'compound warp path' comes.

4.5 Compound Warp Path Calculation:

To compute the compound warp path calculation, first we need to do the stated steps below:

- **Normalization:** Normalization is done following the equation below:

$$\text{Normalized\_value} = \frac{\text{warp\_path\_value} - \text{min\_ warp\_path\_value}}{\text{max\_warp\_path\_value}}$$

- **RMS(Root Mean Square):** RMS is done like equation below:

$$rms\_value = \sqrt{x^2 + y^2 + z^2 + a^2 + d^2}$$

  Here, x=accelerometer x-axis value,
  y=accelerometer y-axis value,
  z=accelerometer z-axis value,
  a=gyroscope angle and
  d=ultrasonic distance value

- **Final Step:** To keep the value between 0 to 1 we need to divide the RMS value by 5:

$$\text{Compound warp path} = \frac{rms\_value}{5}$$

So the tables become like this:

User 1 instance 1 verses user 1 instance 2:

	Sitting	Walking	Running	Laying
Sitting	0.0201	0.2012	0.0258	0.0549
Walking	0.2012	0.1534	0.1807	0.1855
Running	0.0258	0.1807	0.0067	0.0466
Laying	0.0549	0.1855	0.0466	0.0248

User 2 instance 1 verses user 2 instance 2:

	Sitting	Walking	Running	Laying
Sitting	0.0177	0.1716	0.0216	0.2422
Walking	0.1716	0.0212	0.1711	0.3089
Running	0.0216	0.1711	0.0139	0.2429
Laying	0.2422	0.3089	0.2429	0.1284

User 3 instance 1 verses user 3 instance 2:

	Sitting	Walking	Running	Laying
Sitting	0.0432	0.1576	0.1024	0.1529
Walking	0.1576	0.0341	0.3403	0.3084
Running	0.3064	0.3403	0.0217	0.1542
Laying	0.2920	0.2326	0.2521	0.2536

User 1 instance 1 verses user 2 instance 2:

	Sitting	Walking	Running	Laying
Sitting	0.0691	0.3197	0.2586	0.0711
Walking	0.3197	0.0303	0.3086	0.1996
Running	0.2586	0.3086	0.0177	0.0676
Laying	0.0711	0.1996	0.0676	0.0279

User 1 instance 1 verses user 3 instance 2:

	Sitting	Walking	Running	Laying
Sitting	0.0156	0.0818	0.0309	0.0777
Walking	0.0818	0.0238	0.1808	0.1907
Running	0.0309	0.1808	0.0179	0.0701
Laying	0.0777	0.1907	0.0701	0.0303

User 2 instance 1 verses user 3 instance 2:

	Sitting	Walking	Running	Laying
Sitting	0.1423	0.3404	0.2888	0.1449
Walking	0.3404	0.0158	0.2223	0.0434
Running	0.2888	0.2223	0.0189	0.0613
Laying	0.1449	0.0434	0.0613	0.0118

4.6 Activity Recognition:

From the tables, diagonal values should be the minimum values with respect to their columns and rows, because they have the same activity to comparison with. But in general values of instance 1 is used as referenced data and instance 2 as testing samples.

So, the values which can produce minimum distance values, their respective reference activities are the recognized as tester's own activity at that instance.

Chapter 5

Experimental Results

Result analysis is done base on the accuracy of correctly detecting the activity. Like the previous work, if we consider the warp path for each sensor data, we can find more errors than from the compound warp path data.

Success from the previous case is around 90.00% ,which is calculated using single sensor data.

For our case, to increase the success rate and still use the simplified representation of the calculated result we used 3 sensors' data along with total five values simplifid into one and then we calculated the success rate.

For this we used, the 10 testing data(2 for each user) and we compared each with 15 traning data(3 training data for each user).so we get 15*10=150 tables for result calculations.

