

RESEARCH

Open Access



Performance evaluation of multivariate statistical techniques using edge-enabled optimisation for change detection in activity monitoring

Naveed Khan^{1*}, Sally McClean¹, Shuai Zhang¹ and Chris Nugent¹

Abstract

The monitoring of human activities using simple body worn sensors is an important and emerging area of research in machine learning. The sensors capture a large amount of data in a short period of Time in a relatively un-obtrusive manner. The sensor data might have different transitions to be used for deification of different user activities. Therefore, change point detection can be used to classify the transition from one underlying distribution to another. The automatic and accurate change point detection is not only used for different events, however, can also be used for generating real world datasets and responding to changes in patient vital signs in critical situation. Moreover, the huge amount of data can use the current state-of-the-art cloud and edge computing platforms to process the change detection locally and more efficiently. In this paper, we used multivariate exponentially weighted moving Average (MEWMA) for online change point detection. Additionally, genetic algorithm (GA) and particle swarm optimization (PSO) is used to automatically identify an optimal parameter set by maximizing the F-measure. The optimisation approach is implemented over an edge cloud platform so that the data can be processed locally and more accurately. Furthermore, we evaluate our approach against multivariate cumulative sum (MCUSUM) from state-of-the-art in terms of different metric measures such as accuracy, precision, sensitivity, G-means and F-measure. Results have been evaluated based on real data set collected using accelerometer for a set of 9 distinct activities performed by 10 users for total period of 35 minutes with achieving high accuracy from 99.3% to 99.9% and F-measure up to 62.94%.

Keywords Multivariate exponentially weighted moving average, Genetic algorithm, Multivariate cumulative sum, Particle swarm optimization

Introduction

Currently, change point detection has been practiced in various application domains such as quality control and fault detection to maintain and improved the performance of industrial process [1], to detect changes in recognition oriented signals in order to make automatic

segmentation of signal [2] and healthcare [3] to detect change in patient's vital signs to alarm caregiver. The analysis of sequential change point detection is an observed process. The process could be a model to measure and quantify the continuous production process in order to identify the change point which might be cause of devaluation in quality that must be detected and corrected.

The recognition of user's activity is one of the keys to enable context aware system. Therefore, a technique is required to decrease labelling effort and generate more labelled data. The various machine learning technique

*Correspondence:

Naveed Khan
n.khan@ulster.ac.uk

¹ School of Computing, Ulster University, Belfast, UK

have been used for activity recognition (AR) which requires labelled data beforehand for classification [4]. The Crowd Labelling Application (CLAP) has been used to help in collecting large scale labelled activity data in free living environment. The CLAP consists of two main components the AR module and labelling prompt module. The AR module detect and identify the user activity based on Gaussian mixture model while the labelling prompt module display user activity icons on screen which enable user to click on what activity just performed and hence provide ground truth for the dataset [5]. The limitation of this application was to prompt user when each activity is followed by 'standing still' activity.

Hence, the monitoring of context aware system is a real world problem which requires sequential analysis of time series data to identify and detect change during the whole dynamic process. The change detection in these process corresponds to the identification of time points in which the parameters are subject to abrupt changes in tendency at prior unknown time instants [6, 7]. The abrupt change refers to the change in characteristics that happens more quickly with respect to the sample period of measurement. The change point detection has a new emerging application domain of autonomously detecting and identifying transitions in human activities and to monitor these over time such as "walk", "stand", "run", "sit" etc. The detection of these changes points can be utilised to prompt users to solicit activity labelling after switching to a new activity [5] or taking the starting point for window based activity recognition [8]. Moreover, modern smartphones are also equipped with the tiny inbuilt sensors such Accelerometer, Gyro, GPS etc. which can be used to collect social, physiological or environmental data [9]. These sensors are used to detect different transition of movement patterns for various user activities [10].

Change point detection can be online or offline and is used to identify the transition from one underlying distribution to another. However, our focus is on online change point detection because it used in real time system to observe monitor and evaluate data concurrently as available. Such approach is fast, sequential and reduce false alarms [11]. For the purpose of activity recognition, the accurate change point detection forms an intrinsic element of systems that needs user engagement to identify transition within an input data stream. As a key challenge in the overall applicability and usability of such systems depends on detecting changes autonomously in a data stream that correspond to a user's perception of change. Such user engagement might have negatively effect by prompting too often or prompting too little can weaken applicability of the application [12]. Also, the autonomous and accurate

change point in user activities requires the selection of lightweight algorithm to be implemented in an online detection scenario.

In this paper, we used multivariate approaches to analyze and evaluate multivariate data for automatic change point detection. In multivariate data analysis, more than one characteristics of a system evaluated simultaneously and also identify the relational among these characteristics. We proposed the MEWMA approach which tunes the different parameters such as lambda, which weights the current versus historical data, window size and significance value with the aim to achieve better performance and accurate change point detection. Also, we implement MCUSUM a multivariate approach form literature to use as a bench mark to our proposed technique. Moreover, the GA and PSO is used to automatically identify an optimal parameter set using different parameters for MEWMA and MCUSUM, so as to maximize the F-measure (objective function). The proposed scheme is analyzed using different metric measures and the experimental results show that the proposed scheme performs better than the bench mark scheme. The major contributions of our work can be summarized as given below.

- We use multivariate approaches to analyze and evaluate multivariate data for automatic change point detection.
- We proposed the MEWMA approach which tunes different parameters such as lambda, which weights the current versus historical data, window size and significance value with the aim to achieve better performance and accurate change point detection.
- We implement MCUSUM a multivariate approach form the existing literature to use as a benchmark to our proposed technique.
- The GA and PSO is used over an edge-cloud platform to automatically identify an optimal parameter set using different parameters for MEWMA and MCUSUM, so as to maximize the F-measure (objective function).

The remainder of this paper is structured as follows. [Related work](#) section presents an overview of related work specific to change-point detection. In [change point detection algorithm](#) section we provide an overview of change detection algorithms and optimisation approaches. The proposed framework and experimental setup is presented in [Proposed framework](#) section. [Evaluation](#) section presents the evaluation of experimental results of our proposed scheme. Finally, conclusions and future work are presented in [Conclusion and future work](#) section.

Related work

The data collection is an essential part of the activity detection. The online change detection can be used to identify and detect transitions in movement pattern of user activities. The number of approaches have been used in literature for change detection in time series data. However, the varying nature of input data stream for time series data forms significant challenges for various learning algorithms. Hence, the accurate and timely detection of change point in data stream has been examined by many researchers. For instance, the CUSUM (cumulative sum) [6] and GLR (generalized likelihood ratio) [13] approaches has been investigated respectively. In these approaches, the detection of change is monitored in time series data by calculating the logarithm of the likelihood ratio between two consecutive windows. Also, the Bayesian approach [14] has been used for change point detection. In this approach priori information about the underlying distribution of change time is required. At each time instant, the calculated probability corresponds to a change point that has been occurred. The Information-Theoretic [15] approach has been used to detect change in multi-dimensional data stream. This approach is non-parametric technique which does not require assumption of underlying distribution. The relative entropy also called Kullback-Leibler distance has been used to measure the difference between two distributions. Moreover, theory of bootstrapping using statistical methods has been used to identify the statistically significance of calculated measurements. However, more complex method required for K-L distance to increase the change detection performance and power significance of measurement. The Kullback-Leibler Estimation Procedure (KLIEP) [16] has been used to model data distribution based on density ratio estimation. The advantage of this approach is automatic model selection and convergence property. But, the computation cost is significantly increase when large window size is used for change detection. The subspace identification [16] is a non-parametric algorithm that has been used to detect change in time series data. This approach used the subspace to span by the columns of an extended observability matrix which is approximately equal to the one spanned by the sub sequences of time series data. The extended observability matrix column space of subspace method (SSM) has used to estimate the change points in the data. The implicit utilization of generic SSMs has enabled this approach to handle and evaluate rich amount of data more accurately than conventional approaches. In [17], the change detection approach have been used based on fuzzy logic. In the approach, fuzzy entropy has been used to detect the change point in

time series data. The test has been carried out by evaluating change in level and slope of input data and simple regression model have used to test the hypothesis about the detected change points. The approach is efficient in performance but has the limitation that only works for gradual change in the time series data. The One-Class Support Vector Machine algorithm (OCSVM) [18] has been used for change detection in human activities. The high dimensional hyper sphere has been used to model the sensor input data. Moreover, the radi of hyper sphere is used to analyze and evaluate the distribution of change point detection. The increase or decrease in changes correspond various activities. The data is modelled by high dimensional hyper sphere. Change point detection is the distribution based on the analysis of radi of hyper sphere, which changes i.e. increase or decrease correspond to various events. The Auto-associative Neural Networks (AANN) [19, 20] has been used to detect anomaly detection in multivariate time series data. The AANN consist of three layers called input, hidden and output layer. The number of neurons in input and output layers are same while less number of neurons in hidden layer. The AANN is trained using the input layer to encode the data using input layer and forms principle components at bottleneck or hidden layer. Moreover, the principle components are decoded to original data using output layer. The network is trained using the input data and testing data is then applied to detect changes for anomaly detection in time series data. The early detection of anomaly might help in fault diagnostic to take timely action for maintenance. The proposed approach is very effective for anomaly detection but an immediate convergence of AANN required high percentage of normal data for training. Also, the time complexity is quit high and not suitable to be use in real time scenario. Moreover, Deep learning [21, 22] has recently developed as an effective method for activity recognition, making it possible to automate the process of recognising and understanding human behaviours based on data collected from sensors. Researchers have been able to make great strides in this area of study by employing deep neural networks, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs) [23, 24].

