International Journal of Technology (2017) 5: 968-978
ISSN 2086-9614

RECOGNIZING COMPLEX HUMAN ACTIVITIES USING HYBRID FEATURE SELECTIONS BASED ON AN ACCELEROMETER SENSOR

M.N.Shah Zainudin [1,2*], Md Nasir Sulaiman[1], Norwati Mustapha[1], Thinagaran Perumal[1], Raihani Mohamed[1]

[1]*Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia.*
[2]*Faculty of Electronics and Computer Engineering, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia.*

(Received: December 2016/ Revised: October 2017 / Accepted: October 2017)

ABSTRACT

Wearable sensor technology is evolving in parallel with the demand for human activity monitoring applications. According to World Health Organization (WHO), the percentage of health problems occurring in the world population, such as diabetes, heart problem, and high blood pressure rapidly increases from year-to-year. Hence, regular exercise, at least twice a week, is encouraged for everyone, especially for adults and the elderly. An accelerometer sensor is preferable, due to privacy concerns and the low cost of installation. It is embedded within smartphones to monitor the amount of physical activity performed. One of the limitations of the various classifications is to deal with the large dimension of the feature space. Practically speaking, a large amount of memory space is demanded along with high processor performance to process a large number of features. Hence, the dimension of the features is required to be minimized by selecting the most relevant feature before it is classified. In order to tackle this issue, the hybrid feature selection using Relief-f and differential evolution is proposed. The public domain activity dataset from Physical Activity for Ageing People (PAMAP2) is used in the experimentation to identify the quality of the proposed method. Our experimental results show outstanding performance to recognize different types of physical activities with a minimum number of features. Subsequently, our findings indicate that the wrist is the best sensor placement to recognize the different types of human activity. The performance of our work also been compared with several state-of-the-art of features for selection algorithms.

Keywords: Accelerometer; Differential evolution (D); Evolutionary algorithm (EA); PSO; Genetic algorithm (GA); Particle swarm optimization (PSO); Relief-f; Tabu search algorithm

1. INTRODUCTION

Human Activity Recognition (HAR) application has recently gained attention in the intelligent environment field. In such states, monitoring human activity might be extremely important to reduce the fraction of unhealthy conditions. According to the 2016 report from the World Health Organization (WHO), the percentage of diabetes patients has increased incrementally in the world population (WHO, 2016).

*Corresponding author's email: noorazlan@utem.edu.my, Tel: 6019-5654406, Fax: 606-5552001
Permalink/DOI: https://doi.org/10.14716/ijtech.v8i5.879

In looking at this matter, insufficient physical activity is one of the issues. Regular exercise can be thought as one of the simplest solutions, by spending time in at least twice a week engaged in some physical exercises. Also, with the advancement of sensing technology, the use of inertial sensors offers a possible solution. Inertial sensors, such as an accelerometer and gyroscope, provide opportunities to undergo the HAR application process and these sensors have also been equipped in various smartphone models. Hence, everyone can track and monitor their daily exercise without relying on any other additional devices. These micro-machine electromechanical systems (MEMs) sensors are able to record the signals in three-dimensional spaces, where the x-axis (left-and-right movement), the y-axis (up-and-down movement) and the z-axis (upward/backward movement) are monitored (Acharjee et al., 2015). The signal is recorded by quantifying the sense of vibration through the device when movement is triggered. Even so, the choice of the sensor placement also effects to the classification performance (Avci et al., 2010). So, the position of the sensor placement needs to be clearly identified which to ensure it is able to recognize different kinds of actions, particularly in recognizing complex activities. These complex activities could be considered as the activity which consists of a sequence of actions arising from several different parts of the human body.

Also, to deal with the abundance number of features is another challenge. Practically speaking, the training model complexity and the processing time is strongly related to the numbers of features to be processed (Catal et al., 2015). In such states, feature selection was 'pruned' by removing the less significant features before each feature is classified. The features that do not contribute enough information to be described within the particular class are removed from the feature space.