Our collection of data table contains 4*4 blocks for each activity comparison and each block contains five sensor values. We reduced the dimensionality of the input by optmizing to get 1 value for each block. We named the newly constructed table as ‘Compound Warp Path’ table . Here we calculated the values of the diagonal ones, because they need to be minimum (because they are the comparison of the same activities).So, we need to check 4 values for each table.

In the appendix A portion we have included data of one person’s test instances warp path and appendix B portion we have included data one person’s test instances compound warp path data.

So, to check success we need to check the values of 4*150=600 values and from there only 35 values are found wrong. So our success rate in this case, ((600-35)/600)*100 = 94.16%

Chapter 6

Conclusion and Future Works

During the past decade, there has been an exceptional development of microelectronics and computer systems, enabling sensors and mobile devices with unprecedented characteristics.Their high computational power, small size, and low costallow people to interact with the devices as part of theirdaily living. That was the genesis of Ubiquitous Sensing,an active research area with the mainpurpose of extractingknowledge from the data acquired by pervasive sensors.So, time for Human Activity Recognition(HAR) has become very much possible.

Much of the work has been done on this topic.But, satisfactory result has yet to come.We tried to simplyfy this matter by reducing the complexity without compromising accuracy.

In our work, after combining 5 sensor values into a single representation, the success rate increased drastically.

If we get further chance we may implement this process for a larger data set.

Application scope for implementing this process can help a lot in elderly supervision, future gaming, security measures etc.

In uncontrolled environment, noise can affect the reading a lot, which results in failure to recognize the activity.

In near future Samsung and other mobile companies may attempt to include ultrasonic sensor, then smart phone will be sufficient enough to detect the activities using multimodal architectures like our proposed system.

Bibliography

[1] L. Bao and S. S. Intille. Activity recognition from user-annotated acceleration data. In Proceedings of The International Conference on Pervasive Computing, pages 1-17. Springer, 2004.

[2] T. Maekawa, Y. Yanagisawa, Y. Kishino, K. Ishiguro, K. Kamei, Y. Sakurai, and T. Okadome. Object-based activity recognition with heterogeneous sensors on wrist. In Proceedings of The International Conference on Pervasive Computing, pages 246-264. Springer, 2010.

[3] M. Muehlbauer, G. Bahle, and P. Lukowicz. What can an arm holster worn smartphone do for activity recognition? In Proceedings of The International Symposium on Wearable Computers, ISWC ’11, pages 79–82, 2011.

[4] K.-H. Chang, M. Y. Chen, and J. Canny. Tracking free-weight exercises. In Proceedings of The International Conference on Ubiquitous Computing, UbiComp ’07, 2007.

[5] Z. Yan, D. Chakraborty, A. Misra, H. Jeung, and K. Aberer. SAMMPLE: Detecting Semantic indoor activities in practical settings using locomotive signatures. In Proceedings of The 2012 16th Annual International Symposium on Wearable Computers, ISWC ’12, pages 37-40, Washington, DC, USA, 2012.

[6] M. Stikic, D. Larlus, S. Ebert, and B. Schiele. Weakly supervised recognition of daily life activities with wearable sensors. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011.

[7] M. Mahdaviani and T. Choudhury. Fast and scalable training of semi-supervised CRFs with application to activity recognition. In Advances in Neural Information Processing Systems, pages 977–984, Cambridge, MA, 2007. MIT Press.

[8] U. Blanke and B. Schiele. Remember and transfer what you have learned-recognizing composite activities based on activity spotting. In International Symposium on Wearable Computers, 2010.

[9] T. Huynh, M. Fritz, and B. Schiele. Discovery of activity patterns using topic models. In Proceedings of The International Conference on Ubiquitous Computing, UbiComp ’08, pages 10–19, 2008.

[10] D. Minnen, T. Starner, I. Essa, and C. Isbell. Discovering characteristic actions from on-body sensor data. In In Proceedings of The IEEE International Symposium on Wearable Computing, ISWC ’06, pages 11–18, 2006.

