Flexible Multi-Objective Particle Swarm Optimization Clustering with Game Theory to Address Human Activity Recognition Fully Unsupervised

Parham Hadikhani $^{1,1,1,1}$, Daphne Teck Ching Lai $^2$, Wee-Hong Ong $^2$, Mohammad H.Nadimi-Shahraki $^2$, and Seyedali Mirjalili $^2$

$^1$Universiti Brunei Darussalam
$^2$Affiliation not available

November 2, 2023

Abstract

Most research in human activity recognition is supervised, while non-supervised approaches are not completely unsupervised. In this paper, we provide a novel flexible multi-objective particle swarm optimization (PSO) clustering method based on game theory (FMOPG) to discover human activities fully unsupervised. Unlike conventional clustering methods that estimate the number of clusters and are very time-consuming and inaccurate, an incremental technique is introduced which makes the proposed method flexible in dealing with the number of clusters. Using this technique, clusters that have a better connectedness and good separation from other clusters are gradually selected. To improve the convergence speed of PSO in achieving the best solution and dealing with spherical shape clusters, updating of particles’ velocity is modified using the concept of mean-shift vector. To solve multi-objective optimization problems, Nash equilibrium in game theory is used to select the optimal solution on the pareto front. Gaussian mutation is also employed on the pareto front to generate diverse solutions and create a balance between exploitation and exploration. The proposed method is compared with state-of-the-art methods on five challenging datasets. FMOPG has improved clustering accuracy by 3.65% compared to automated methods. Moreover, the incremental technique has improved the clustering time by 71.18%.
Flexible Multi-Objective Particle Swarm Optimization Clustering with Game Theory to Address Human Activity Discovery Fully Unsupervised

Parham Hadikhani*, Daphne Teck Ching Lai and Wee-Hong Ong

Abstract—Most research in human activity recognition is supervised, while non-supervised approaches are not completely unsupervised. Moreover, these methods cannot be used in real-time applications due to high calculations. In this paper, we provide a novel flexible multi-objective particle swarm optimization clustering method based on game theory (FMOPG) to discover human activities fully unsupervised. Unlike conventional clustering methods that estimate the number of clusters and are very time-consuming and inaccurate, an incremental technique is introduced which makes the proposed method flexible in dealing with the number of clusters and improves the speed of clustering. By adopting this technique, clusters with a better connectedness and good separation from other clusters are gradually selected. Updating of particles' velocity is modified by adopting the concept of mean-shift vector to improve the convergence speed of PSO in achieving the best solution and dealing with non-spherical shape clusters. Multi-objective optimization problem is mapped to game theory by adopting Nash equilibrium to select the optimal solution on the pareto front. Gaussian mutation is also employed on the pareto front to generate diverse solutions and create a balance between exploitation and exploration. Moreover, a smart grid-based method is proposed to initialize the population to generate diverse solutions and reduce the variance between the worst and best clustering results. The proposed method is compared with state-of-the-art methods on seven challenging datasets. FMOPG has improved clustering accuracy by 3.65% compared to other automated methods. Moreover, the incremental technique has improved the clustering time by 71.18%.

Index Terms—Human activity discovery, unsupervised learning, clustering, feature extraction, incremental manner, multi-objectives optimization, dimension reduction, skeleton sequence.

I. INTRODUCTION

HUMAN activity recognition (HAR) has a tremendous impact in computer vision and is considered by many researchers due to its many applications including security surveillance, health care, game control, and robot vision [1]. HAR aims to analyze ongoing activities automatically so that it can correctly categorize the activities performed. As shown in Fig 1, a HAR system has five major steps. In the first step (Fig 1(a)), people’s activities are recorded. Recent researches have shown that 3D-skeleton data is reliable and can be easily recorded with low-cost depth sensors [2], [3]. Further, depth sensor have the advantage of not capturing personal identity images. Each frame of 3D-skeleton data has three-dimensional coordinates of the body’s joints and is appropriate for displaying human activities [7].

