International Journal of Knowledge Based Computer Systems
8 (1 & 2) June & December 2020, 01-10
http://www.publishingindia.com/ijkbcs/

Efficient Classification Techniques of Human Activities from Smartphone Sensor Data using Machine Learning Algorithms

Diponkor Bala[1*] and G. M. Waliullah[2]
[1]Department of Computer Science & Engineering, Islamic University, Kushtia, Bangladesh.
Email: diponkor.b@gmail.com
[2]Department of Computer Science & Engineering, Islamic University, Kushtia, Bangladesh.
*Corresponding Author

Abstract: Increasing use of accelerometer and protractor sensors in recent years has created a field of study for the definition of human activities. This issue is tried to be solved by using machine learning methods. For this, it is solved by extracting different properties from the obtained signals, obtaining the characteristics specific to the activity and classifying these properties. In this study, the time and frequency domain properties of 4 different human activities were extracted, then a pre-treatment step was applied in accordance with the obtained feature set, and then the size was reduced with PCA and Fisher 'LDA methods. The k-NN classifier and perceptron classifiers were designed for the obtained feature set and the classification process was performed. In this study, the classification success of these methods using different parameters has been examined and the results are shown.

Keywords: Feature extraction, Fisher's LDA, Gradient descent method, Human activity identification, Kessler's reconstruction, k-NN classifier, PCA.

I. Introduction

The development of sensor technologies and the reduction of costs and the use of accelerometer and protractor sensors with wearable devices have opened the door to important researches in the perception of human activities. In addition, these sensors are integrated with mobile devices and their sensitivity is sufficient, allowing researchers to easily collect and analyze this data. With these developments, it leads the development of mobile assistants at the mobile application point and the development of health assistant programs that allow more accurate activity tracking [1-10].

Machine learning algorithms are actively used to classify the obtained data. Different machine learning algorithms have been proposed in the literature. These algorithms have different success rates at the point of classifying activities. These achievements vary according to the data and application collected by each research group conducting the study [11-13].

II. Dataset

In this study, a public "MobiAct V2.0" dataset [14] is used. This data set includes 2 different scenarios. Scenario-1 was built on Fall Detection. Scenario-2 was created on the recognition of Human Activities. In this study, scenario-2 data were studied. This scenario has acceleration and angle data for 11 different activities. Data were collected over 60 candidates collected for each activity. Also, different numbers of trials and lengths of data are different for each activity [15] [16].

This data set was collected using a Samsung Galaxy S3 mobile phone. The LSM330DLC sensor used in the device provides 3-axis acceleration and angle information. Sampling frequency for the acceleration sensor is 87 Hz. Sampling frequency for the protractor is 200 Hz. In this study, the 11-class problem is considered as a 4-class problem. In addition, the classification process was made using only the acceleration information. Information on the classes is presented in Table I.

Table I: Data Information

Code	Activity	Trials	Duration	Description
STD	Standing	1	5 m	Standing
WAL	Walking	1	5 m	Normal Walking
JOG	Jogging	3	30 s	Jogging
JUM	Jumping	3	30 s	Cont. Jumping

III. Methods

For the 4-class problem dealt with, 2 different feature matrices were obtained by using the information about the time and frequency space from the data of each class. Matrix sizes obtained for time and frequency domains are 33 dimensions. So different feature extraction methods were used for both spaces. Then, outlier extraction was performed on these feature matrices. For the Outlier extraction process, the $\pm 3\sigma$ limit condition was applied for each feature and the feature samples outside this limit were removed from the data set. Later, each feature taking values at different intervals on the data set recovered from the outlier may affect the algorithms used in classification. To prevent this, dynamic range has been moved between [0 1]. Then, "Pearson Relationship Coefficients" were found to analyze the relationship between properties. These coefficients are between [-1 + 1] and allow analysis of which properties are related to each other. If the relationship between 2 features is close to +1, these two features show an increase feature together. Likewise, if this coefficient gives a value close to -1, then these two properties are inversely related to each other. A feature set was created using this feature and the effect of this choice on the classification was examined. Apart from this selection, different size reduction methods were applied to the feature sets obtained at the beginning. The methods used here are PCA (Principle Component Analysis) and the other method is Fisher's Linear Discriminant Analysis methods. Here, different number of eigenvalues were selected for the PCA method and classification success in different sizes was observed. Fisher's Method, on the other hand, reduced a c-class problem to (c-1) dimension and reduced it to 3 dimensions for this problem and the classification success was examined for the designed classifiers. The applied method and analysis methods are given below as a block diagram.

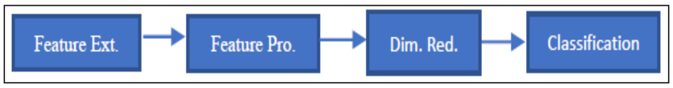
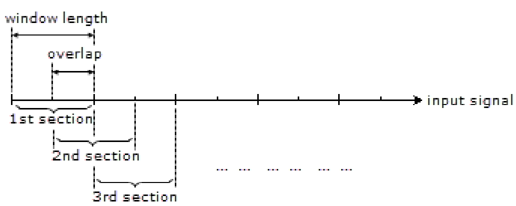


Fig. 1: Method Flow

IV. Feature Extraction

A. Time Domain

The feature extraction process in time space for 4 classes of different lengths given in Table I was performed as follows. Acceleration data were collected so that 87 samples were recorded in 1 second. Here, the studies in the literature have been examined and feature extraction from the data is performed by using windows with sizes corresponding to periods varying between 2-3 seconds. In this study, a rectangular window with 218 point length corresponding to 2.5 seconds was used and 50% overlap corresponding to 109 samples was used. In other words, as shown in the image below, different properties are derived from the 2.5 second signals obtained from each section.