[11] D. Wyatt, M. Philipose, and T. Choudhury. Unsupervised activity recognition using automatically mined common sense. In Proceedings of The 20th national conference On Artificial Intelligence - Volume 1, AAAI’05, pages 21–27. AAAI Press, 2005.

[12] Z. Zhang. Microsoft Kinect sensor and its e↵ect. IEEE Multimedia, 19(2):4–10, 2012. [102] J. Zheng and L. M. Ni. An unsupervised framework for sensing individual and cluster behavior patterns from human mobile data. In Proceedings of the 2012 ACM Conference on Ubiquitous Computing, UbiComp ’12, pages 153–162, New York, NY, USA, 2012. ACM.

[13] D. H. Hu, X.-X. Zhang, J. Yin, V. W. Zheng, and Q. Yang. Abnormal activity recognition based on HDP-HMM models. In Proceedings of the 21st International Jont Conference on Artifical Intelligence, IJCAI’09, pages 1715–1720, San Francisco, CA,USA, 2009.

[16] Sasiwan Paiyarom, Poj Tungamchit, Rachaporn Keinprasit, Prakasith Kayasit. Activity Monitoring System using Dynamic Time Warping for the Elderly and Disabled people.Computer, Control and communication, 2009, 2nd International Conference, pages 1-4, Karachi.

[17] Siddharth S. Rautaray, Anupam Agrawal.Vision based hand gesture recognition for human computer interaction: a survey.Springer Science+Business Media Dordrecht 2012

[18] Stan Salvador and Philip Chan. FastDTW: Toward Accurate Dynamic Time Warping in Linear Time and Space dept. of Computer Sciences. Florida Institute of Technology. Melbourne, FL 32901.

[19] Heng-Tze Cheng. Learning and Recognizing Te Hierarchical and Sequential Structure of Human Activities. Teses and Dissertations at Research Showcase @ CMU , December, 2013

[20] Source : Jorge L. Reyes-ortiz, Davide anguita, Alessandro ghio, Luca oneto. - Non linear complex systems laboratory DITEN - Universitã degli studi di genova, Genoa I-16145, Italy. Activityrecognition '@' smartlab.Ws www.Smartlab.Ws

Appendix A

U1 I4 vs U1 I1

	Sitting	Walking	Running	Laying
Sitting	218.489, 0.371, 0.022, 2.958, 290.995	731.896, 852.053, 848.003, 738.094, 245.497	1047.445, 884.269, 590.407, 852.222, 1120.578	290.455, 88.099, 899.830, 850.789, 1173.899
Walking	574.434, 522.180, 523.028, 486.763, 444.483,	1641.183, 1943.699, 1949.448, 845.933, 548.571	2179.595, 2145.054, 1342.163, 2177.347, 1557.512	29.093, 2106.178, 2073.819, 2127.524, 1628.397
Running	598.995, 522.526, 522.532, 545.098, 479.806	1649.649, 1943.998, 2.015, 1949.123, 70.844, 188.251	60.749, 109.214, 483.494, 634.041, 206.408	44.287. 79.468, 165.532, 88.650, 132.37
Laying	294.455, 89.059, 879.750, 680.978, 1145.867	83.916,106.468, 263.81,975.84, 176.101	156.321, 117.335, 110.725, 182.930, 181.198	125.567, 14.467, 14.976, 118.935, 259.823

U1 I4 vs U1 I2

	Sitting	Walking	Running	Laying
Sitting	75.905, 0.231, 1.080, 0.582, 6.749	290.455, 520.404, 521.660, 444.544, 94.934	337.567, 521.035, 522.447, 556,961, 189.271	290.455, 521.091, 516.978, 556.945, 168.232
Walking	396.061, 522.370, 522.435, 483.558, 102.025	29.093, 2.578, 2.228, 49.718, 0.061	20.577, 3.350, 0.302, 85.200, 97.740	29.093, 3.246, 7.619, 74.809, 75.887
Running	416.876, 522.716, 521.938, 541.891, 144.039,	50.507, 20.273, 542.699, 175.806, 163.204	56.222, 118.428, 484.018, 609.061, 184.817	1640.92,49.317, 51.963,63.54, 223.349
Laying	29.455, 521.091, 516.978, 556.945, 168.232	105.910, 89.151, 85.664, 171.576, 193.321	44.287. 79.468, 165.532, 88.650, 132.37	123.775, 39.921, 97.609, 117.408, 213.453