In this article, several contributions were carried out. A hybrid feature selection method using Relief-f feature ranking with a well-known evolutionary algorithm, known as differential evolution (DE) was proposed to select the most relevant features. Secondly, we also proposed an adaptive parameter mechanism without relying on the exhaustive process to find the optimum parameter values. Lastly, our proposed feature selection method also performed an outstanding degree of accuracy which was better than several well-known feature selection algorithms, such as particle swarm optimization (PSO), evolutionary algorithm (EA), genetic algorithm (GA) and the Tabu search algorithm. This paper is organized as follows: Section II explains the background work; Section III describes the materials and methods; Sections IV discusses the proposed feature selection.; Section V presents the results and discussion; Section VI presents the conclusions.

2. RELATED WORKS

As previously mentioned, one of the challenges pertaining to classification accuracy is to deal with the huge number of meaningless features. Hence, by eliminating the irrelevant or redundant features, the processing time tends to be minimized, which yields an improvement in classification performance. The feature selection method is divided into two major categories, namely wrapper and filter (Olvera-Lopez et al., 2010). In the first category, the method relies on the predictor to optimize the selection of the features in the selection process. Meanwhile, the second category relies on the general characteristics of the training data and the feature selection process continues as a pre-processing step with an independence induction algorithm. According to Hall and Smith (1998), the correlation based features selection (CFS) method is proven in reducing the number of features for real or artificial data. Arif et al. (2014) carried out research on the accelerometer performance, using a kNN model and using CFS with a reduced scatter search feature selection. Other related work in feature selection also been done by Akhavian and Behzadan (2015). The respective authors proposed two different filter strategies, namely CFS and Relief-f feature selection in recognizing the construction activity. Challita et al. (2015) proposed work on a combination of Elastic Net and Relief to select features in order to predict the behavior of rotation machine.

In other works, Capela et al. (2015) used Relief-f, CFS, and fast correlation methods to recognize human activity from three different types of users, namely able-bodied, elderly, and stroke patients. On the other hand, an evolutionary algorithm (EA) also gained an attention among researchers in feature reduction. Khushaba et al. (2011) proposed a feature selection method by utilizing differential evolution (DE) method with wavelet packet transform (WPT) to search for the best subset features in solving the global optimization problem. The results received also have been compared with two widely known optimization algorithms, namely genetic algorithm (GA) and particle swarm optimization (PSO). Nwankwor et al. (2013) proposed a new hybrid model of PSO with DE called particle swarm differential evolution (HPSDE) for use in solving the dimension problems in the engineering domain. The HPSDE method was reported to be able to perform an outstanding degree of accuracy when compared with the other two methods, namely DE and PSO algorithms, respectively. Another work reported from Pant and Thangaraj (2008) in tackling the feature dimension problem. In their work, a hybrid model of DE-PSO is proposed to prove the effectiveness in solving various real-life problems and other optimization problems.

Another work related to the DE algorithm also has been reported from Omran et al. (2005). In their article, a self-adaptive differential evolution (SDE) is proposed to solve various optimization problems. In such situations, the performance of the proposed SDE method capable of achieving a decent performance, which is better than the other DE-version algorithms in all the benchmark functions. Ghosh et al. (2013) proposed a feature subsets generation method, using self-adaptive differential evolution called SADE. The Relief-f feature ranking method has also been applied to remove duplicate features. A fuzzy k-nearest neighbor (kNN) classifier is used to validate the performance of the proposed method. Apolloni et al. (2016) combined a wrapper feature selection method based on a binary differential evolution (BDE) algorithm with a rank-based filter feature selection method. Other works, including those by Ijjina and Mohan (2014) utilized an evolutionary algorithm in human activity recognition using a genetic algorithm (GA) and deep convolution neural network (CNN). The authors reported the result shows a high level of improvement in the performance of the classifier. Das et al. (2015) combined a swarm algorithm, using particle swarm optimization (PSO) with multilayer perceptron (MLP) to classify the tasks undertaken in data mining. The authors also compared the proposed method with MLP and GA-MLP methods. Additionally, Prasad et al. (2015) also proposed a hybrid PSO-GA with the MLP method to solve a complex problem in data mining. The proposed method shows results that are statically significant in terms of steadiness and efficiency.