The analysis of literature reflects that the current change-point detection methods tend to be more sophisticated in nature. The modelling data distribution in multidimensional data stream is a challenging task, where most of the approaches discussed in [6, 13, 16, 17, 25, 26] have applied only for univariate data. Moreover, prior knowledge is often required about the possible change points and their distribution which could make the implementation of these methods more challenging like an automatic, online

change detection application. Furthermore, the other weaknesses could be the observation of numerous estimation parameters, monitoring descriptors and tuning variables. The alarm increases when analysed multivariate data simultaneously.

Multivariate change point detection algorithm
Multivariate exponentially weighted moving average (MEWMA) change point detection algorithm

The Multivariate Exponentially Weighted Moving Average (MEWMA) is a statistical control method to monitor simultaneously two or more correlated variables and also provide sensitive detection of small, moderate and abrupt shifts in time series data. In the proposed solution, the MEWMA is used to analyze all the covarying time-series data at the same time thus taking into account the inter-relationship among the variables. The MEWMA statistic incorporates information of all prior data including historical and current observation with a user-defined weighted factor [27, 28]. Moreover, MEWMA can be used to detect shift of any size in the process. The multivariate EWMA is an extension of univariate EWMA to multivariate data [28] in order to monitor and analyze the multivariate process. The MEWMA is defined as:

$$Z_i = \Lambda X_i + (1 - \Lambda)Z_{i-1} \tag{1}$$

Where Z_i is the i^{th} MEWMA vector, Λ is the diagonal matrix with elements λ_i for $i= 1..p$ where p is the number of dimensions in input data, and $0 < \lambda_i \leq 1$, and X_i is the i^{th} input vector, $i=1,2,3..n$. The out of control signal is defined using Eq. 2.

$$T_i^2 = Z_i' \Sigma_{Z_i}^{-1} Z_i \tag{2}$$

Where Z_i is the MEWMA vector and Z_i' is its transpose and Σ_{Z_i} is the variance covariance matrix of Z_i . The $h (> 0)$ is chosen to achieve a specified in control signal.

Multivariate cumulative control sum chart (MCUSUM) change point detection algorithm

The cumulative sum control chart is often used when small changes is more important in the data. The multivariate data processing is more complicated than univariate due to multidimensional data processing and evaluation. The Multivariate Cumulative Control Chart (MCUSUM) is a statistical method that use cumulative sum of the input observations to find the smaller and persistent shifts in the process data. The MCUSUM approach proposed by [29] replacing the scalar quantity of univariate cumulative sum into vectors.

$$C_i = (S_{i-1} + X_i - \mu_i)' \Sigma_{S_i}^{-1} (S_{i-1} + X_i - \mu_i) \tag{3}$$

$$S_i = \begin{cases} 0 & \text{if } C_i \leq k \\ (S_{i-1} + X_i - \mu_i)(1 - k/C_i) & \text{if } C_i > k \end{cases} \tag{4}$$

where X_i is the input vector of p -dimensional set of observations for $i= 1..p$ and μ_i is the target vector represents the mean of the input observations while $k (> 0)$ is the reference value and optimal value for k is .5 [29], which is used for tuning a specific shift [30, 31]. The C_i is the generalized length of the CUSUM vector. Initially MSUCUM starts with $S_0 = 0$ and then sequentially calculates the MCUSUM vector. The Σ_{S_i} is the covariance matrix of S_i is the multivariate CUSUM vectors. The out of control signal is calculated using the following Equation

$$Y_i = \sqrt{S_i' \Sigma_{S_i}^{-1} S_i} < h \tag{5}$$

where S_i is the MCUSUM vector and S_i' is its transpose. Σ_{S_i} is the covariance matrix of S_i and $h (> 0)$, is chosen to achieve specified in-control signal.

As discussed earlier that MEWMA and MCUSUM are used to analyze and evaluate multivariate data. Therefore, in multivariate analysis for both algorithms, we consider the data stream of length q consisting of specific data points $X_1, X_2, X_3 \dots X_q$ e.g. $X_1 = (2.362, -9.261, 2.473)$ where the elements represent the x, y and z values of 3-dimensional accelerometer signal. In general, a sequence of data points X_1 to X_q may contain different distributions. In particular, the two subsequence's $X_1, X_2, X_3 \dots X_{i-1}$ and $X_i, X_{i+1} \dots X_q$ may follow different distributions (say for example D1 and D2). The D1 and D2 can be equal or different. In each data stream, MEWMA is used to evaluate the position of change points and calculate the exponentially weighted moving average of multivariate input vectors X_i to provide accurate change point detection. However, MCUSUM is used to calculate the Cumulative sum for each input vector X_i to identify the position and detection of accurate change points in the data stream. The objective of using both algorithms is to determine and identify the position of accurate change points in the data stream. In MEWMA, the Z_i represents the MEWMA vector and is calculated by using the multivariate input vectors as shown in Eq. 1. In addition, the variance - covariance matrix of Z_i is calculated recursively and represented by Σ_{Z_i} to find T_i^2 as shown in Eq. 2. Likewise, in MCUSUM S_i is the MCUSUM vector calculated by using the multivariate input as shown in Eq. 4 and the covariance matrix of S_i is calculated and represented by Σ_{S_i} to find Y_i as shown in Eq. 5.

Furthermore, in our experiments different window sizes (1s,2s,3s) are used to analyze the input data using sliding window with an increment of 1 data point to

perform sequential analysis. The window sizes are used to evaluate the sequence from inside the window. These window sizes are chosen to combine some historical data with new data to balance the data and identify if the change happens. Also, these are reasonable sizes that are taken from experimentation. Moreover, when T_i^2 for MEWMA and Y_i for MCUSUM in Eqs. 2 and 5 are calculated respectively, we consider number of possible values h (0.05, 0.025, 0.01, 0.005) in order to evaluate the confidence of the entire window. The condition is verified, if T_i^2 and Y_i is greater than h , then x_i will be labelled as a change point within the data stream otherwise not. The significance values are used in literature to define regions where the test statistics are unlikely to lie. Evolutionary Algorithms (EAs) is the most significant branch of computational intelligence that have much potential to be used in many application areas. The basic concepts of EAs are inspired by observing the biological structure of nature, for instance, the selection and genetic changes could be used to find the optimal solution for a given optimization problem [32]. The Genetic Algorithm(GA) and Particle Swarm Optimization(PSO) are the heuristics population-based search methods and are widely used for optimization problems.

Genetic algorithm

A Genetic Algorithm is a well-known heuristic search algorithm that mimics the process of evolution. The GA starts with a random sample of variable sets and repeatedly modifies a population of individual solutions. The various criterion can be used for the selection process to obtain evaluation for each individual towards desired solution. The best individual selected as input for the next generation. The GA is used for solving optimization problems based on natural selection, the process that drives biological evolution [33]. The optimization modifies input characteristics of a system using a mathematical process to find the minimum or maximum output. The objective of the fitness function in the GA is used to find the optimal solution to a system.

Particle swarm optimization (PSO)

The Particle Swarm Optimization (PSO) is a population based stochastic optimization technique inspired by social behaviour of bird flocking or fish schooling. The algorithm is initialized with a population of random solutions in order to find and search the optima by updating generations. In PSO, each particle represents a solution and the population of solutions is called swarm of particles. Each particle keep track of its coordinates in problem space which associated the best possible solution achieved so far and is called personal best (pbest). Moreover, another best value is also tracked which is

obtained in the neighbours of the particle and is called local best(lbest). Once each particle takes all the population as its topological neighbours, the best value is a global best and is called gbest. The best position of the particle is selected by calculating the velocity. Once a new position is reached, the best position of each particle and the best position of the swarm are updated as needed. The velocity of each particle is then adjusted based on the experiences of the particle [34]. The PSO have similar functionality with GA, the algorithm is initialized with a population of random solutions in order to find and search the optima by updating generations. However, Unlike GA, PSO has no evolution operators such as crossover and mutation. In PSO, the potential solutions are called particles which fly through the problem space by following the current optimum particles.