U1 I4 vs U1 I3

	Sitting	**Walking**	**Running**	**Laying**
Sitting	25.819, 0.150, 0.0, 0.0, 1.579	306.966, 522.535, 523.273, 468.490, 94.751	342.669, 522.988, 522.671, 551.397, 154.531	306.966, 522.971, 520.494, 545.226, 141.704
Walking	288.503, 522.435, 523.045, 484.042, 96.405,	4.849, 0.072, 0.093, 8.831, 0.016	24.149, 0.423, 0.352, 69.304, 61.627	4.849, 0.405, 2.638, 62.94, 48.406
Running	309.317, 522.781, 522.547, 542.377, 138.399,	30.207, 0.394, 0.739, 369.939, 229.747	10.577, 41.131, 469.383, 531.345, 061.738	44.287. 79.468, 165.532, 88.650, 132.37
Laying	306.966, 522.971, 520.494, 545.226, 141.704	84.361, 14.468, 14.146, 426.916, 262.957	156.321, 117.335, 110.725, 182.930, 181.198	123.775, 39.921, 97.609, 117.408, 213.453

U1 I4 vs U2 I1

	Sitting	**Walking**	**Running**	**Laying**
Sitting	73.257, 1.661, 0.483, 1.521, 1.947	324.151, 522.517, 522.725, 500.228, 94.686	335.009, 522.710, 523.289, 532.907, 118.727	324.151, 522.712, 521.636, 530.537, 113.354
Walking	396.805, 521.037, 523.547, 482.659, 92.722,	8.965, 0.049, 0.328, 16.669, 0.034	20.123, 0.148, 0.612, 50.218, 24.483	8.965, 0.151, 1.445, 47.784, 19.127
Running	417.619, 521.383, 523.051, 540.994, 134.717,	12.444, 0.404, 0.163, 369.939, 229.747	60.749, 109.214, 483.494, 634.041, 206.408	1640.92,49.317, 51.963,63.54, 223.349
Laying	324.151, 522.712, 521.636, 530.537, 113.354	83.916,106.468, 263.81,975.84, 176.101	161.539, 108.033, 106.184, 134.888, 137.853	125.567, 14.467, 14.976, 118.935, 259.823

U1 I4 vs U2 I2

	Sitting	**Walking**	**Running**	**Laying**
Sitting	41.916, 3.032, 1.176, 1.863, 5.849	326.859, 522.513, 523.475, 505.275, 94.677	334.637, 522.657, 522.441, 529.992, 112.587	326.859, 522.669, 521.803, 528.226, 108.934
Walking	351.825, 519.242, 522.265, 482.056, 89.647,	11.745, 0.053, 0.421, 21.847, 0.043	19.741, 0.093, 0.620, 47.226, 18.342	11.745, 0.096, 1.275, 45.411, 14.590
Running	372.639, 519.588, 521.769, 540.391, 131.641,	9.662, 0.409, 0.0481, 175.806, 163.204	10.577, 41.131, 469.383, 531.345, 061.738	164.92,49.317, 51.963,63.54, 223.349
Laying	326.859, 522.669, 521.803, 528.226, 108.934	215.910, 89.151, 85.664, 171.576, 193.321	44.287. 79.468, 165.532, 88.650, 132.37	32.252, 49.317, 103.729, 71.497, 131.835