3. METHODOLOGY

3.1. PAMAP2 Datasets

In our study, a public dataset compiled on activity recognition from PAMAP2 (Reiss & Stricker 2012) was used. Three Inertial Measurement Units (IMUs) were placed at a dominant position on the wrist, chest and on the dominant side of the ankle. Each IMU consists of a 3D accelerometer sensor, a 3D gyroscope sensor, a 3D magnetometer sensor, and a temperature and an orientation sensor. A 100-Hz sampling rate was used during the data collection. Nine subjects, including one woman and eight men, were asked to perform several activities. Each subject was asked to complete eighteen activities, such as lying down, sitting, standing, walking, running, cycling, Nordic walking, watching TV, computer work, car driving, ascending walking, descending walking, ironing, folding laundry, house cleaning, playing soccer and rope jumping. In this study, only the sensor signals obtained from an accelerometer sensor placed on the wrist were used.

3.2. Window Segmentation

In order to extract the features from these recorded accelerometer signals, the raw data stream needed to be divided into several segments before applying any further calculations. A sliding window, sized at 6 seconds with a 50% overlap between two consecutive window segments was applied. This amount is believed to be sufficient to describe the activity. Meanwhile, overlapping was chosen to reduce the probabilities of error state noise since there were transitions between two or more activities (Su et al., 2014). So, every window segment was given 64 samples with 32 samples overlapping between two consecutive windows.

3.3. Feature Extraction

Originally, the sensor data stream consisted of a limited number of characteristics to describe the activity. In such states, the correct classification rate might decrease when using a limited number of features. Therefore, several additional features need to be extracted to help the classifier model to learn about more characteristic in the activity class. The feature extraction aims to discover the characteristics of each activity by reducing the representation of data. During this process, several features in the collection of time domain features, such as minimum and maximum values, mean, variance, standard deviation, skewness, kurtosis, and correlation, were extracted. Due to the ease and the direct extraction of figures out of the window segment, time domain features were applied. The extracted signal is referred as a feature vector and later it will be used as an input variable for the classifier model (Su et al., 2014; Avci et al., 2010). The success of the chosen features is measured in how accurately the classifier model is able to differentiate and recognize the activity.

4. PROPOSED FEATURE SELECTION

This section describes the proposed feature selection method in detail. In this section, a hybrid feature selection method utilizes filter ranking approach, namely Relief-f, which is combined with most widely-used evolutionary algorithms such as the differential evolution. The detailed explanation of our proposed method is described in the next subsection.

4.1. Relief-f Feature Ranking Strategy

Relief-f is one of the most widely known feature ranking methods used to deal with the problem of feature dimensionality. Relief-f is extensively applied in solving various feature selection problems, including those involved in solving the dimension of the microarray DNA data and hyperspectral images. This method ranks the features according to a method of scoring weight related to a distance function. Then, each one of the features is rearranged, according to its relevance and then, the most highly ranked features are selected and applied to a predictor (Robnik-Siknja & Kononeko, 2003). In each instance, this method finds the nearest hit (data point from the same class) and nearest misses (data point from a different class). These data are calculated by weighting them based on their relevance (Akhavian & Behzadan, 2015; Capela et al., 2016). Equation 1 below shows how to calculate the feature weight based on its relevance.

$$w_i = \sum_{j=1}^{N} (x_i^j - nearmiss(x^i)_j)^2 - (x_i^j - nearhit(x^i)_j)^2 \tag{1}$$

where w is the weight or scoring value of the i^{th} feature, x_i^j is the value of the i^{th} feature for a point x^j and N is the total number of data points. $nearhit\ x^j$ and $nearmiss\ x^j$ are the nearest data points to x^j in the same and different classes, respectively. A Manhattan distance function with the number of nearest neighbors, calculated as being 3 was applied. In order to minimize the number of features, the feature boundary threshold is proposed. In this study, a threshold value of $w_i = 0.02$ is applied to define the feature boundary. In such circumstances, any feature value below the scoring value of $w_i < 0.02$ is removed before it is fed into the DE algorithm.