Proposed framework

The real dataset has been used for experiments to evaluate MEWMA and MCUSUM approaches for change-point detection using GA and PSO to find the optimal parameter set. As we are evaluating the multivariate data, the x, y and z acceleration magnitude is captured and used as input to MEWMA and MCUSUM approaches. Initially, both approaches were used to analyze and evaluate different parameters including λ (0.1 to 1), the significance values (0.05,0.025,0.01,0.005) and the window sizes (1s,2s,3s) to find accurate change point. Hence, GA and PSO was used to find and identify the optimal set of parameters for MEWMA and MCUSUM. The F-measure metric was used to analyze and evaluate the optimal change point in activity monitoring using GA and PSO. Thus, a detected changes point is considered true, if its index lies in the data stream, $l = z - (f) \dots z + (f)$ where z is the index of manually labelled change point and f is the sampling frequency in Hz. Moreover, the data with detected change points is sent periodically to the edge device which runs the GA and PSO for optimization as shown in Fig. 1. As sensors have limited storage capacity, battery life, and computational power, data processing is done at the edge device. Edge device include computational resources such as processor, memory, and storage. These resources enable them to locally process and analyse data and execute computationally intensive and complex algorithms.

Experimental setup

The dataset was collected form ten healthy participants using 3-axis accelerometer sensor in order to evaluate the change point detection algorithms. The participants consist of five females and five males wearing shimmer sensing platform [35] placed on their chest, right wrist ankle. The data for different activities was collected and captured

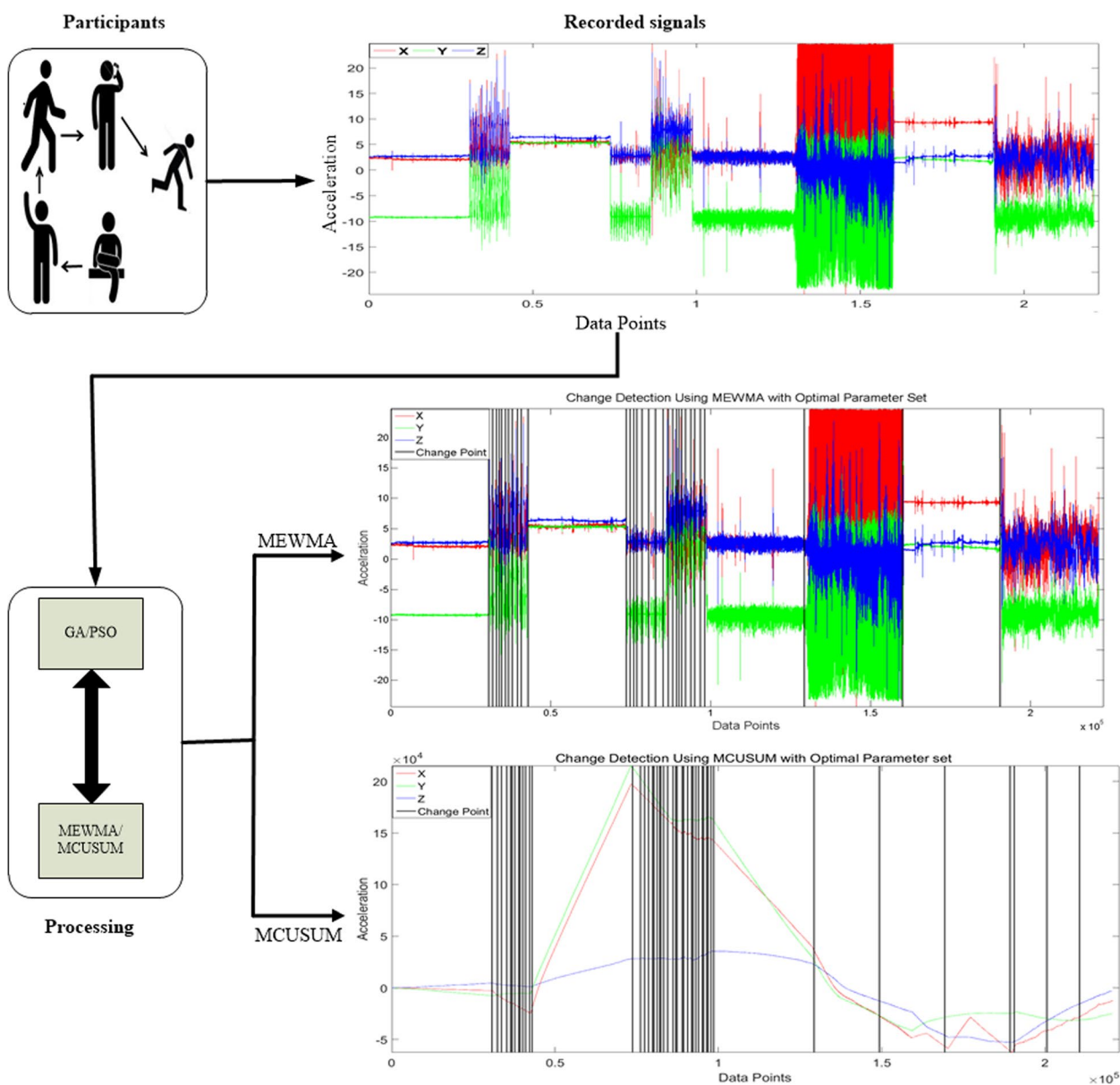


Fig. 1 The system model

with a sample frequency of 102.4 Hz. The nine various activities performed by each participant is presented in Table 1. The different activities were classified as static, transitional and dynamic. In static activities, the participant was asked to remain comfortably still such stand, sit while in transitional activities, the data captures the transition between two activities such as stand to walk, sit to lie. Moreover, the dynamic activities imply that the activity inherently contains meaningful human movements such as walking, running. The change points in the dataset were labelled manually based on the recorded time a participant was to change an activity. For each participant,

the resultant dataset contains a continuous data stream of approximately 35 minutes activities carried out according to the sequence given in Table 1. The 95 labelled transitions recorded for each participant, which in total becomes 950 in total for 10 participants. In the dataset, most of the transitions are from static to dynamic activities and vice versa. However, the dataset also contains transitions form dynamic to dynamic activity like waking to running. After the data collection, the activity execution of accelerometer data was wirelessly streamed to a received computer via Bluetooth communication protocol.

Table 1 Nine various activities performed by each participant

Activity Seq.	Label	Type	Description
1	Standing	Static	Stand for 5 minutes(min)
2	Stand-sit	Transitional	stand for 10 seconds(s),Sit for 10s(15 repetitions)
3	Sleeping	Static	Lie on sofa for 5 min
4	Stand-walk	Transitional	Stand for 10s,Walk for 20s(15 repetitions)
5	Sit-Lie	Transitional	Sit for 10s,Lie for 10s(15 repetitions)
6	Walking	Dynamic	Walk on treadmill at constant speed of 5 min
7	Running	Dynamic	Run on treadmill at constant speed of 5 min
8	Watching TV	Static	Sit on sofa for 5 min
9	Vacuum	Dynamic	Vacuum for 5 min

The purpose of our proposed approach is to find and identify the optimal parameter set for MEWMA and MCUSUM for detecting change points for different high level activities such as stand, sit, sleep, walk run etc. The MEWMA and MCUSUM examples for different activities are shown in Figs. 2 and 3 respectively.

In Fig. 2, the *x*, *y* and *z* axis represent the MEWMA vectors of the input observation of the accelerometer signal while the vertical lines presents the change detection points detected by the MEWMA algorithm.

In Fig. 3, the *x*, *y* and *z* axis represent the MCUSUM vectors of the input observation of the accelerometer signal while the vertical lines presents the change detection points detected by the MCUSUM algorithm.

Parameter optimization using GA and PSO

The objective function of GA and PSO is used to find the optimal solution to a system. In our case, each distinct combination of the three variables provides a single solution in the population, namely λ_i , the window size, and the significance for MEWMA and *k*, the window size, and the significance variable for MCUSUM. Over a number of generations, these solutions “evolve” towards the optimal solution.