9 different feature extraction procedures were applied for each axis in time space. Here, a total of 33 dimensional time-space feature matrix was obtained.

a) *Mean:* The average value [17-21] for each axis of the 2.5 second windows obtained for each class was calculated as follows. Here N = 218 and x [n] is the signal for each axis n, expresses the value in the index.

$$\mu_{x,y,z} = \frac{1}{N} * \sum_{n=1}^{N} (\mathrm{x[n]}) \tag{1}$$

b) *Standard Deviation:* Standard deviation value [17-21] for each axis of 2.5 second windows obtained for each class has been calculated as follows. Here N = 218 and x[n] is the signal for each axis n, expresses the value in the index.

$$\sigma_{x,y,z} = \sqrt{\frac{1}{N-1} \sum_{n=1}^{N} |x[n] - \mu_{x,y,z}|^2} \tag{2}$$

c) *Median:* The median value [17-21] for each axis of the 2.5 second windows obtained for each class was calculated.

d) *Max-Min Value:* The max and min values [17-21] for each axis of the 2.5 second windows obtained for each class were calculated as follows. Standard deviation: To change the default, adjust the template as follows:

$$\max_i = \arg\max\{x_i\,[n]\} \quad i = x, y, z\ axis \tag{3}$$

$$\min_i = \arg\min\{x_i\,[n]\} \quad i = x, y, z\ axis \tag{4}$$

e) *Zero Shear Rate:* Zero shear rate for each axis of 2.5 second windows obtained for each class was calculated.

f) *R.M.S. Value***:** The rms (root mean square) value [17-21] for each axis of 2.5 second windows obtained for each class has been calculated as follows. Here N = 218 and x [n] is the signal for each axis n, expresses the value in the index.

$$\mathrm{x}_{RMS_{x,y,z}}[n] = \sqrt{\frac{1}{N} \sum_{n=1}^{N} |x[n]|^2} \tag{5}$$

g) *Skewness (3rd Moment):* 3rd Moment value [17-21] for each axis of 2.5 second windows obtained for each class was calculated as follows. Here N = 218 and x [n] is the signal for each axis n, expresses the value in the index.

$$skewness_i = \frac{1}{N-1} * \frac{\sum_{n=1}^{N} (x_i[n] - \mu_i)^3}{\sigma_i{}^3} \qquad i = x, y, z\ axis \tag{6}$$

Skewness information gives information about the symmetry of the obtained distribution with respect to the normal distribution. Normal distribution is considered to be perfectly symmetrical and the skewness information is 0.

h) *Kurtosis (4th Moment):* The 4th Moment value [17-21] for each axis of the 2.5 second windows obtained for each class has been calculated as follows. Here N = 218 and x [n] is the signal for each axis n, expresses the value in the index.

$$kurtosis_i = \frac{1}{N-1} * \frac{\sum_{n=1}^{N}(x_i[n] - \mu_i)^4}{\sigma_i{}^4} \quad i = x, y, z\ axis \tag{7}$$

This feature gives information about the flatness of the distribution. The normal distribution kurtosis value is 3 and generally the kurtosis value of the distribution is compared with the normal distribution.

i) *AR Model:* It calculates the 2nd order AR model coefficient values [22] for each axis of 2.5 second windows obtained for each class by solving the yule-walker equation as follows. Here N = 218 and x [n] is the signal for each axis n, expresses the value in the index.

$$\begin{pmatrix} r_0 & \cdots & r_{P-1} \\ \vdots & \ddots & \vdots \\ r_{P-1} & \cdots & r_0 \end{pmatrix} \begin{pmatrix} a_1 \\ \vdots \\ a_P \end{pmatrix} = -\begin{pmatrix} r_1 \\ \vdots \\ r_P \end{pmatrix}$$

$$R * a = -r$$

$$a = -R^{-1} * r \tag{8}$$

For p = 2, coefficients were calculated and used as properties.

The time domain feature extraction process is complete. Then, feature extraction in the frequency domain was performed.

B. Frequency Domain

For the frequency domain feature [23] extraction, it is first divided into 10 equal parts using a rectangular window with a total length of 30 seconds for STD and WAL classes with a data length of 5 minutes, in order to equalize the sample numbers of features from the data collected for each class, and the resulting hamming window Welch PSD estimation method was applied with 75% overlap on the 2.5 second segments. Likewise, for the JOG and JUM classes with a total length of 30 seconds, 10 equal parts of 10 seconds in length were obtained using a rectangular window. Then, by using a hamming window, the welch PSD estimation method was applied to the 2.5 second segments with 75% overlap. Then, the value of 8 peaks of the obtained estimate was used as a feature. In addition, spectral min and average values were used as properties for each axis.

a) *Welch Power Spectral Density Estimation:* With the Welch method, the periodogram of each block of the signal divided into overlapping blocks is calculated and the average of these PSDs [23] is calculated. The variance of the Welch method is less than the periodogram.

$$S_{welch} = \frac{1}{M}\sum_{i=1}^{M}\frac{1}{P} * \frac{1}{N} * \left|\sum_{n=0}^{N-1} x_r[n] * w[n] * e^{-i\omega n}\right|^2 \tag{9}$$

where

$$P = \frac{1}{M}\sum_{n=0}^{N-1}|w(t)|^2$$

Below is the welch psd estimation result for each class.