4.2. Feature Extraction

Storn and Price (1997) proposed the DE algorithm, which is simple and easy to implement, to solve the various optimization problems. Since the searching process is done in parallel with direct search, the DE algorithm has an advantage compared to other evolutionary algorithms. It is easy to use and it has good convergence and fast implementation properties. However, the DE algorithm utilizes the same evolutionary operators like selection, recombination, and mutation as the very well-known algorithm genetic algorithm (GA) (Khushaba et al., 2008). In comparison with GA, the DE algorithm uses distance and direction information from the current population in the searching process. So, the performance of the DE algorithm depends on the manipulation of the target vector and difference vector in order to obtain a trial vector. The first step in the DE algorithm is to create a population of NP members, each of D-dimensional real-value parameters, where NP) is the population size, and D is the number of parameters to be optimized. Mutation is the main operator in DE and in order to create a trial vector, the weight difference vector between two population members x_{r1} and x_{r2} are added to a third member x_{r0}. In D-dimensional search space, each target vector $x_{i,j}$, is a mutant vector create from the current generation g, as shown in Equation 2 below:

$$v_{j,i,g} = x_{j,r0,g} + F * \left(x_{j,r1,g} - x_{j,r2,g}\right) \tag{2}$$

where $r_0, r_1, r_2 \in \{1,2,3, \ldots, NP\}$ are randomly chosen integers that must be different values from each other and also be different from the running index i. $F \in (0,1)$ is a scaling factor that controls the rate which the population evolves. In this work, the value of is adaptively changed. In order to increase the diversity of the perturbed parameter vectors, a uniform crossover is introduced to build trial vectors out of parameter values that have been copied from two different vectors. This process is also known as discrete recombination (Khushaba et al., 2011). The parent vector or real vector is mixed with the mutated vector to produce a trial vector $u_{ji,g}$ as shown in Equation 3 below:

$$u_{ji,g+1} = \begin{Bmatrix} v_{j,i,g} \ \ if\ rand\ (0,1) \ll c_r \\ x_{j,i,g} \qquad otherwise \end{Bmatrix} \tag{3}$$

where $u_{ji,g}$ is the j^{th} dimension from i^{th} trial vector along the current population g. The crossover probability $c_r \in (0,1)$ is a user-defined value that controls the fraction of the parameter values that are copied from the mutant. Selection is the next step to choose the vector between the target vector and the trial vector with the aim of creating an individual vector for the next generation. If the newly-generated vector results in a lower objective function value (better fitness) than the predetermined population member, then the resulting vector replaces the initial vector with which it was compared (Palit & Popovic 2005).

At this stage, the selected ranking features were applied as input for the DE algorithm. The number of desired features (DNF), the size of population (PSIZE) and the number of generations (GEN) composes the necessary parameter, which needs to be defined. Khushaba et al. (2011) stated that the value of PSIZE and GEN is initialized as 50 respectively. In order to minimize the searching complexity, the number of PSIZE and GEN were adaptively defined according to the number of 'pruned' features. So, this way, the number of PSIZE and GEN variables has been initialized from the number of ranking features, accordingly.

5. EXPERIMENTAL RESULTS

This section describes the experimental procedures and results of the proposed work. In order to go through the experiment, we managed and conducted the experiment using two different tools. The necessary process, including the preprocessing, segmentation, data extraction, and feature selection was done using the Matlab 2013 tool. Meanwhile, widely-recognized machine learning tools (WEKA) were used to evaluate the performance of the proposed method based on several performance metrics (Hall & Smith, 1998). Each window segment went through the extraction process and 8 features were derived from each window segment. Hence, this process resulted in a total of 24 features (8 features × 3 dimensions).