Our objective function then tries to find the solution with the maximum F-measure value given a range of input values for both algorithms. The F-measure is used as the measure to find the overall effectiveness of the activity recognition or change detection by combining

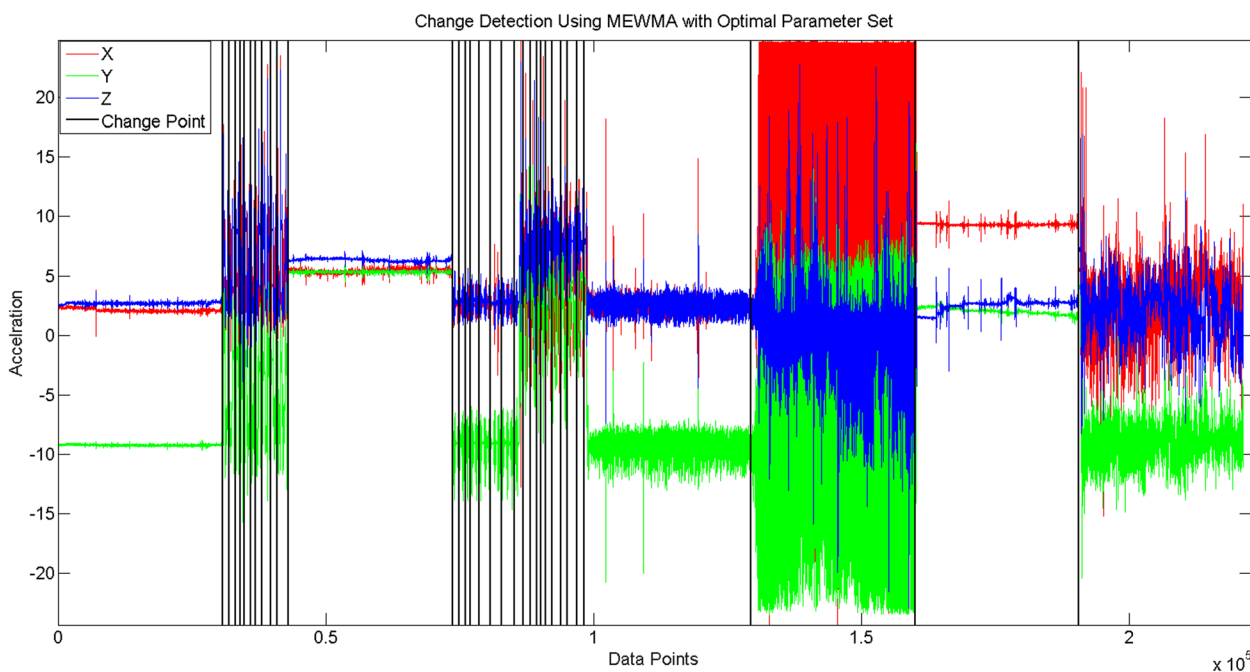


Fig. 2 Real dataset example of change point detection using MEWMA for different activities

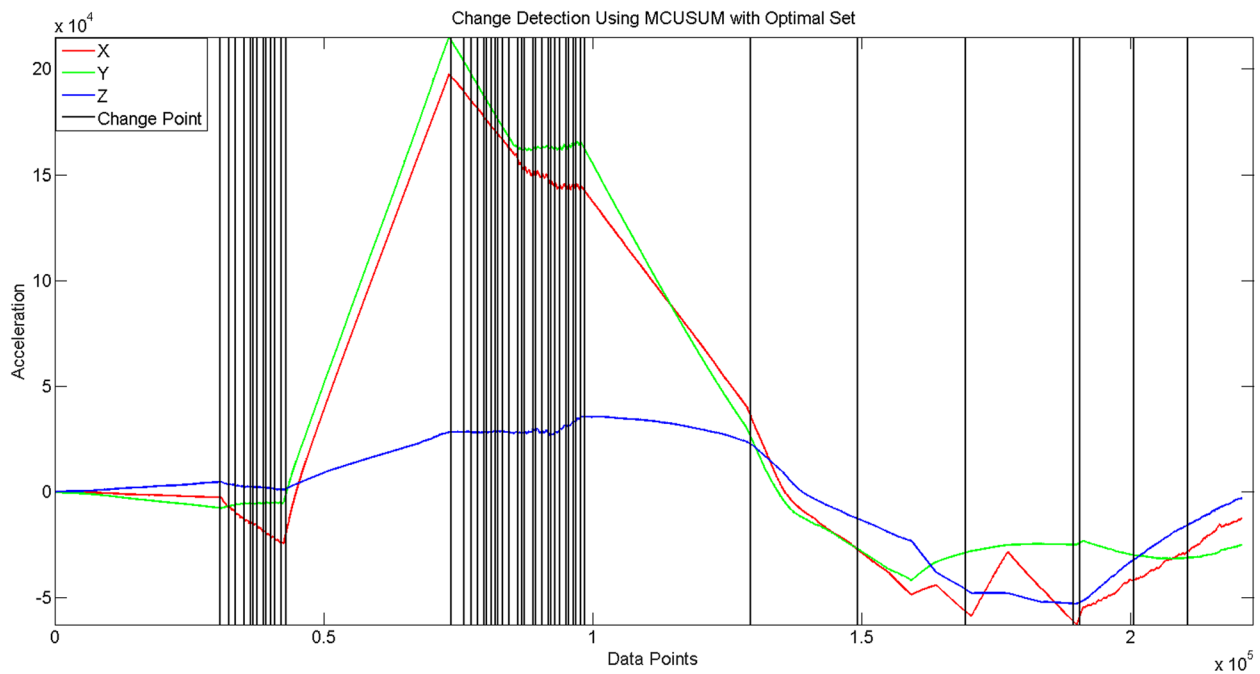


Fig. 3 Real dataset example of change point detection using MCUSUM for different activities

the precision and recall. The objective function for GA and PSO using MEWMA and MCUSUM can be defined as follows in Eqs. 6 and 7 respectively.

$$F - measure_{max} = \max_{(\lambda_i, win_size, sig_value)} (F - measure_{MEWMA}) \quad (6)$$

$$F - measure_{max} = \max_{(k, win_size, sig_value)} (F - measure_{MCUSUM}) \quad (7)$$

Both algorithms MEWMA and MCUSUM use the three variables as input where window size ranges from 1s, 2s and 3s and significance values of 0.05, 0.01, 0.025, 0.005 are same for both algorithms. However, MEWMA used λ_i ranges from 0.1 to 1 and MCUSUM used $k=0.5$ as a standard value presented in [29] as shown in Eqs. 6 and 7 respectively.

The objective function defined in Eqs. 6 and 7 are initialized by upper and lower bounds of the three parameters to find the maximum F-measure with the

optimal parameter set. The Matlab 2015b global optimization tool box (Matlab-Toolbox,2015) was used for experiments and the GA and PSO parameters are set according to our experimental setup as shown in Tables 2 and 3. For GA, our proposed model uses Eq. 6 as the fitness function by initializing upper and lower bounds of the three parameters to find the maximum F-measure with the optimal parameter set. After the exploration with different parameter settings, the optimal GA parameters, which maximize the fitness function of the F-measure, are shown in Table 2. The selection function in the GA chooses the parents for the next generation based on their scale values by evaluating the fitness function. As we need to find the maximum value of the fitness function using Eq. 6, the individual with the maximum value of the fitness function has greater chance for reproduction and also for generation of offspring. Here we used a stochastic uniform distribution to build in randomness. The

Table 2 Genetic algorithm (GA) parameters

Parameters	GA
Population Size	50
Selection	Stochastic uniform
Reproduction	0.8
Crossover	Scattered
Mutation	Adaptive feasible
Generations	100

Table 3 Particle Swarm optimization (PSO) Parameters

Parameters	PSO
Swarm Size	50
Initial Swarm	<i>pswcreationuniform</i>
HybridFcn	<i>fmincon</i>
Max Iterations	100

reproduction function helps to determine how the GA creates children at each new generation. Elite count or the crossover fraction can be used to create new children at each generation. The first method specifies the number of individuals that are guaranteed to survive in next generation. However, the later method specifies the fraction of the next generation which crossover produces; we here use reproduction probability 0.8 and mutation with probability 0.2 so as to allow some new values to take part in the optimization process.

The crossover combines two individuals or parents to form a new individual or child for the next generation. Different methods such as constraint dependent, scattered, heuristic, and arithmetic approaches can be used depending on the problem requirement. We choose the scatter method to make random selection. In the population, the mutation function makes small random changes in the individuals, which provide genetic diversity and enable the GA to search in a broader space. Different methods can be used for this, such as the Gaussian function, uniform function, and adaptive feasible function for random modification. We choose an adaptive feasible solution because it randomly generates directions that are adaptable with respect to the last successful generation.

The GA process, illustrated in Fig. 4 with respect to the GA parameters proposed in Table 2, is described as follows [36]:

- Initialize the population size is with the number 50, which specifies how many individuals there are in each of the iterations. Usually, the number 50 is used for a problem with five or fewer variables, and the number of 200 is used otherwise.
- Check the termination condition of the algorithm to determine if the number of generations has exceeded the maximum value. If so, the GA algorithm is terminated, otherwise, continue with the following steps.
- Calculate the maximum value of the fitness function using Eq. 6.
- The individuals are selected from the current population applying a stochastic uniform function. Each parent corresponds to a section proportional to its expectation. The algorithm moves along in steps of equal size. At each step, a parent is allocated from the section uniformly.
- The individuals are then reproduced randomly with a fraction using the crossover operation. The scatter function is used to select the genes where the vector is 1 from the first parent and 0 from the second parent before combining them to form a child.
- Mutation is then applied with the adaptive feasible method to randomly generate individuals in the population.
- Finally, a new generation is updated and the GA algorithm loops back to check the termination condition.