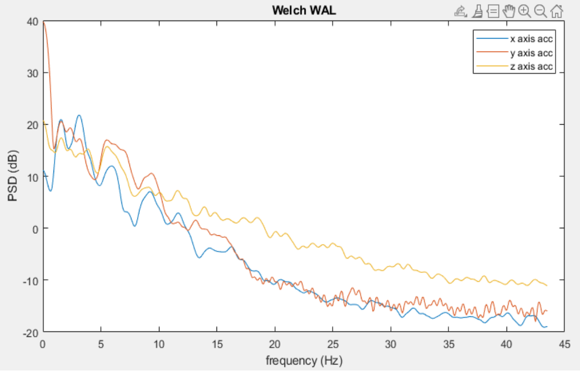


Fig. 2: WAL Subject-1 30 sec nfft = 4096

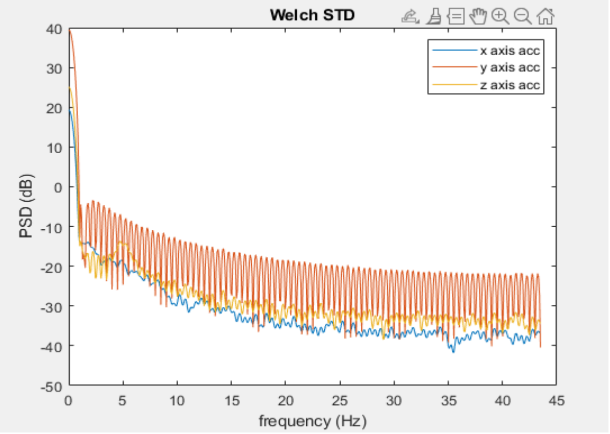


Fig. 3: STD Subject-1 30 sec nfft = 4096

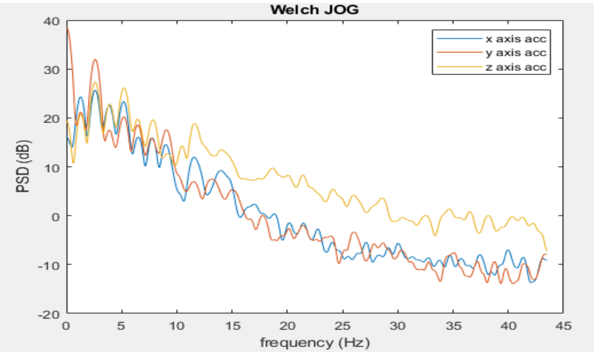


Fig. 4: JOG Subject-1 10 sec nfft: 1024

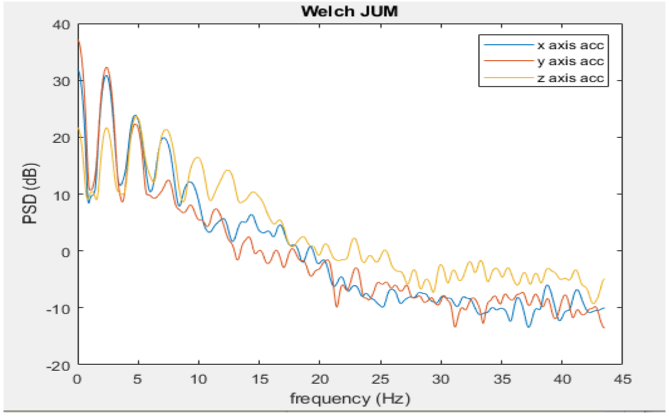


Fig. 5: JUM Subject-1 10 sec nfft = 1024

After extracting the features, the feature processing step was applied separately in time and frequency domains and feature matrices were obtained.

V. Feature Processing

First, outlier detection was performed on the obtained feature matrices. By checking whether this process is within ± 3σ for each feature, outlier extraction was performed. Then, the max-min normalization [24-26] process was carried out due to the distribution of each feature in a different dynamic range.

$$\bar{x}_{i,n} = \frac{x_{i,n} - min\{x_{1:S,n}\}}{max\{x_{1:S,n}\} - min\{x_{1:S,n}\}} \qquad (10)$$

i = 1, 2, 3,…, S (number of samples)

n = 1, 2, 3,…, D (number of features)

These operations were applied for the time and frequency domain feature matrices respectively. The data sets obtained were shuffled. Then, 70% of the total data for the classification process was allocated so that 30% of the training was tested, and the total number of feature samples obtained for both spaces are given below.

Table II: Time Space Feature Matrix Point Number

Data Set	Total Number of Samples	STD Number	WAL Number	JOG Number	JUM Number
RawFeature	36538	15775	11996	4434	4333
cleanFeature	30183	13629	9113	3696	3745

Table III: Time Space Training Matrix Point Number

Data Set	Training Sample Number	Training STD Number	Training WAL Number	Training JOG Number	Training JUM Number
RawFeature	25576	11078	8380	3089	3029
cleanFeature	21128	9595	6327	2623	2583

Table IV: Time Space Test Matrix Point Number

Data Set	Number of Test Samples	Test-STD Number	Test-WAL Number	Test-JOG Number	Test-JUM Number
RawFeature	10962	4697	3616	1345	1304
cleanFeature	9055	4034	2786	1073	1162

Table V: Frequency Space Feature Matrix Point Number

Data Set	Total Number of Samples	STD Number	WAL Number	JOG Number	JUM Number
RawFeature	2168	623	469	543	533
cleanFeature	1919	539	412	482	486

Table VI: Frequency Space Training Matrix Point Number

Data Set	Training Sample Number	Training STD Number	Training WAL Number	Training JOG Number	Training JUM Number
RawFeature	1517	387	297	330	329
cleanFeature	1343	370	292	334	347

Table VII: Frequency Domain Test Matrix Point Number

Data Set	Number of Test Samples	Test-STD Number	Test-WAL Number	Test-JOG Number	Test-JUM Number
RawFeature	651	236	172	213	204
cleanFeature	576	169	120	148	139

VI. Dimension Reduction

The feature matrices obtained are 33-dimensional and this dimension is quite high in terms of computational cost to perform the classification process. For this reason, 2 different dimension reduction methods [27-29] have been implemented. One of these methods is PCA and the other is Fisher's Linear Discriminant Analysis method.