Average accuracy was calculated to measure the average classification performance. However, this metric might unsuitable to measure the performance of imbalanced class distribution. Hence, additional metrics like precision, recall, F-measure were used (Wang et al., 2015; Zhang et al., 2015). The experiment was done under two conditions; using all extracted features (FS1) and using reduced features (FS2). In order to validate our classification performance, the extracted data were divided into two different groups of subsets where 70% was used for training and 30% was reserved for testing. In the training process, the 10-fold cross validation strategy was applied. The training dataset was divided into 10 equal sizes of subsets. In each run, 9 subsets were applied to train the model and the remaining 1 subset was reserved for testing. This process was repeated in 10 times and average performance was calculated to produce the final predictive result. In this work, we use the random forest classifier model to figure out the quality of the performance of the proposed method. Table 1 presents the classification performance of training subsets from all features (FS1).

Table 1 Classification results of training subsets from all features FS1

Activity	Accuracy	Precision	Recall	F-measure
Laying down	0.980	0.994	0.980	0.987
Sitting	1.000	0.999	1.000	0.999
Standing	0.986	0.995	0.986	0.990
Walking	0.975	0.974	0.975	0.974
Running	1.000	0.998	1.000	0.999
Cycling	0.996	0.999	0.996	0.997
Nordic walking	1.000	0.993	1.000	0.996
Watching TV	0.948	0.998	0.948	0.972
Computer work	0.994	0.986	0.994	0.990
Car driving	0.968	0.989	0.968	0.979
Ascending walk	0.992	0.996	0.992	0.994
Descending walk	0.997	0.991	0.997	0.994
Laundry	0.973	0.967	0.973	0.970
House cleaning	0.999	0.982	0.999	0.990
Soccer	0.992	0.986	0.992	0.989
Rope jumping	0.987	0.984	0.987	0.985
Average	0.987	0.987	0.987	0.987

It is clearly being seen that most activities were recorded above 94% on average. However, walking and laundry were recorded at a slightly lower level of performance than the others, below 97%. Next, we managed the experiment to evaluate the effectiveness of our proposed method.

As discussed in section 4, a two-stage hybrid feature selection is introduced. In the first stage, the extracted features were ranked to measure the scoring value. Thereafter, the ranking features later on were ‘pruned’ according to the value of the specified feature boundary. In this case, we noticed that a feature score lower than 0.02 was believed to be irrelevant since there was no improvement in terms of accuracy when a feature valued at below the selected value was included. Consequently, only 20 features above the chosen feature boundary remained in the next process.

In the second stage, the selected 20 features were applied as data input to the DE algorithm. The number of desired features (DNF), the size of population (PSIZE) and the number of generations (GEN) is the necessary parameter needed to be defined. In order to minimize the searching complexity, the number of PSIZE and GEN parameters was adaptively defined, according to the number of ‘pruned’ features. So, in this way the number of PSIZE and GEN parameters is initialized as 20, accordingly. We also completed several experiments to define the optimum number of DNF parameters. In such states, 15 are considered as the most relevant number of features. We also noticed that the accuracy was decreased when we increased the number of features to be classified. Therefore, the generated feature subsets were comprised into new feature subsets represented by FS2.

Table 2 shows the classification result of training and testing for feature subsets FS2. There was an improvement in terms of accuracy; on average, 95% accuracy was received from all activities, while the F-measure and precision measures were recorded above 97%. The average accuracy of training subsets from all FS2 features was achieved above 98% for all metrics. Surprisingly, the activities thought to be the poorest from FS1 (walking and laundry), also shown an improvement, where 97.5% and 97% accuracy, respectively these were recorded in the same way. For testing, more than 97% of the precision and F-measures were obtained from all activities. However, walking was recorded the lowest among the others, where 97% accuracy was received. It could be proven that the wrist is not considered as a good sensor placement in the context of recognizing that walking primarily consists of dominant lower leg motions. Laundry also is measured at being somewhat lower than the others since this kind of activity may contain a sequence of actions to be performed, based on multiple features.