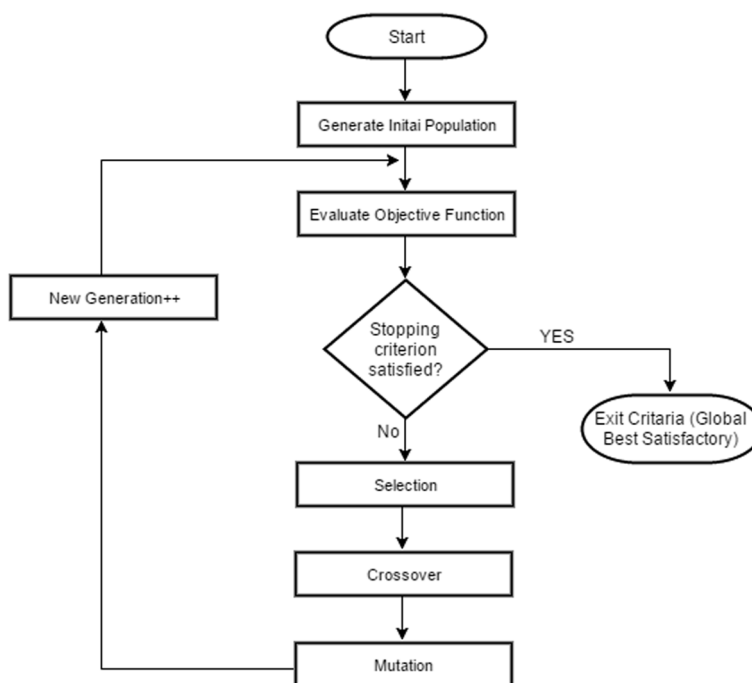


Fig. 4 Flow chart of various stages to perform genetic algorithm (GA) optimization

The default value for the generations is 100 multiplied by the number of variables used, but we choose the best value for generation by experimentation with different values

The GA parameters are presented in Table 2.

Likewise, for PSO, our proposed model uses Eq. 7 as the fitness function by initializing upper and lower bounds of the three parameters to find the maximum F-measure with the optimal parameter set. After the exploration with different parameter settings, the optimal PSO parameters, which maximize the fitness function of the F-measure, is presented in Table 3.

Initially, the PSO creates particles at random with uniform distribution using *pswcreationuniform* function within the defined lower bound and upper bound given in Eq. 7. The Hybrid function is used to perform constrained or unconstrained minimization or maximization. In our experiments, we used *fmincon* function which provide constrained maximization for our objective function. The rest of the options *MaxStallIterations*, *MaxStallTime*, *ObjectiveLimit* etc are kept Matlab default for PSO, detailed information can be found in [37].

The PSO process, illustrated in Fig. 5 with respect to the PSO parameters proposed in Table 3, is described as follows [38].

- Initialize the population size with the number 50, which specifies how many individuals there are in each of the iterations. Usually, the number 50 is used for a problem with five or fewer variables, and the number of 200 is used otherwise.
- Initialize swarm and each particle randomly with initial position and velocity with the search space.
- Calculate the maximum value of the objective function using Eq. 7.
- Initially, the first objective values and positions are inevitably considered as personal best values and personal best positions. Further, the global best value and position are chosen based on the best fitness value among all particles and that the particle value and position are selected as global best value and position in the whole swarm population.
- If the stopping criteria becomes false, then the velocity and position of the particles are updated.
- Finally, a new generation is updated and the PSO algorithm loops back to calculate the fitness value and updated position for each particle. The updated personal best value and position is compared to the previous personal best value and position. If the new fitness value is better than previous one then the per-

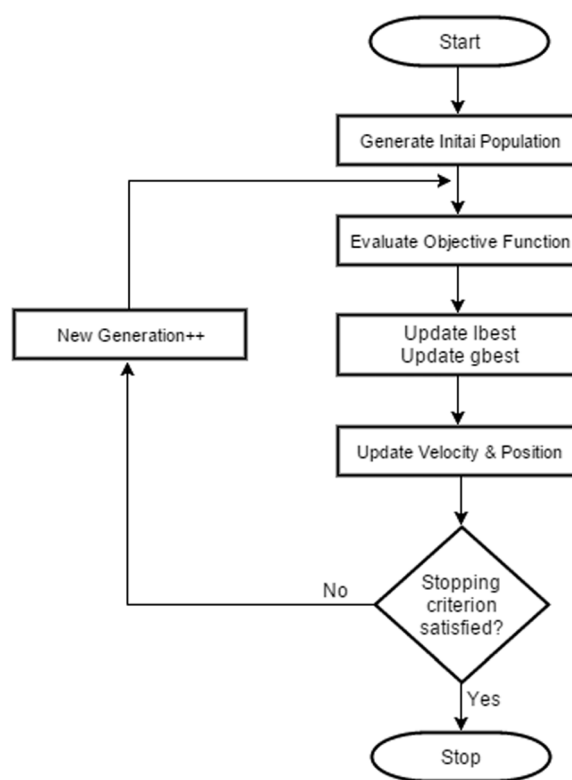


Fig. 5 Flow chart of various stages to perform Particle Swarm optimization (PSO) optimization

sonal best value and position are updated. The same process is carried out for updating the global best value and position.

- The process is continued till the termination condition is satisfied. The default value for the generations is 100 multiplied by the number of variables used, but we choose the best value for generation by experimentation with different values.

The PSO parameters are presented in Table 3.

Evaluation

The positive and negative detection cases were defined as, the true positive (TP) which is the correctly identified change point and true negative (TN) which are the non-transitional points which are not labelled as change. The false positive (FP) is the non-transitional point which the algorithm highlights as a change and false negative (FN) occurs when the algorithm is unable to detect changes in the user’s activity. The accuracy, precision, sensitivity and G-means and F-measure metrics were used for evaluation of optimal parameter selection for MEWMA and MCUSUM algorithm. The GA and PSO are used for optimal parameters selection for both change detection

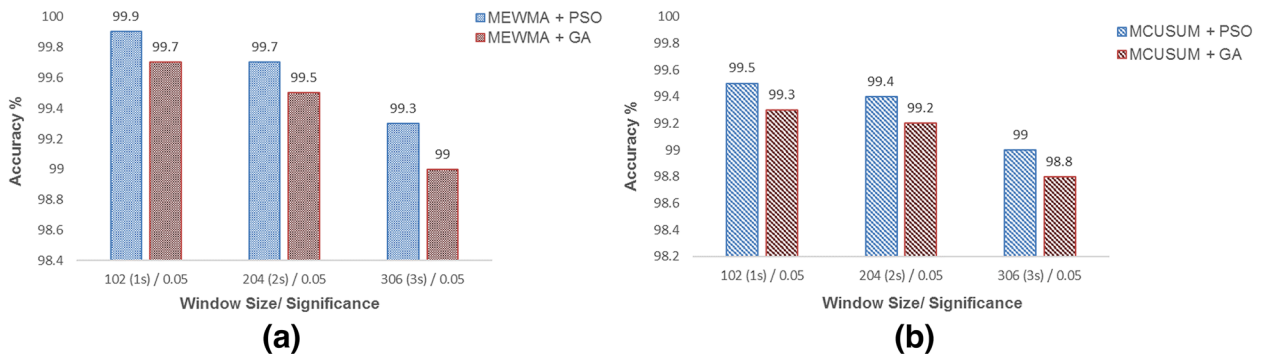


Fig. 6 a Comparison of accuracy between MEWMA (PSO and GA). b Comparison of accuracy between MCUSUM (PSO and GA)

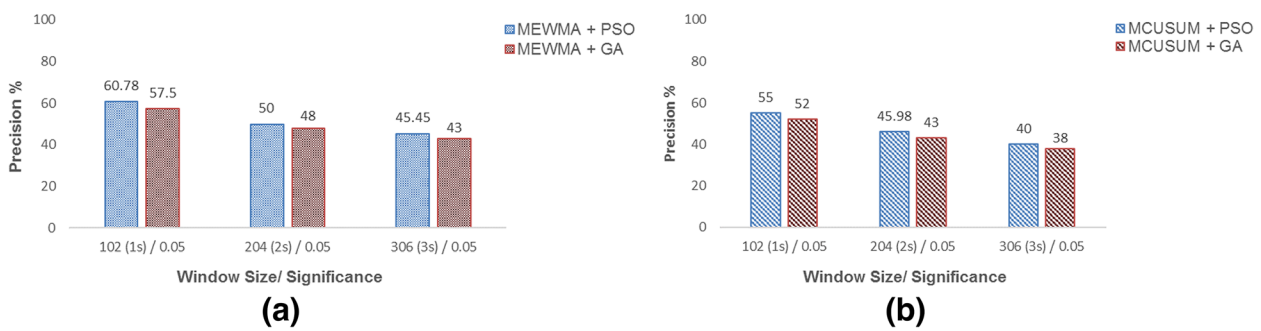


Fig. 7 a Comparison of precision between MEWMA (PSO and GA). b Comparison of precision between MCUSUM (PSO and GA)

algorithms. The F-measure metrics were used as an objective function to analyze and evaluate the optimal change point in activity monitoring using GA and PSO. Moreover, the evaluation metrics such as accuracy, precision, sensitivity, G-Means and F-measure were used for evaluation. In Figs. 6, 7, 8, 9, 10 and 11 results are shown based on experiments for evaluation of optimal parameter selection for MEWMA and MCUSUM algorithm using PSO and GA.