A. PCA (Principle Component Analysis)

The covariance matrix of the property matrix obtained in the time and frequency domain is calculated. Then the eigenvalue and eigenvectors of the obtained covariance matrix are calculated. Eigenvalues are listed in descending order. Then the eigenvectors corresponding to the largest eigenvalue are arranged in the same order.

$$\Sigma = E * D * E^T \tag{11}$$

$$E = \begin{pmatrix} e_{1,1} & \cdots & e_{1,d} \\ \vdots & \ddots & \vdots \\ e_{d,1} & \cdots & e_{d,d} \end{pmatrix}, \qquad d = number\ of\ features$$

$$D = \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_d \end{bmatrix}, \qquad d = number\ of\ features$$

D: is the eigenvalue matrix, in descending order.

E: is the eigenvector matrix. He is a column eigenvector.

$$A = \begin{pmatrix} e_{1,1} & \cdots & e_{1,k} \\ \vdots & \ddots & \vdots \\ e_{d,1} & \cdots & e_{d,k} \end{pmatrix} \tag{12}$$

Where, k = desired size and k <d must be

$$\underline{y} = A^T * \left(\underline{x} - \underline{\mu}\right) \tag{13}$$

The y matrix obtained at the end of the process is the feature matrix of reduced size. Here, the value of k is chosen not to exceed d, and the A matrix that makes size reduction is obtained [31] [32]. This matrix is derived from training data and applied directly to the test data. Test data are projected on the same plane. After this stage, the classification process can be carried out.

For the time-space training and test data, PCA was applied for k = 3. Visually given below.

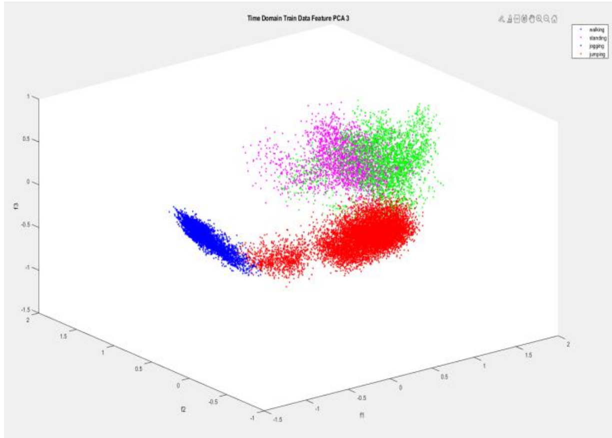


Fig. 6: PCA k = 3 Training Dataset

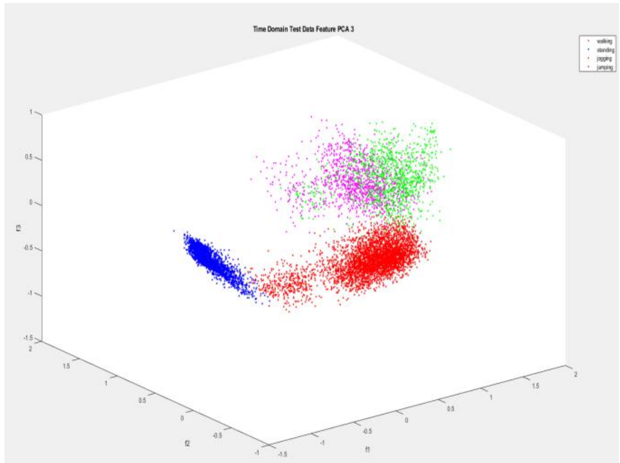


Fig. 7: PCA k = 3 Test Datasets

B. Fisher's LDA

Fisher's LDA is a method [33] [34] that reduces the data set to c-1 dimension. Where c represents the class number. For this reason, for a 4-class problem, fisher's LDA will project the data into a 3-dimensional space. Below is a mathematical calculation.

a) *Within Class Scatter Matrix:*

$$S_i = \sum_{x \epsilon D_i} (x - m_i) * (x - m_i)^T \tag{14}$$

Where

$$m_i = \frac{1}{n_i} \sum_{x \epsilon D_i} x$$

$$S_w = \sum_{i=1}^{c} S_i$$

m_i means the average value for each class. Here i is expressed as the number of classes.

b) *Between Class Scatter Matrix:*

$$S_b = \sum_{i=1}^{c} n_i * (m_i - m) * (m_i - m)^T \tag{15}$$

Where

$$m = \frac{1}{n} \sum x$$

Here m corresponds to the mean of all data. m_i means the average value for each class. n_i corresponds to the number of elements for each class.