In order to validate the performance, we also made a comparison with several states-of-the-art feature selection methods. Several well-known feature selection algorithms, such as particle swarm optimization (PSO) (Das et al., 2015; Prasad et al., 2015), the evolutionary algorithm (EA) (Arif & Kattan, 2015), the genetic algorithm (GA) (Ijjina & Mohan, 2014; Das et al., 2015; Prasad et al., 2015) and the Tabu search algorithm (Arif & Kattan, 2015) were utilized. Table 3 shows the performance comparison in term of accuracy and the overall time taken to build the training model. PSO, EA, and GA were reported at 96% of accuracy, followed by the Tabu search algorithm, where 95% accuracy was obtained. Even though there is no large difference between our methods and others, the overall time taken recorded from our proposed method produced the fastest results, where a rate of 34.61 seconds was obtained. The genetic algorithm (GA) recorded the longest time (45.17 seconds). On the other hand, we also compared the performance of our experimental results with several reported work in activity recognition. Table 4 shows the comparison with the previously reported work.

Table 2 Classification results of training and testing features FS2

Activity	Training		Testing	
	Accuracy	Precision	Accuracy	Precision
Laying down	0.981	0.993	0.982	0.993
Sitting	1.000	0.999	1.000	0.999
Standing	0.987	0.996	0.988	0.995
Walking	0.976	0.975	0.976	0.974
Running	1.000	0.998	0.998	0.997
Cycling	0.997	0.998	0.999	0.997
Nordic walking	1.000	0.994	1.000	0.994
Watching TV	0.950	0.998	0.952	0.997
Computer work	0.995	0.987	0.994	0.989
Car driving	0.966	0.989	0.943	0.978
Ascending walk	0.991	0.997	0.991	0.997
Descending walk	0.996	0.992	0.998	0.989
Laundry	0.974	0.970	0.975	0.968
House cleaning	0.999	0.983	0.999	0.983
Soccer	0.992	0.986	0.996	0.991
Rope jumping	0.989	0.982	0.989	0.989
Average	0.988	0.988	0.988	0.988

Table 3 Comparison with other feature selection algorithms

Algorithm	Accuracy (Training)	Accuracy (Testing)	Time (in seconds)
Particle swarm optimization (PSO)	0.960	0.962	35.04
Evolutionary algorithm	0.966	0.968	36.58
Genetic algorithm (GA)	0.967	0.969	45.17
Tabu search algorithm	0.956	0.958	36.02
Proposed method	0.988	0.988	34.61

Table 4 Comparison with previous related work

References	No. of activities	No. of participants	No. of features	Accuracy
Maurer et al. (2006)	6	6	134	80-84%
Sun et al. (2010)	7	6	66	94%
Karantonis et al. (2006)	10	6	-	90.8%
Allen et al. (2006)	8	6	25	91%
Mi-hee et al. (2009)	5	5	-	99%
Kwapisz et al. (2010)	6	29	43	91.7%
Catal et al. (2015)	6	36	43	97.2%
Kastner et al. (2013)	6	30	561	96%
Arif et al. (2014)	6	36	30	95.3%
Arif et al. (2015)	17	9	58	98%
Proposed method	**16**	**9**	**15**	**98.8%**

6. DISCUSSION

According to Table 4, our results show remarkable performance in comparison with other work by recognizing different activities perfectly. Although some of the activities recorded reflect somewhat unsatisfactory performance, we could state that our method is capable of producing a decent accuracy. Moreover, even though there was no enormous difference between the findings of Arif et al. (2015), we were able to minimize the number of features to be classified. The number of generating features was significantly decreased more than 70% than the reported work. Furthermore, the author also reported the accuracy had increased when they combined the sensor data stream of three sensor placements. In this case, we only used one sensor placement (the wrist) which could improve human comfort. Also, it was proved in our work that the performance of our proposed method also was able to show the amazing performance achieved in reducing the number of features in comparison with the other reported work.