U1 I4 vs U2 I3

	Sitting	**Walking**	**Running**	**Laying**
Sitting	135.833, 3.418, 2.399, 2.529, 3.146	329.052, 522.516, 522.571, 509.354, 94.672	334.347, 522.621, 522.439, 527.661, 108.045	329.051, 522.623, 521.942, 526.383, 105.407
Walking	239.362, 518.911, 520.687, 481.197, 81.597	13.991, 0.051, 0.486, 26.028, 0.049	19.447, 0.057, 0.621, 44.836, 13.686	13.992, 0.059, 1.132, 43.522, 10.974
Running	241.812, 519.257, 520.191, 539.532, 133.591,	7.421, 0.405, 0.009, 70.844, 188.251	56.222, 118.428, 484.018, 609.061, 184.817	44.287. 79.468, 165.532, 88.650, 132.37
Laying	329.051, 522.623, 521.942, 526.383, 105.407	73.296,24.658, 352.371,104.403, 133.84	156.321, 117.335, 110.725, 182.930, 181.198	123.775, 39.921, 97.609, 117.408, 213.453

U1 I4 vs U3 I1

	Sitting	**Walking**	**Running**	**Laying**
Sitting	8.837, 5.457, 5.579, 3.641, 1.808	334.595, 523.414, 523.291, 518.727, 94.667	333.728, 522.571, 522.498, 523.309, 97.846	334.104, 522.679, 522.283, 522.285, 97.546
Walking	307.397, 528.394, 529.003, 485.729, 93.995	19.281, 0.031, 0.609, 35.616, 0.055	18.811, 0.015, 0.563, 39.803, 3.251	19.169, 0.018, 0.784, 39.329, 2.930
Running	328.212, 528.741, 528.506, 544.064, 135.989,	2.284, 0.346, 0.952, 175.806, 163.204	15.577, 41.131, 69.383, 21.345, 65.789	1640.92,49.317, 51.963,63.54, 223.349
Laying	334.104, 522.679, 522.283, 522.285, 97.546	84.361, 14.468, 14.146, 426.916, 262.957	44.287. 79.468, 165.532, 88.650, 132.37	125.567, 14.467, 14.976, 118.935, 259.823

U1 I4 vs U3 I2

	Sitting	**Walking**	**Running**	**Laying**
Sitting	14.716, 5.752, 5.579, 7.041, 2.056	335.564, 522.599, 522.431, 521.424, 94.667	333.569, 522.569, 522.573, 520.991, 95.001	335.564, 522.579, 522.389, 521.159, 95.374
Walking	301.232, 528.707, 529.003, 491.792, 96.632	20.651, 0.039, 0.629, 38.371, 0.054	18.648, 0.019, 0.486, 38.014, 1.422	20.651, 0.019, 0.675, 38.178, 1.172
Running	322.047, 529.053, 528.506, 539.927, 138.626,	1.297, 0.321, 0.119, 369.939, 229.747	10.577, 41.131, 469.383, 531.345, 061.738	44.287. 79.468, 165.532, 88.650, 132.37

Laying	335.564, 522.579, 522.389, 521.159, 95.374	83.916,106.468, 263.81,975.84, 176.101	161.539, 108.033, 106.184, 134.888, 137.853	32.252, 49.317, 103.729, 71.497, 131.835

U1 I4 vs U3 I3

	Sitting	**Walking**	**Running**	**Laying**
Sitting	58.728, 5.566, 4.796, 5.579, 1.099	337.000, 522.622, 522.409, 524.078, 94.677	333.416, 522.569, 522.721, 519.590, 93.287	337.000, 522.579, 522.493, 520.918, 93.287
Walking	369.394, 528.510, 528.087, 490.002, 95.801,	22.117, 0.057, 0.652, 41.079, 0.054	18.493, 0.025, 0.348, 36.583, 1.522	22.117, 0.019, 0.568, 37.354, 1.522
Running	390.208, 528.855, 527.590, 548.337, 137.796,	1.316, 0.298, 0.142, 175.806, 163.204	56.222, 118.428, 484.018, 609.061, 184.817	1640.92,49.317, 51.963,63.54, 223.349
Laying	337.000, 522.579, 522.493, 520.918, 93.287	205.910, 89.151, 85.664, 171.576, 193.321	156.321, 117.335, 110.725, 182.930, 181.198	123.775, 39.921, 97.609, 117.408, 213.453