The eigenvalue and eigenvectors of the matrix A obtained by multiplying these two matrices are found. Here, the eigenvectors corresponding to the largest (c-1) particle eigenvalue of the A matrix are selected. When the created W matrix is multiplied with the data set, the new data set obtained is the data set with reduced size.

$$A = (S_w^{-1} * S_b) \tag{16}$$

Where

$$W = \begin{bmatrix} w_{1,1} & \cdots & w_{1,(c-1)} \\ \vdots & \ddots & \vdots \\ w_{d,1} & \cdots & w_{d,(c-1)} \end{bmatrix}$$

$$\underline{y} = W^T * \underline{x}$$

The W matrix is obtained by using the training dataset and by multiplying the test dataset with the W matrix, it is actually provided to project on the same plane as the training dataset.

The distribution of test and training data in time space in 3-dimensional space is shown below.

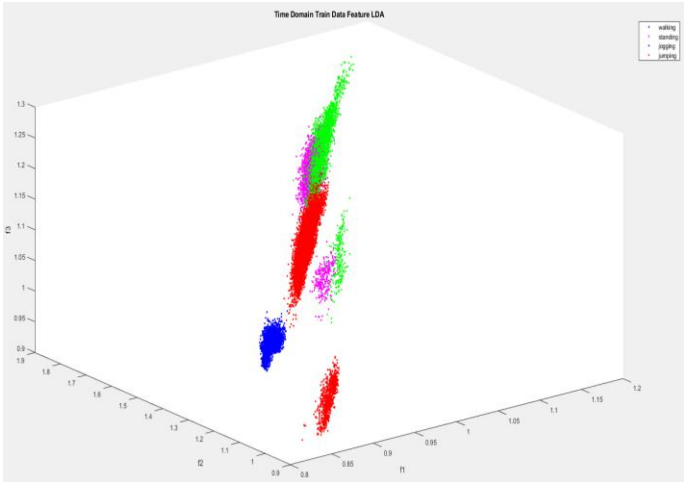


Fig. 8: Fisher LDA Training Dataset

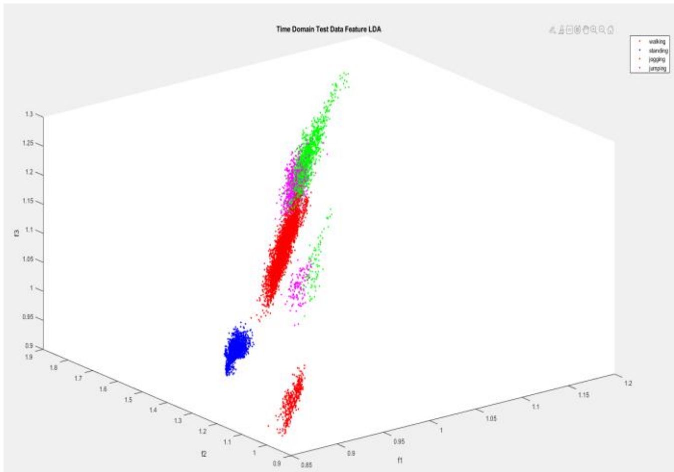


Fig. 9: Fisher LDA Test Dataset

VII. Classification

In this study, 2 different classifiers were designed. The first of the classifiers is the nonparametric k-NN classifier [35]. The other classifier is the perceptron algorithm [36]. Here, the Gradient Descent algorithm is used as the loss function. Since it is a 4-class problem, the data was expanded using Kessler's Reconstruct method and a classifier was designed in that way.

A. k-NN Rule (k Nearest Neighbor Rule)

The tags of the nearest k neighbors [35] are looked at and it is the non-parametric classifier that tells the most tags in which class the test point is in that class.

In this study, the classification results are presented by selecting different k values. In addition, Euclidean distance and manhattan distances were used to find the closest point, and their effects on success were examined.

Results are presented using two different comparison scenarios.

Scenario-1: Different size reduction method has been used.

Scenario-2: Dimension reduction is as in scenario-1. The k-NN algorithm is made using the different distance calculation function.

a) *Scenario-1 Results*: k = 55 and distance calculation was done by Euclid.

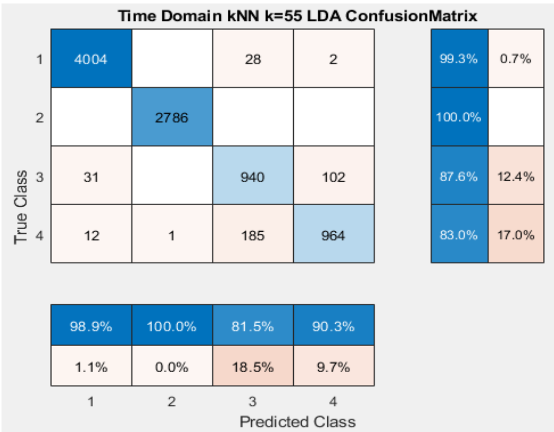


Fig. 10

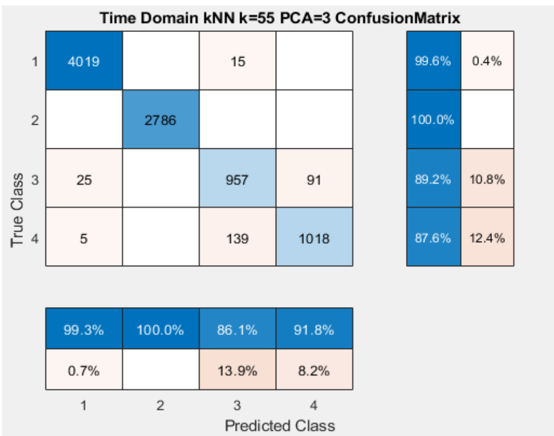


Fig. 11

b) *Scenario-1 Results:* k = 55 and the distance calculation was made with Manhattan.