In the feature extraction step (Fig 1(b)), the purpose is to find the most relevant set of compact and descriptive information that show the distinct patterns of each activity. Good features help the HAR system to distinguish and identify human activities well. Feature extraction methods can be divided into 3 categories namely displacement, statistical, and orientation. In this paper, a combination of all approaches presented in [8] is used to describe the human body’s posture and movement. Human activity discovery (HAD) is one of the crucial tasks in a HAR system. In this step (Fig 1(c)), there is no prior knowledge and information about the activities observed. Activities are identified and clustered without supervision based on the similarities between the extracted features. The concept of HAD is like a child learning to understand movement of people around him/her. A child is able to perceive the difference between activities but has no knowledge of the concept or label of them and initially tries to differentiate them based on similarities and differences. HAD is very challenging due to the lack of label information to learn about the human activities. Although unsupervised methods have been proposed for HAR, they require well-segmented videos, where each segment contains exactly one activity sample. These methods are also aware of the number of performed activities. In reality, the received videos are not segmented and there is no information about the number of performed activities. In this paper, we address the aforementioned challenges in dealing with unsegmented videos by using the sliding window technique. Moreover, an incremental technique is applied to multi-objective clustering to deal with the problem of finding the number of clusters.

In the learning step (Fig 1(d)), the HAR system is trained by applying a learning algorithm on clustered activities to obtain a model for accurate classification of each activity discovered. The model is then used to classify new input activities ((Fig 1(e))). These two steps have made significant progress and many supervised methods have been proposed for learning and modeling the activities, such as dynamic time warping [9], hidden Markov model [10], CNN based [11], [12] and RNN based approaches [13], [14], [15], [16]. However, HAD has been ignored by these methods and their
models are trained using ground truth labels. In real-world scenarios, there is no label, leading to a severe decline in the efficiency of these methods because of their dependence on labeled data. Supervised models must first be extensively trained using labeled data to identify human activities. This data must be labeled by humans and can be proven to error. Training the model is not only time consuming, but it is also not scalable due to the use of limited training data. In other words, the model has difficulty identifying activities performed by different people as well as new activities that have not been trained for their models. This problem severely reduces the performance of HAR. But unlike supervised models, in unsupervised methods, no one needs to understand and then label the input data. This makes unsupervised methods more feasible than the supervised methods because the labeling process is removed in unsupervised method and the recognition of activities are done based on received patterns. It is also scalable because it can detect new activities based on received patterns and can be used anywhere.

In this paper, a novel, Flexible Multi-Objective Particle swarm optimization clustering based on Game theory (FMOPG) is proposed to perform the HAD fully unsupervised on 3D skeleton data. The main contributions are summarized as follows: 1) A flexible multi-objective Particle Swarm Optimization (PSO) clustering based on game theory and incremental technique is proposed to estimate the number of clusters and find good clusters. 2) A new smart grid-based swarm initialization method is introduced to initialize PSO particles. 3) To deal with getting stuck in local optima and premature convergence, a method for updating particle velocities based on mean shift clustering is introduced to prevent PSO from getting stuck in the local optimum and to detect and separate non-linear clusters by using the kernel function in mean shift clustering. 4) The proposed method is highly competitive and superior to the compared latest proposed algorithms for HAD on seven challenging 3D skeleton-based activities datasets.

The remainder of this paper is structured as follows: in section II, the research background and relevant methods for human activity discovery are reviewed. The proposed approach is described in section III. In section IV, the proposed method is evaluated and compared with other methods, and finally, in Section V, the conclusion is stated.

II. RELATED WORK

A. Automatic clustering

The number of clusters cannot be predetermined in real-world data clustering analysis, and establishing the appropriate number of clusters for a huge and complex dataset is a tricky process. Many methods have been proposed to perform multi-objective clustering automatically. In multi-objective clustering with automatic K determination (MOCK) [17], non-convex solutions were removed from the pareto front. Then, the knee was determined as the number of clusters based on gap statistics. In [18] the number of clusters was determined from the best solution by performing the algorithm from predefined maximum to minimum numbers of clusters. In [19] multiple clustering validity indexes were employed to find the optimal number of clusters and the best possible solution. The proposed method went a step further and instead of using validity index methods for estimating the number of clusters, which is time-consuming and problem-specific, an incremental technique was employed, which like human learning, gradually identifies and categorizes data.