The overall accuracy can be defined as $TP + TN / (TP + TN + FP + FN)$ and the MEWMA with

PSO achieved highest accuracy's of 99.9%, 99.7% and 99.3% for window sizes (1s, 2s and 3s), λ (0.5, 0.6 & 0.7) and $p=0.05$ for the optimal parameter set for 9 different activities. Correspondingly, the MEWMA with GA achieved highest accuracies of 99.7%, 99.5% and 99% for window sizes (1s, 2s & 3s), λ (0.5, 0.6 & 0.7) and $p=0.05$ for the optimal parameter set of 9 different activities as shown in Fig. 6 (a).

The MCUSUM with PSO achieved highest accuracies 99.5%, 99.4% and 99% for window size (1s, 2s & 3s), $k=0.5$ and $p=0.05$ for the optimal parameter set of 9 different activities. Correspondingly, the MCUSUM with

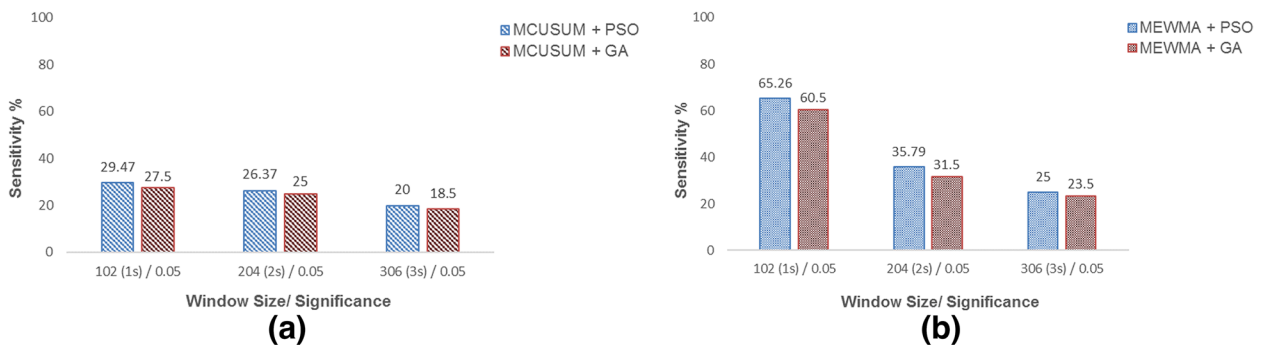


Fig. 8 a Comparison of sensitivity between MEWMA (PSO and GA). b Comparison of sensitivity between MCUSUM (PSO and GA)

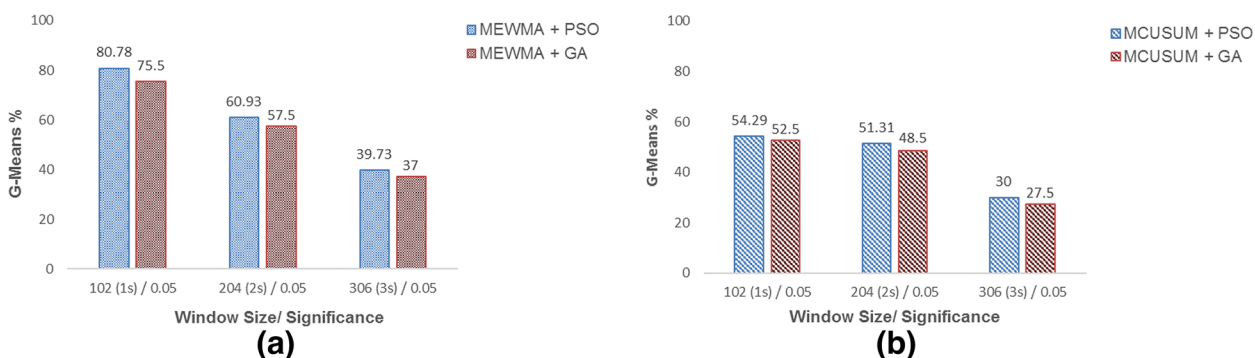


Fig. 9 a Comparison of G-means between MEWMA (PSO and GA). b Comparison of G-means between MCUSUM (PSO and GA)

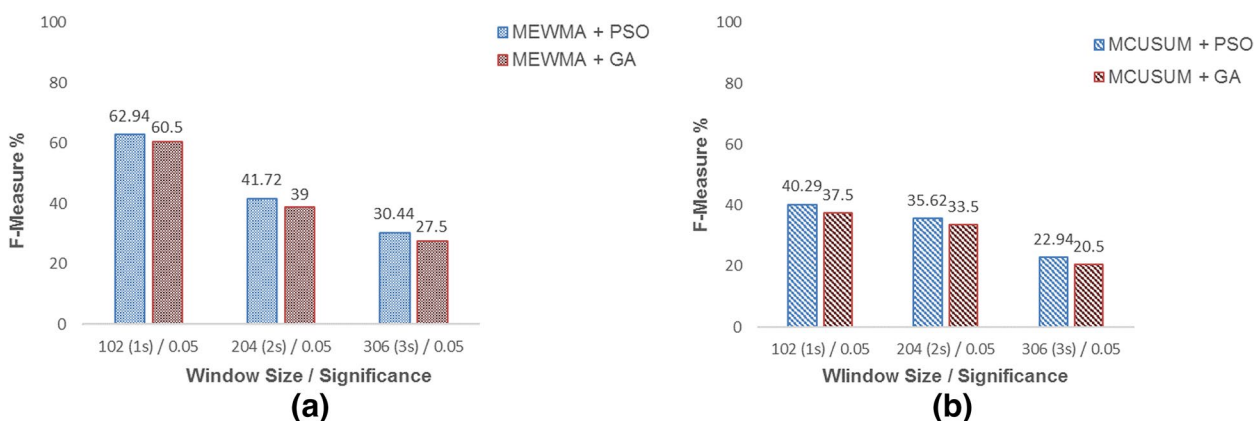


Fig. 10 a Comparison of F-measure between MEWMA (PSO and GA). b Comparison of F-measure between MCUSUM (PSO and GA)

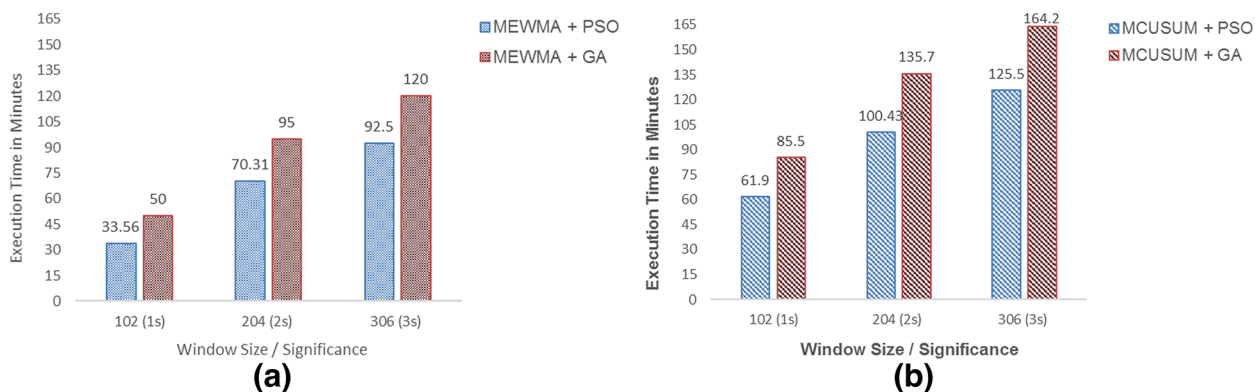


Fig. 11 a Comparison of computational cost between MEWMA (PSO and GA) between MEWMA (PSO and GA). b Comparison of computational cost between MEWMA (PSO and GA)

GA achieved highest accuracies of 99.3%, 99.2% and 98.8% for window size (1s ,2s & 3s), $k=0.5$ and $p=0.05$ for the optimal parameter set of 9 different activities as shown in Fig. 6 (b). The accuracy is relatively high for both MEWMA (PSO & GA) and MCUSUM (PSO

& GA) because of the relatively high disproportionate number of TNs in the data. The reason is the class imbalance problem [39] in our dataset. The MEWMA and MCUSUM with PSO achieved highest accuracy as compared to MEWMA and MCUSUM with GA. A

one-sided t -test is performed to find the statistical significance for the accuracy metric for 10 experiments repeatedly performed for each approach as MEWMA with PSO and MCUSUM with PSO. The results of the one-sided t -test evaluate that the MEWMA with PSO is statistically significant by achieving the significance 0.0207 which is less than the standard p -value=0.05. Therefore, MEWMA with PSO outperformed than MCUSUM with PSO by achieving higher accuracy for accurate change point detection.