U1 I4 vs U4 I1

	Sitting	**Walking**	**Running**	**Laying**
Sitting	103.857, 3.254, 5.033, 3.716, 1.940	342.233, 522.738, 522.355, 532.676, 94.667	332.953, 522.576, 2826.856, 515.260, 86.694	341.665, 522.585, 522.816, 517.588, 86.921
Walking	426.951, 526.119, 528.504, 488.015, 93.656	27.068, 0.175, 0.707, 49.843, 0.043	18.020, 0.019, 2411.556, 32.167, 8.132	26.872, 0.030, 0.255, 33.958, 7.913
Running	447.765, 526.464, 528.007, 546.351, 135.650,	5.585, 0.181, 0197, 70.844, 188.251	60.749, 109.214, 483.494, 634.041, 206.408	44.287. 79.468, 165.532, 88.650, 132.37
Laying	341.665, 522.585, 522.816, 517.588, 86.921	73.296,24.658, 352.371,104.403, 133.84	44.287. 79.468, 165.532, 88.650, 132.37	125.567, 14.467, 14.976, 118.935, 259.823

U1 I4 vs U4 I2

	Sitting	**Walking**	**Running**	**Laying**
Sitting	253.382, 4.646, 3.371, 4.908, 547.503, 1.446	343.555, 522.802, 522.347, 537.104, 94.686	332.790, 522.591, 501.522, 513.625, 83.726	343.455, 522.604, 522.947, 515.560, 84.918
Walking	90.828, 527.611, 526.647, 489.368, 93.416	28.795, 0.239, 0.715, 53.771, 0.035	17.853, 0.033, 39.500, 30.502, 11.157	28.795, 0.041, 0.201, 35.467, 17.532

Running	111.340, 527.957, 526.151, 550.542, 135.411,	7.382, 0.138, 0.205, 369.939, 229.747	10.577, 41.131, 469.383, 531.345, 61.738	156.321, 117.335, 110.725, 182.930, 181.198
Laying	343.455, 522.604, 522.947, 515.560, 84.918	84.361, 14.468, 14.146, 426.916, 262.957	161.539, 108.033, 106.184, 134.888, 137.853	32.252, 49.317, 103.729, 71.497, 131.835

U1 I4 vs U4 I3

	Sitting	**Walking**	**Running**	**Laying**
Sitting	176.588, 6.405, 4.097, 7.516, 1.639	345.781, 522.889, 522.344, 540.230, 94.699	332.612, 522.613, 422.793, 511.785, 80.651	345.781, 522.631, 523.089, 514.205, 79.716
Walking	496.791, 529.473, 527.361, 492.208, 93.326	31.059, 0.328, 0.719, 57.529, 0.026	17.672, 0.048, 0.215, 28.628, 14.452	31.059, 0.066, 0.205, 37.087, 16.107
Running	517.606, 529.819, 526.864, 545.527, 135.321,	9.646, 0.125, 0.209, 369.939, 229.747	58.577, 41.131, 469.383, 231.345, 11.738	44.287. 79.468, 165.532, 88.650, 132.37
Laying	345.781, 522.631, 523.089, 514.205, 79.716	83.916,106.468, 263.81,975.84, 176.101	156.321, 117.335, 110.725, 182.930, 181.198	123.775, 39.921, 97.609, 117.408, 213.453