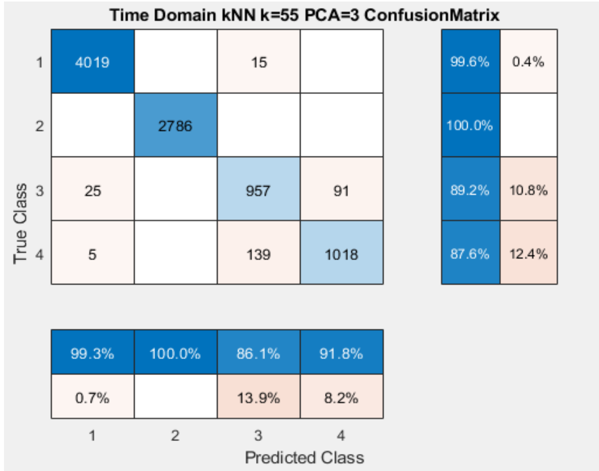


Fig. 12

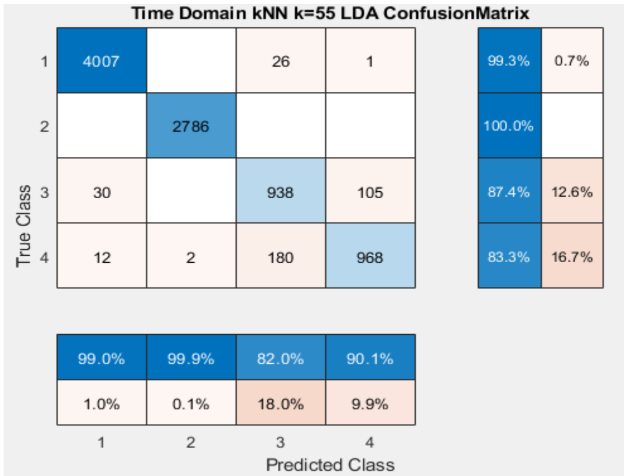


Fig. 13

B. Perceptron Algorithm

For the perceptron algorithm [36], which is a linear classifier, first of all, the data set obtained as a result of dimension reduction is expanded according to Kessler's reconstruction. Then, using the Gradient Descent Algorithm, the W weight vector is obtained by updating at each iteration. Then by checking the state $g_i(x) > g_j(x)$, x point i. assigned to the class. The results obtained are shown below for the different size reduction method.

Learning ratio for the following results: 0.0002 and 200 iterations were run.

The results are performed using different parameters for both classification algorithms. Time and frequency domain results are given in the following results table with success percentages.

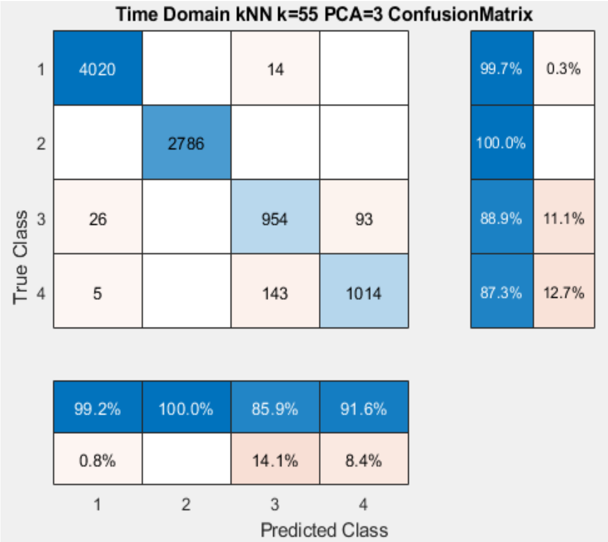


Fig. 14: Fisher's LDA Result

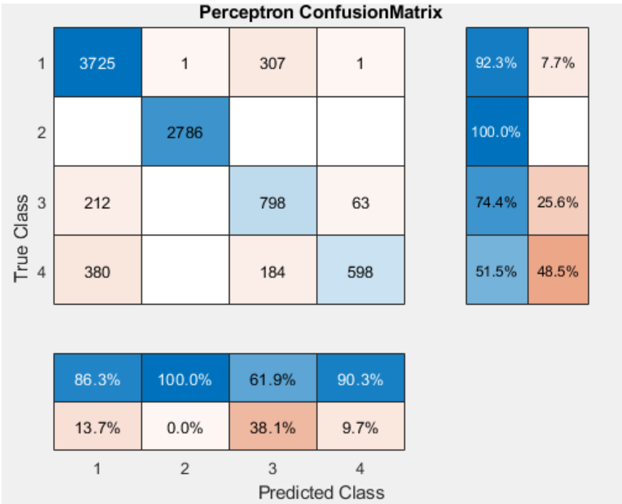


Fig. 15: PCA k = 3 Result

a) *Wit Time Space Results*

TABLE VIII: K-NN ALGORITHM RESULTS

K Point Number	Distance Calculation	Size Reduction Method	Accuracy (%)
55	euclidean	PCA = 3	96.078
55	manhattan	PCA = 3	96.896
55	euclidean	Fisher's LDA	96.013
55	manhattan	Fisher's LDA	96.068
55	euclidean	PCA = 8	98.023
201	euclidean	Fisher's LDA	95.527
201	euclidean	PCA = 3	95.860
201	euclidean	PCA = 8	96.887