7. CONCLUSION

An accelerometer sensor deeply set within a smartphone provides opportunities for researchers to facilitate their data collection in activity recognition. However, the high number of irrelevant features is believed to be an extremely important challenge. The model complexity as well as the time and memory space to process the enormous number of features is relatively increased when running a high number of irrelevant features. Additionally, accuracy also tends to decrease when including too many less informative features to classify. Hence, a hybrid feature selection method using Relief-f and differential evolution is proposed in order to minimize and eliminate the least significant features before it is fed into the classifier model. The PAMAP2 activity dataset is used as a benchmark study to figure out the quality of our performance for the proposed method. Our experimental results proven an outstanding level of performance in recognizing different physically complex activities in comparison with several state-of-the-art feature selection algorithms. For future work, we are planning to evaluate the performance in recognizing the activity using the combinations of several sensor placements attached to the subject's body. Additionally, we also encourage other researchers to improve upon our methods in different domains.

8. REFERENCES

Acharjee, D., Mukherjee, A., Mandal, J. K., & Mukherjee, N., 2016. Activity Recognition System using in-built Sensors of Smart Mobile Phone and Minimizing Feature Vectors. *Microsystem Technologies*, Volume 22(11), pp. 2715–2722. http://link.springer.com/10.1007/s00542-015-2551-2.

Akhavian, R., Behzadan, A.H., 2015. Construction Equipment Activity Recognition for Simulation Input Modeling using Mobile Sensors and Machine Learning Classifiers. *Advanced Engineering Informatics*, Volume 29(4), pp. 867–877

Apolloni, J., Leguizamón, G., Alba, E., 2016. Two-hybrid Wrapper-filter Feature Selection Algorithms Applied to High-dimensional Microarray Experiments. *Applied Soft Computing Journal*, Volume 38, pp. 922–932

Arif, M., Bilal, M., Kattan, A., 2014. Better Physical Activity Classification using Smartphone Acceleration Sensor. *Journal of Medical Systems* Volume 38(9), pp. 1–10.

Arif, M., Kattan, A., 2015. Physical Activities Monitoring using Wearable Acceleration Sensors Attached to the Body. *PLoS ONE*, Volume 10(7), pp.1–16

Avci, A., Bosch, S., Marin-Perianu, M., Marin-Perianu, R., & Havinga, P., 2010. Activity Recognition using Inertial Sensing for Healthcare, Wellbeing and Sports Applications: A Survey. n *Architecture of computing systems (ARCS), 2010 23rd international conference* *on*, pp. 1–10

Capela, N. A., Lemaire, E. D., Baddour, N., Rudolf, M., Goljar, N., & Burger, H., 2016. Evaluation of a Smartphone Human Activity Recognition Application with Able-bodied and Stroke Participants. *Journal of NeuroEngineering and Rehabilitation*, Volume 13(1), pp. 5

Capela, N.A., Lemaire, E.D., Baddour, N., 2015. Feature Selection for Wearable Smartphone-based Human Activity Recognition with Able-bodied, Elderly, and Stroke Patients. *PLoS ONE*, Volume 10(4), pp. 1–18

Catal, C., Tufekci, S., Pirmit, E., & Kocabag, G., 2015. On the Use of an Ensemble of Classifiers for Accelerometer-based Activity Recognition. *Applied Soft Computing Journal*, Volume ? pp. 1–5

Challita, N., Khalil, M., Beauseroy, P., 2015. New Techniques for Feature Selection: A Combination between Elastic Net and Relief. In *IEEE*. pp. 3–8

Das, H., Jena, A. K., Nayak, J., Naik, B., & Behera, H. S., 2015. Computational Intelligence in Data Mining. *Smart Innovation, Systems, and Technologies*, Volume 2, pp. 1–14

Ghosh, A., Datta, A., Ghosh, S., 2013. Self-adaptive Differential Evolution for Feature Selection in Hyperspectral Image Data. *Applied Soft Computing Journal*, Volume 13(4), pp. 1969–1977

Hall, M., Smith, L., 1998. Feature Subset Selection: A Correlation-based Filter Approach. *Progress in Connectionist-Based Information Systems*, Volume 1 and 2, pp. 855–858