U1 I4 vs U5 I1

	Sitting	**Walking**	**Running**	**Laying**
Sitting	86.817, 1.683, 1.863, 2.910, 0.329	357.378, 523.436, 522.395, 562.616, 102.104	331,866, 522.765, 522.473, 504.232, 84.986	357.378, 523.657, 523.794, 508.377, 86.445
Walking	405.728, 524.353, 535.030, 487.192, 94.434	42.828, 0.883, 0.668, 79.667, 0.524	16.914, 0.202, 0.586, 20.382, 27.872	42.828, 0.257, 0.754, 25.163, 31.169
Running	426.543, 524.698, 524.533, 544.364, 136.438,	21.415, 0.524, 0.158, 175.806, 163.204	56.222, 118.428, 484.018, 609.061, 184.817	1640.92,49.317, 51.963,63.54, 223.349
Laying	357.378, 523.657, 523.794, 508.377, 86.445	125.910, 89.151, 85.664, 171.576, 193.321	161.539, 108.033, 106.184, 134.888, 137.853	32.252, 49.317, 103.729, 71.497, 131.835

U1 I4 vs U5 I2

	Sitting	Walking	Running	Laying
Sitting	88.801, 1.741, 0.755, 1.863, 1.466	365.735, 522.102, 522.373, 576.879, 394.887	331.451, 522.874, 522, 447, 498.697, 90.023	365.735, 522.942, 524.201, 505.144, 94.160
Walking	411.144, 524.422, 523.233, 486.029, 93.416	51.288, 1.693, 0.688, 94.679, 405.182	16.493, 0.312, 0.613, 15.429, 35.108	51.287, 0.380, 1.166, 21.883, 40.387
Running	431.758, 524.667, 522.739, 544.365, 135.411,	29.874, 1.457, 0.178, 70.844, 188.251	60.749, 109.214, 483.494, 634.041, 206.408	44.287. 79.468, 165.532, 88.650, 132.37
Laying	365.735, 522.942, 524.201, 505.144, 94.160	84.361, 14.468, 14.146, 426.916, 262.957	44.287. 79.468, 165.532, 88.650, 132.37	32.252, 49.317, 103.729, 71.497, 131.835

U1 I4 vs U5 I3

	Sitting	Walking	Running	Laying
Sitting	140.143, 1.071, 0.0, 0.0, 1.504	380.723, 1004.917, 520.537, 606.708, 556.819	330.747, 522.993, 522.440, 491.036, 98.631	380.723, 522.882, 524.794, 500.313, 103.286
Walking	462.531, 523.456, 523.043, 484.042, 94.538	66.433, 1042.234, 2.524, 124.799, 475.181	15.782, 0.434, 0.621, 12.143, 44.608	66.432, 0.383, 1.768, 16.995, 56.660
Running	483.345, 524.102, 522.547, 542.377, 136.532,	45.019, 1042.348, 2.916, 175.806, 163.204	56.222, 118.428, 484.018, 609.061, 184.817	1640.92,49.317, 51.963,63.54, 223.349
Laying	380.723, 522.882, 524.794, 500.313, 103.286	83.916,106.468, 263.81,975.84, 176.101	161.539, 108.033, 106.184, 134.888, 137.853	123.775, 39.921, 97.609, 117.408, 213.453

Appendix B

U1 I4 vs U1 I1

	Sitting	Walking	Running	Laying
Sitting	0.0381	0.1492	0.2076	0.1803
Walking	0.1096	0.0246	0.4083	0.3881
Running	0.1142	0.2878	0.0195	0.0226
Laying	0.1728	0.0781	0.0310	0.0308

U1 I4 vs U1 I2

	Sitting	Walking	Running	Laying
Sitting	0.0077	0.0843	0.1466	0.0888
Walking	0.0891	0.0058	0.0101	0.0077
Running	0.0923	0.0685	0.0676	0.1521
Laying	0.0848	0.0280	0.0226	0.0183

U1 I4 vs U1 I3

	Sitting	Walking	Running	Laying
Sitting	0.0047	0.0857	0.0903	0.0886
Walking	0.0856	0.0042	0.0062	0.0045
Running	0.0886	0.0636	0.0589	0.0622
Laying	0.0886	0.0428	0.031	0.0283