The precision defined as $TP/TP + FP$ and the maximum precisions attained for MEWMA with PSO are 60.78%, 50% and 45.45% while for MEWMA with GA are 57.50%,48% and 43% for the optimal set of parameters using the same window sizes, lambda values and significance value as discussed earlier. The precision of MEWMA (PSO & GA) is represented in Fig. 7 (a).

Likewise, the MCUSUM (PSO & GA) has achieved maximum precision of about 55%, 45.98%, 40% while for MCUSUM with GA is about 52%,43% ,38% for the same window sizes, $k=0.5$ and significance values for the optimal set of parameters as discussed earlier. The precision of MCUSUM with PSO and GA is represented in Fig. 7 (b). The higher precision is achieved for MEWMA (PSO & GA) than MCUSUM (PSO & GA) as shown in Fig. 7 (a) and (b). However, the MEWMA with PSO improved than MCUSUM with PSO approximately 5.60% for each window size for accurate change point detection using optimal parameter set. The reason for low precision is due to the high number of occurrences of false alarms as our algorithm is very sensitive and detects possible change points even if they are small. A one-sided t -test is performed to find the statistical significance for the precision metric for 10 experiments repeatedly performed for each approach of MEWMA with PSO and MCUSUM with PSO. The results of the t -test evaluate that the MEWMA with PSO is statistically significant by achieving the significance 0.0388 which is less than the standard p -value.

Moreover, the sensitivity (also known as Recall) is defined $TP/TP + FN$ and the maximum sensitivity values achieved by MEWMA with PSO are 65.26%, 35.79% and 25% while 60.5%, 31.50% and 23.50% for MEWMA with GA using the same optimal parameter set with window sizes (1s,2s,3s), λ (0.5, 0.6 & 0.7) and $p=0.05$ as shown in Fig. 8 (a). The MEWMA with PSO has approximately 4.5% higher sensitive value on average for each window size than MEWMA with GA. Likewise, the highest sensitivity was achieved for MCUSUM with PSO is about 29.47%, 26.37% and 20% while 27.5%, 25% and 18.50% for MCUSUM with GA using optimal parameter set with window sizes (1s,2s,3s), $k=0.5$ and $p=0.05$ as shown in Fig. 8 (b). The MCUSUM

with PSO is improved approximately 1.5% on average for each window size than MCUSUM with GA.

However, the analysis of MEWMA with PSO results in about 35.79%, 9.42% and 5% higher sensitivity values in each window size respectively as compared to MCUSUM(PSO). Also, MEWMA(GA) is improved about 33%, 6.5% and 5% for each window size respectively compared to MCUSUM(GA) as shown in Fig. 8 (a) and (b). A one-sided t -test is performed to find the statistical significance for the sensitivity metric for 10 experiments repeatedly performed for each approach i.e. MEWMA with PSO and MCUSUM with PSO. The results of the t -test evaluate that the MEWMA with PSO is highly statistically significant by achieving the significance 0.0069 which is less than the standard p -value.

The G-means and F-measure can be defined using Eqs. 8 and 9 Respectively.

$$\text{G-means} = \sqrt{\frac{TP}{TP + FN} \times \frac{TN}{TN + FP}} \quad (8)$$

$$\text{F-measure} = \frac{2 \times \text{Recall} \times \text{Precision}}{(\text{Recall} + \text{Precision})} \quad (9)$$

The MEWMA with PSO achieved highest G-means is about 80.78%, 60.93% and 39.73% for window size (1s ,2s and 3s), λ (0.5, 0.6 & 0.7) and $p=0.05$ for the optimal parameter set for 9 different activities. On the other hand, the MEWMA with GA achieved highest G-means is about 75.5%, 57.5% and 37% for window size (1s ,2s and 3s), λ (0.5, 0.6 & 0.7) and $p=0.05$ for the optimal parameter set of 9 different activities as shown in Fig. 9 (a). The MEWMA with PSO is improved approximately 3% on average for each window size than MEWMA with GA.

Similarly, the MCUSUM with PSO achieved highest G-means for about 54.29%, 51.31% and 30% for window size (1s ,2s & 3s), $k=0.5$ and $p=0.05$ for the optimal parameter set of 9 different activities. On the other hand, the MCUSUM with GA achieved highest accuracy is about 52.5%, 48.5% and 27.5% for window size (1s ,2s and 3s), $k=0.5$ and $p=0.05$ for the optimal parameter set of 9 different activities as shown in Fig. 9 (b). The MCUSUM with PSO is improved approximately 2.5% on average for each window size compared with MCUSUM with GA.

However, the G-mean analysis of MEWMA (PSO & GA) and MCUSUM (PSO & GA) was improved compared with MEWMA (PSO) with about 26.5%, 9.5% and 9.7% for each window size respectively as compared to MCUSUM(PSO). Also, MEWMA(GA) is improved about 23%, 9% and 9.5% for each window size respectively compared to MCUSUM(GA) as shown in Fig. 9 (a) and (b). A one-sided t -test is performed to find the statistical significance for the accuracy metric for 10 experiments

repeatedly performed for each approach i.e. MEWMA with PSO and MCUSUM with PSO. The results of the t -test evaluate that the MEWMA with PSO is statistically significant by achieving the significance 0.0431 which is less than the standard p -value.

Likewise, The maximum F-Measure was achieved for MEWMA with PSO for about 62.94%, 41.72% and 30.44% compared with 60.5%, 39% and 27.5% for MEWMA with GA using optimal parameter set with window sizes (1s,2s,3s), λ (0.5, 0.6 & 0.7) and $p=0.05$ as shown in Fig. 10 (a). The MEWMA with PSO is improved approximately 2.7% on average for each window size than MEWMA with GA.

Likewise, the highest F-Measure that was achieved for MCUSUM with PSO is about 40.29%, 35.62% and 22.94% while 37.5%, 33.5% and 20.5% for MCUSUM with GA using optimal parameter set with window sizes (1s,2s,3s), $k=0.5$ and $p=0.05$ as shown in Fig. 10 (b). The MCUSUM with PSO is improved approximately 2.4% on average for each window size than MCUSUM with GA. A one-sided t -test is performed to find the statistical significance for the accuracy metric for 10 experiments repeatedly performed for each approach of MEWMA with PSO and MCUSUM with PSO. The results of the t -test suggest that the MEWMA with PSO is statistically significant by achieving the significance 0.0246 which is less than the standard p -value.

However, the F-Measure analysis of MEWMA (PSO & GA) and MCUSUM (PSO & GA) was improved with MEWMA (PSO) by about 22.65%, 6.1% and 7.5% for each window size respectively as compared to MCUSUM(PSO). Also, MEWMA(GA) is improved about 23%, 6.5% and 7% for each window size respectively compared to MCUSUM(GA) as shown in Fig. 10 (a) and (b) respectively.

Computational cost

The empirical analysis of computational cost of both algorithm MEWMA (PSO & GA) and MCUSUM (PSO & GA) for accurate change detection using optimal parameter selection. The techniques are implemented in Matlab 2015b and experiments are performed on a system with processor 3.40 GHz and 8GB RAM. The Matlab *tic toc* function is used to calculate the time for optimal parameter set with accurate change and high metric measures.

The results in Fig. 11 (a) presents that MEWMA (PSO) took less time at about 17.56 min, 24.69 min and 27.5 min respectively for each window size compared with MEWMA (GA) for optimal solution of accurate change detection. Likewise, MCUSUM (PSO) also took less time about 23.6 min, 35.27 min and 40.20 min respectively for each window size toward optimal solution of accurate change detection as shown in Fig. 11 (b).

Furthermore, the MEWMA (PSO) outperforms than MCUSUM (PSO) in achieving low computational cost of about 28.34 min, 30.12 min and 33 min for the same window sizes towards optimal solution. Similarly, the MEWMA (GA) also performed better than MCUSUM(GA) by using minimal computational cost of about 35.5 min, 40.70 min and 44.20 min for the same window sizes towards optimal solution. Also, the t -test results justify computational efficiency of MEWMA with PSO over MCUSUM with PSO by proving statistical significance with 95% confidence achieved after 10 repeated experiments were investigated. The PSO and GA are both population based algorithm, however, PSO is a relatively recent heuristic search algorithm compared with GA. PSO is computationally efficient because it uses less number of functions than GA for evaluation towards optimal solutions [40]. Hence, as we are more inclined towards online activity monitoring which require lightweight algorithm for evaluation of data. Therefore, the analysis of current results reflects that the MEWMA with PSO is a good choice for online implementation for accurate change point detection.