TABLE IX: PERCEPTRON ALGORITHM RESULTS

Learning Rate	Number of Iterations	Size Reduction Method	Accuracy (%)
0.002	300	PCA = 3	89.917
0.002	300	Fisher's LDA	91.297
0.0002	300	PCA = 3	92.297
0.02	300	PCA = 3	91.198
0.02	300	Fisher's LDA	89.342

b) *Frequency Space Results*

TABLE X: K-NN ALGORITHM RESULTS

K Point Number	Distance Calculation	Size Reduction Method	Accuracy (%)
11	euclidean	PCA = 3	96.527
11	manhattan	PCA = 3	96.180
11	euclidean	Fisher's LDA	92.708
11	manhattan	Fisher's LDA	93.055
11	euclidean	PCA = 8	99.305
45	euclidean	Fisher's LDA	93.229
45	euclidean	PCA = 3	95.315
45	euclidean	PCA = 8	98.437

TABLE XI: PERCEPTRON ALGORITHM RESULTS

Learning Rate	Number of Iterations	Size Reduction Method	Accuracy (%)
0.002	300	PCA = 3	93.402
0.002	300	Fisher's LDA	91.840
0.0002	300	PCA = 3	93.618
0.02	300	PCA = 3	90.451
0.02	300	Fisher's LDA	93.402

VIII. CONCLUSION

In the human activity classification study, the problem of classifying 4 different activities was addressed. The problem has been analyzed in two different ways, in time and frequency domain. Different parameters and different scenarios were realized in both spaces and scenario-based confusion matrix representation was also shown in detail in tables for different parameters.

The following results caught my attention in this study. PCA size reduction projects more distinct class features than Fisher's LDA. This situation can be associated with high accuracy in scenarios where PCA is used in the classification results. Another result was seen when examining the effect of distance calculation method on accuracy in scenarios using k-NN classifier. It shows that in the case where PCA is used in Time Space, the calculation of manhattan distance gives better results than Euclidean distance. But it shows that Euclidean distance works better in the frequency space data set.

In general, it can be said within the scope of this study that PCA = 8 has the best results for both spaces and generally the k-NN classification algorithm gives better results than the perceptron. My favorite method for this study is the PCA = 8 k = 55 k-NN algorithm for time space. It is the k-NN method for the frequency domain for PCA = 8 and k = 11.

Acknowledgment

We would like to cordially thank all the teachers of the department of Computer Science and Engineering, Islamic University-Bangladesh and most importantly our beloved parents for their continuous support.

References

[1] J. Morales, and D. Akopian, “Physical activity recognition by smartphones, a survey,” *Biocybernetics and Biomedical Engineering*, vol. 37, no. 3, pp. 388-400, 2017, doi: 10.1016/j.bbe.2017.04.004.

[2] J. Qi, P. Yang, A. Waraich, Z. Deng, Y. Zhao, and Y. Yang, “Examining sensor-based physical activity recognition and monitoring for healthcare using internet of things: A systematic review,” *Journal of Biomedical Informatics*, vol. 87, pp. 138-153, 2018, doi: 10.1016/j.jbi.2018.09.002.

[3] R.-A. Voicu, C. Dobre, L. Bajenaru, and R.-I. Ciobanu, “Human physical activity recognition using smartphone sensors,” *Sensors*, vol. 19, no. 3, p. 458, 2019, doi: 10.3390/s19030458.

[4] A. Bayat, M. Pomplun, and D. A. Tran, “A study on human activity recognition using accelerometer data from smartphones,” *Procedia Computer Science*, vol. 34, pp. 450-457, 2014, doi: 10.1016/j.procs.2014.07.009.

[5] P. Casale, O. Pujol, and P. Radeva, “Human activity recognition from accelerometer data using a wearable device,” *Pattern Recognit. Image Anal.,* 6669, 2011, pp. 289-296, doi: 10.1007/978-3-642-21257-4_36.

[6] S. Mysore, “Human activity recognition from sensor data,” 2018.

[7] J. Lester, T. Choudhury, and G. Borriello, “A practical approach to recognizing physical activities,” *Pervasive*, 2006.

[8] W. Xingfeng, and K. Heecheol, “Detecting user activities with the accelerometer on android smartphones,” *Journal of Mutimedia Information System*, vol. 2, no. 2, pp. 207-214, 2015.

[9] D. M. Pober, J. Staudenmayer, C. Raphael, and P. S. Freedson, “Development of novel techniques to classify physical activity mode using accelerometers,” *Med Sci Sports Exerc.*, vol. 38, no. 9, pp. 1626-1634, 2006.

[10] J.-H. Cho, J. T. Kim, and T.-S. Kim, “Smart phone-based human activity classification and energy expenditure generation in building environments,” *Proc. 7th Int. Symp. Sustain. Healthy Buildings*, 2012.

[11] N. Ravi, N. Dandekar, P. Mysore, and M. L. Littman, “Activity recognition from accelerometer data,” AAAI, 3, pp. 1541-1546, 2005.

[12] A. Mannini, and A. M. Sabatini, “Machine learning methods for classifying human physical activity from on-body accelerometers,” *Sensors*, vol. 10, no. 2, pp. 1154-1175, 2010, doi: 10.3390/s100201154.

[13] H. Chen, S. Mahfuz, and F. Zulkernine, “Smart phone based human activity recognition,” *2019 IEEE Int. Conf. Bioinformatics and Biomedicine (BIBM),* 2019, pp. 2525-2532, doi: 10.1109/BIBM47256.2019.8983009.