Ijjina, E.P., Mohan, C.K., 2014. Human Action Recognition using Action Bank Features and Convolutional Neural Networks. *In*: 2014 Asian Conference on Computer Vision (ACCV), pp. 178–182

Khushaba, R. N., Al-Ani, A., AlSukker, A., & Al-Jumaily, A., 2008. A Combined Ant Colony and Differential Evolution Feature Selection Algorithm. *Ant Colony Optimization and Swarm Intelligence*, Volume 5217, pp. 1–12

Khushaba, R.N., Al-Ani, A., Al-Jumaily, A., 2011. Feature Subset Selection using Differential Evolution and a Statistical Repair Mechanism. *Expert Systems with Applications*, Volume 38(9), pp. 11515–11526

Kwapisz, J.R., Weiss, G.M., Moore, S., 2011. Activity Recognition using Cell Phone Accelerometers. *ACM SIGKDD Explorations Newsletter*, Volume 12, p.74

Nwankwor, E., Nagar, A.K., Reid, D.C., 2013. Hybrid Differential Evolution and Particle Swarm Optimization for Optimal Well Placement. *Computational Geosciences*, Volume 17(2), pp. 249–268

Olvera-Lopez, J. A., Carrasco-Ochoa, J. A., Martinez-Trinidad, J. F., & Kittler, J., 2010. A Review of Instance Selection Methods. *Artificial Intelligence Review*, Volume 34(2), pp. 133–143

Omran, M.G.H., Salman, A., Engelbrecht, A.P., 2005. Self-adaptive Differential Evolution. *Computational Intelligence and Security, Pt 1, Proceedings*, Volume 3801, pp. 192–199

Palit, A. K., & Popovic, D. (2005). Computational Intelligence in Time Series Forecasting. Automatisierungstechnik. http://doi.org/10.1007/1-84628-184-9

Pant, M., Thangaraj, R., 2008. Hybrid Differential Evolution-particle Swarm Optimization Algorithm for Solving Global Optimization Problems. Jurnal ? Volume ?(?), pp.

Prasad, C., Mohanty, S., Naik, B., Nayak, J., & Behera, H. S., 2015. Computational Intelligence in Data Mining. *Smart Innovation, Systems and Technologies*, Volume 1, pp. 1–14

Reiss, A., Stricker, D., 2012. Introducing a New Benchmarked Dataset for Activity Monitoring. *In*: Proceedings - International Symposium on Wearable Computers, ISWC, pp. 108–109

Robnik-Siknja, M., Kononeko, I., 2003. Theoretical and Empirical Analysis of RelifF and RReliefF. *Mach Learn*, Volume 53, pp. 23–69

Storn, R., Price, K., 1997. Differential Evolution–a Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. *Journal of Global Optimization*, Volume 11(4), pp. 341–359

Su, X., Tong, H., Ji, P., 2014. Activity Recognition with Smartphone Sensors. *Tsinghua Science and Technology*, Volume 19(3), pp. 235–249

Sun, L., Zhang, D., Li, B., Guo, B., & Li, S., 2010. Activity recognition on an accelerometer embedded mobile phone with varying positions and orientations. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Volume 6406 LNCS, pp. 548–562

Wang, C. Y., Hu, L. L., Guo, M. Z., Liu, X. Y., & Zou, Q., 2015. imDC: An Ensemble Learning Method for Imbalanced Classification with miRNA Data. *Genetics and Molecular Research*, Volume 14(1), pp. 123–133

WHO, 2016. WHO | World Health Day 2016: WHO Calls for Global Action to Halt the Rise in and Improve Care for People with Diabetes. Available online at http://www.who.int/mediacentre/news/releases/2016/world-health-day/en/, Accessed on April 8, 2016

Zhang, D., Ma, J., Yi, J., Niu, X., & Xu, X., 2015. An Ensemble Method for Unbalanced Sentiment Classification. *In*: International Conference on Natural Computation (ICNC). pp. 440–445