Conclusion and future work

The multivariate approaches are used to analyze and evaluate multivariate data for automatic change point detection. In multivariate data analysis, more than one characteristics of a system evaluated simultaneously and the approach also identify the relationship among these characteristics. The proposed MEWMA approach tunes the different parameters such as lambda, which weights the current versus historical data, window size and significance value with the aim of achieving better performance and accurate change point detection. Also, we implement MCUSUM a multivariate approach to use as a bench mark for our proposed technique. Moreover, the GA and PSO are used to automatically identify an optimal parameter set using different parameters for MEWMA and MCUSUM, so as to maximize the objective function i.e. the F-measure. The evaluation is performed using different metric measures and the experimental results show that the proposed scheme outperforms than the bench mark scheme. Also, the computation cost is less than the benchmark approach. Moreover, t-tests were also performed for each evaluation metric and the results show that the proposed approach is statistically significantly better than the benchmark technique.

The limitation of this work is that we are using the same lambda value across all variants, however, there is a possibility of using a set of lambda values simultaneously (one for each variant) that could be referred as fully Multivariate approach, and which will be

addressed in our future work. A key part of the future work will focus on class imbalance problem to explore and investigate the different online class imbalance learning approaches that can be used to balance the minority class in the dataset and possibly improve the classification results. In addition, accelerometer placement will be explore for different locations to obtain more and detailed information related to each activity which can be used for analysis of change detection in different user activities.

Acknowledgements

The authors would like to thank the School of Computing, Ulster University for supporting this work.

Authors' contributions

N.K wrote the paper. S.M, S.Z, and C.N reviewed the paper.

Funding

This work received no specific grant from any funding agency.

Availability of data and materials

The datasets used during the current study are available from the corresponding author on reasonable request.

Declarations

Competing interests

The authors declare no competing interests.

Received: 20 January 2023 Accepted: 1 June 2023

Published online: 19 June 2023

References

- Sarkar P, Meeker WQ (1998) A bayesian on-line change detection algorithm with process monitoring applications. *Qual Eng* 10(3):539–549
- Pikoulis E, Psarakis EZ (2015) Automatic seismic signal detection via record segmentation. *IEEE Trans Geosci Remote Sens* 53(7):3870–2884
- Clifton DA, Wong D, Clifton L, Wilson S, Way R, Pullinger R, Tarassenko L (2013) A large-scale clinical validation of an integrated monitoring system in the emergency department. *IEEE J Biomed Health Inform* 17(4):835–842
- Ohmura R, Takasaki W (2011) Response time improvement in accelerometer-based activity recognition by activity change detection. In: 13th international conference on Ubiquitous computing. ACM pp 589–590
- Cleland I, Han M, Nugent C, Lee H, McClean S, Zhang S, Lee S (2014) Evaluation of prompted annotation of activity data recorded from a smart phone. *Sensors* 14(9):15861–15879
- Basseville M, Nikiforov IV (1993) *Detection of abrupt changes: theory and application* (Vol. 104). Englewood Cliffs: prentice Hall
- Patterson T, Khan N, McClean S, Nugent C, Zhang S, Cleland I, Ni Q (2016) Sensor-based change detection for timely solicitation of user engagement. *IEEE Trans Mob Comput PP(99):1–1*. <https://doi.org/10.1109/TMC.2016.2640959>
- Ni Q, Patterson T, Cleland I, Nugent C (2016) Dynamic detection of window starting positions and its implementation within an activity recognition framework. *J Biomed Inform* 62:171–180
- Kanhere SS (2013) Participatory sensing: Crowdsourcing data from mobile smartphones in urban spaces. In: *International Conference on Distributed Computing and Internet Technology*. Springer pp 19–26
- Stikic M, Larlus D, Ebert S, Schiele B (2011) Weakly supervised recognition of daily life activities with wearable sensors. *IEEE Trans Pattern Anal Mach Intell* 33(12):2521–2537
- Khan N, McClean S, Zhang S, Nugent C (2016) Optimal parameter exploration for online change-point detection in activity monitoring using genetic algorithms. *Sensors* 16(11):1784
- Patterson T, Khan N, McClean S, Nugent C, Zhang S, Cleland I, Ni Q (2016) Sensor-based change detection for timely solicitation of user engagement. *IEEE Transactions on Mobile Computing* 16(10):2889–2900.
- Gustafsson F (1996) The marginalized likelihood ratio test for detecting abrupt changes. *IEEE Trans Autom Control* 41(1):66–78
- MohammadDjafari A, Féron O (2006) Bayesian approach to change points detection in time series. *Int J Imaging Syst Technol* 16(5):215–221
- Dasu T, Krishnan S, Venkatasubramanian S, Yi K (2006) An information-theoretic approach to detecting changes in multi-dimensional data streams. In: *In Proc. Symp. on the Interface of Statistics, Computing Science, and Applications*. Citeseer
- Kawahara Y, Yairi T, Machida K (2007) Change-point detection in time-series data based on subspace identification. In: *Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on*. IEEE pp 559–564
- Kumar K, Wu B (2001) Detection of change points in time series analysis with fuzzy statistics. *Int J Syst Sci* 32(9):1185–1192
- Vlasveld RQ (2014) *Temporal segmentation using support vector machines in the context of human activity recognition* (Doctoral dissertation, Universiteit Utrecht)
- Hu X, Qiu H, Iyer N (2007) Multivariate change detection for time series data in aircraft engine fault diagnostics. In: *Systems, Man and Cybernetics, 2007. ISIC. IEEE International Conference on*. IEEE pp 2484–2489
- Xu C, Chai D, He J, Zhang X, Duan S (2019) Innohar: A deep neural network for complex human activity recognition. *IEEE Access* 7:9893–9902
- Gu F, Chung MH, Chignell M, Valaee S, Zhou B, Liu X (2021) A survey on deep learning for human activity recognition. *ACM Comput Surv (CSUR)* 54(8):1–34
- Cheng X, Zhang L, Tang Y, Liu Y, Wu H, He J (2022) Real-time human activity recognition using conditionally parametrized convolutions on mobile and wearable devices. *IEEE Sensors J* 22(6):5889–5901
- Mutegeki R, Han DS (2020) A cnn-1stm approach to human activity recognition. In: 2020 international conference on artificial intelligence in information and communication (ICAIC). IEEE, pp 362–366
- Dua N, Singh SN, Semwal VB, Challa SK (2023) Inception inspired cnn-gru hybrid network for human activity recognition. *Multimedia Tools Appl* 82(4):5369–5403
- Kifer D, Ben-David S, Gehrke J (2004) Detecting change in data streams. In: *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30. VLDB Endowment* pp 180–191
- Bifet A, Gavalda R (2007) Learning from time-changing data with adaptive windowing. In: *Proceedings of the 2007 SIAM International Conference on Data Mining*. SIAM pp 443–448
- Khoo MB (2004) An extension for the univariate exponentially weighted moving average control chart. *Matematika* 20:43–48
- Pan X, Jarrett JE (2014) The multivariate ewma model and health care monitoring
- Crosier RB (1988) Multivariate generalizations of cumulative sum quality-control schemes. *Technometrics* 30(3):291–303
- Hongcheng L (2007) *Multivariate extensions of cusum procedure*. Thesis
- hamed MS, Mansour MM, Enayat M. Abd E (2016) Mcusum control chart procedure-monitoring the process mean with application. *J Stat Adv Theory Appl* 16(1):105–132
- Holzinger K, Palade V, Rabadan R, Holzinger A (2014) Darwin or lamarck? future challenges in evolutionary algorithms for knowledge discovery and data mining. *Interactive Knowledge Discovery and Data Mining in Biomedical Informatics: State-of-the-Art and Future Challenges* 35–56
- Malhotra R, Singh N, Singh Y (2011) Genetic algorithms: Concepts, design for optimization of process controllers. *Comput Inf Sci* 4(2):39
- Hu X, Shi Y, Eberhart R (2004) Recent advances in particle swarm. In *Proceedings of the 2004 congress on evolutionary computation (IEEE Cat. No. 04TH8753)*. IEEE 1:90–7
- Burns A, Greene BR, McGrath MJ, O'Shea TJ, Kuris B, Ayer SM, Stroiescu F, Cionca V (2010) Shimmer™-a wireless sensor platform for noninvasive biomedical research. *IEEE Sensors J* 10(9):1527–1534

36. McCall J (2005) Genetic algorithms for modelling and optimisation. *J Comput Appl Math* 184(1):205–222
37. Matlab-Toolbox MT (2015) Particle swarm optimization (global optimization toolbox). <https://uk.mathworks.com/help/gads/particle-swarm.html>. Accessed 1 Jan 2023
38. Chavan S, Adgokar NP (2015) An overview on particle swarm optimization: basic concepts and modified variants. *Int J Sci Res* 4(5):255–260
39. Galar M, Fernandez A, Barrenechea E, Bustince H, Herrera F (2012) A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. *IEEE Trans Syst Man Cybern Part C (Appl Rev)* 42(4):463–484
40. Hassan R, Cohanim B, De Weck O, Venter G (2005) A comparison of particle swarm optimization and the genetic algorithm. In 46th AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference (p. 1897)

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

Submit your next manuscript at ▶ [springeropen.com](https://www.springeropen.com)