U1 I4 vs U2 I1

	Sitting	Walking	Running	Laying
Sitting	0.0078	0.0871	0.0888	0.0882
Walking	0.0890	0.0043	0.0038	0.0035
Running	0.0921	0.0632	0.0695	0.1521
Laying	0.0882	0.0781	0.0264	0.0318

U1 I4 vs U2 I2

	Sitting	Walking	Running	Laying
Sitting	0.0052	0.0874	0.0886	0.0882
Walking	0.0872	0.0044	0.0139	0.0537
Running	0.0902	0.0605	0.0589	0.0924
Laying	0.0882	0.0329	0.0226	0.0173

U1 I4 vs U2 I3

	Sitting	Walking	Running	Laying
Sitting	0.0130	0.0875	0.0884	0.3155
Walking	0.0837	0.0045	0.0049	0.0289
Running	0.0863	0.0701	0.0676	0.0926
Laying	0.0882	0.0371	0.0310	0.0283

U1 I4 vs U3 I1

	Sitting	Walking	Running	Laying
Sitting	0.0041	0.0881	0.0881	0.0881
Walking	0.0869	0.0049	0.0187	0.0148
Running	0.0899	0.0204	0.0087	0.1521
Laying	0.0881	0.0428	0.0226	0.0118

U1 I4 vs U3 I2

	Sitting	Walking	Running	Laying
Sitting	0.0042	0.0881	0.0880	0.0881
Walking	0.0869	0.0051	0.0074	0.095
Running	0.0896	0.0362	0.0589	0.0226
Laying	0.0881	0.0781	0.0264	0.0173

U1 I4 vs U3 I3

	Sitting	Walking	Running	Laying
Sitting	0.0067	0.0882	0.0880	0.0881
Walking	0.0890	0.0012	0.0048	0.0049
Running	0.0921	0.0804	0.0676	0.1521
Laying	0.0881	0.0323	0.0310	0.0283

U1 I4 vs U4 I1

	Sitting	Walking	Running	Laying
Sitting	0.0103	0.0887	0.2806	0.0881
Walking	0.0909	0.0057	0.2312	0.0244
Running	0.0940	0.0272	0.0195	0.0226
Laying	0.0881	0.0471	0.0526	0.0318

U1 I4 vs U4 I2

	Sitting	Walking	Running	Laying
Sitting	0.0687	0.0889	0.0865	0.0881
Walking	0.0825	0.0021	0.0049	0.0039

Running	0.0854	0.0362	0.0289	0.0310
Laying	0.0881	0.0428	0.0264	0.0173

U1 I4 vs U4 I3

	Sitting	**Walking**	**Running**	**Laying**
Sitting	0.0166	0.0890	0.0825	0.0881
Walking	0.0941	0.0062	0.0092	0.0098
Running	0.0971	0.0362	0.0180	0.0226
Laying	0.0881	0.0781	0.0310	0.0283

U1 I4 vs U5 I1

	Sitting	**Walking**	**Running**	**Laying**
Sitting	0.0090	0.0903	0.1275	0.0885
Walking	0.0903	0.0077	0.0019	0.0041
Running	0.0929	0.0205	0.0676	0.1521
Laying	0.0885	0.0287	0.0264	0.0173

U1 I4 vs U5 I2

	Sitting	**Walking**	**Running**	**Laying**
Sitting	0.0091	0.1013	0.1036	0.0887
Walking	0.0898	0.0475	0.0817	0.0749
Running	0.0929	0.0202	0.0695	0.0226
Laying	0.0887	0.0428	0.0226	0.0173

U1 I4 vs U5 I3

	Sitting	**Walking**	**Running**	**Laying**
Sitting	0.0134	0.1435	0.0871	0.0892
Walking	0.0918	0.1212	0.0021	0.0068
Running	0.0950	0.1093	0.0676	0.1521
Laying	0.0892	0.0781	0.0264	0.0283

Thank You