[14] https://bmi.hmu.gr/the-mobifall-and-mobiact-datasets-2/

[15] D. Micucci, M. Mobilio, and P. Napoletano, “UniMiB SHAR: A new dataset for human activity recognition using acceleration data from smartphones,” *Applied Sciences,* vol. 7, no. 10, p. 1101, 2017, doi: 10.3390/app7101101.

[16] G. Vavoulas, C. Chatzaki, T. Malliotakis, M. Pediaditis, and M. Tsiknakis, “The MobiAct dataset: Recognition of activities of daily living using smartphones,” *2nd Int. Conf. Inf. and Commun. Technologies for Ageing Well and e-Health*, 2016, pp. 143-151, doi: 10.5220/0005792401430151.

[17] S. Fan, Y. Jia, and C. Jia, “A feature selection and classification method for activity recognition based on an inertial sensing unit,” *Information*, vol. 10, no. 10, p. 290, 2019, doi: 10.3390/info10100290.

[18] S. J. Preece, J. Y. Goulermas, L. P. J. Kenney, D. Howard, K. Meijer, and R. H. Crompton, “Activity identification using body-mounted sensors - A review of classification techniques,” *Physiological Measurement*, vol. 30, no. 4, R1-33, 2009, doi: 10.1088/0967-3334/30/4/R01.

[19] S. Rosati, G. Balestra, and M. Knaflitz, “Comparison of different sets of features for human activity recognition by wearable sensors,” *Sensors*, vol. 18, no. 12, p. 4189, 2018, doi: 10.3390/s18124189.

[20] https://actigraphcorp.force.com/support/s/article/What-is-the-feature-extraction-tool-and-how-does-it-work

[21] J. Zhu, R. San-Segundo, and J. M. Pardo, “Feature extraction for robust physical activity recognition,” *Human-Centric Computing and Information Sciences*, vol. 7, 2017, Art. no. 16, doi: 10.1186/s13673-017-0097-2.

[22] B. Pappachan, W. Caesarendra, T. Tjahjowidodo, and T. Wijaya, “Frequency domain analysis of sensor data for event classification in real-time robot assisted deburring,” *Sensors*, vol. 17, no. 6, p. 1247, 2017, doi: 10.3390/s17061247.

[23] Md. Aktaruzzaman, N. Scarabottolo, and R. Sassi, “Parametric estimation of sample entropy for physical activity recognition,” *2015 37th Annu. Int. Conf. IEEE Eng. Medicine and Biol. Soc. (EMBC)*, Milan, Italy, Aug. 25-29, 2015, doi: 10.1109/EMBC.2015.7318401.

[24] C. Reining, F. M. Rueda, M. ten Hompel, G. A. Fink, and F. Niemann, “Human activity recognition for production and logistics - A systematic literature review,” *Information*, vol. 10, no. 8, p. 245, 2019, doi: 10.3390/info10080245.

[25] I. M. Pires, N. Garcia, N. Pombo, F. Flórez-Revuelta, S. Spinsante, M. C. C. Teixeira, and E. Zdravevski, “Pattern recognition techniques for the identification of activities of daily living using mobile device accelerometer,” 2018, doi: 10.7287/peerj.preprints.27225.

[26] Y.-L. Hsu, S.-C. Yang, H.-C. Chang, and H.-C. Lai, “Human daily and sport activity recognition using a wearable inertial sensor network,” *IEEE Access*, pp. 1-1, 2018, doi: 10.1109/ACCESS.2018.2839766.

[27] B. Prapochanang, T. Leauhatong, and N. Nattaitanakul, “A novel dimension reduction technique for physical activity recognition from a waist-mounted accelerometer,” pp. 532-536, 2016, doi: 10.12792/iciae2016.097.

[28] I. E. Moudden, H. Jouhari, S. E. Bernoussi, and M. Ouzir, “Learned model for human activity recognition based on dimensionality reduction,” 2018.

[29] I. E. Moudden, Benyacoub, and S. E. Bernoussi, “Modeling human activity recognition by dimensionality reduction approach,” 2016.

[30] M. Abidine, B. Fergani, and I. Menhour, “Activity recognition from smartphones using hybrid classifier PCA-SVM-HMM,” pp. 1-5, 2019, doi: 10.1109/WINCOM47513.2019.8942492.

[31] M. Abidine, and B. Fergani, “Evaluating a new classification method using PCA to human activity recognition,” *Int. Conf. Comput. Medical Appl. (ICCMA’2013),* 2013, doi: 10.1109/ICCMA.2013.6506158.

[32] K. Walse, R. V. Dharaskar, and V. M. Thakare, “PCA based optimal ANN classifiers for human activity recognition using mobile sensors data,” 2016, doi: 10.1007/978-3-319-30933-0_43.

[33] Y. Saez, A. Baldominos, and P. Isasi, “A comparison study of classifier algorithms for cross-person physical activity recognition,” *Sensors*, vol. 17, no. 1, p. 66, 2017, doi: 10.3390/s17010066.

[34] X. Xi, M. Tang, S. M. Miran, and Z. Luo, “Evaluation of feature extraction and recognition for activity monitoring and fall detection based on wearable sEMG sensors,” *Sensors*, vol. 17, no. 6, p. 1229, 2017, doi: 10.3390/s17061229.

[35] A.-M. Mandong, and U. Munir, “Smartphone based activity recognition using K-Nearest Neighbor algorithm,” 2018.

[36] S. Shalev-Shwartz, “Perceptron algorithm,” 2008, doi: 10.1007/978-0-387-30162-4_287.