B. Initialization methods

Initialization is one of the most influential factors in clustering and population-based algorithms. One of the common methods is Forgy’s method [20]. There is no theoretical basis in this method and data points are selected as centroids randomly. In the initialization method of k-means++ [21], one data point is selected randomly as a centroid. The distance of all data points from the selected centroid is computed, and a data point based on maximum probability is selected. The further the data point is from selected centroids, the higher the chances of being selected as a centroid. However, all of the methods mentioned above perform initialization in a random point, which makes these methods prone to local minima. For this reason, Bajer al el. [22] employed a clustering approach to find the potential area and used Cauchy mutation to generate individuals from each selected area. Although they enhance the initialization, their method suffers from high complexity and is time-consuming in high-dimensional data. The main difference between our method and the previous method is that our method does not start from a random point, and its complexity is low. The proposed method divides the feature space into the $k \times k$ hyper square grid cell and selects the cells with the largest population according to the number of clusters. Centroids are randomly selected from those cells.

C. Human activity recognition and discovery

HAR is one of the most challenging topics in computer vision. Many works presented in this domain use RGB-videos [23], [24], [25] and sensor-based [26], [27], [28] data. However, these data have problems such as a large amount of information to process, high amount of noise, difficulty in recording data, and high cost [13]. 3D skeleton-based data are widely used in recent research to mitigate some of the problems of the other data types [2], [3], [4], [5], [6]. Most methods introduced for HAR are supervised focusing in Fig 1 part (d) and (e), training from labeled data. [29] described the relation between skeleton joints in groups activity recognition, they used deep reinforcement learning and formulate joints as a Markov model to select informative ones [30], continuously learned from skeleton activity using a brain-inspired elastic network [31], and a hybrid CNN-LSTM to extract spatial and temporal features [32]. Miao et al. [33] introduced a novel graph convolution operator that collected dynamic gradient information linked to local motion between the central joint and its associated adjacent joints for feature aggregation. Furthermore, because adjacency matrices are typically time-consuming, features were aggregated nearby the central joint using a simple graph shift operation and point-wise convolutions. In [34] a method was proposed for encoding the spatiotemporal information of a 3D skeleton. The
Fig. 1: Conceptual framework for human activity recognition system: (a) RGB-D sensor captures input frames and converts them to skeletal data. Then (b) extracting features from skeleton data, (d) activities are clustered based on the similarities and differences of features. Next, the system (d) learns the model for each activity based on clusters obtained in discovery step and finally (e) human activities can be recognized.

joints of each sequence were first coded into three-dimensional colored dots and then projected onto three orthogonal planes. Three CNNs were used to extract features from each image. The final classification of the presented sequence was obtained by combining the CNN scores. Zhou et al. [35] proposed a method for learning a visual pose model and a pose lexicon model. Their method is made up of two-level hidden Markov models. On one level, the alignment between the visual poses and semantic poses was represented, while on the other level, a visual pose sequence was represented as a Gaussian mixture. Then, activities were classified by formulating the classification problem based on the acquired lexicon. Li et al. [36] introduced a scheme to represent relationship between skeleton joints. They used sub-network to represent spatial and temporal skeleton joint connectivity graph with a frame attention model and LSTM network respectively. Gao et al. [37] presented a graph regression to extract spatial and temporal features. They combined graph regression with graph convolutional network to deal with data that is not arranged in normal graph. Koniusz et al. [38] proposed a novel feature representation to capture importance interaction between visual information. They designed dynamic compatibility kernels to build spatial and temporal relationship among features. The performance of these methods was strongly dependent on training data labeled with ground truth. In our proposed method, no labeled data is used to categorize activities. It automatically categorizes activities based on differences and similarities.

A branch of works tries to explore HAR in an unsupervised manner. Mohammadzade et al. [39] projected temporal and spatial features into the low dimensional unwarped space by using a joint learning strategy. Yang et al. [5] designed a skeleton cloud colorization scheme that encodes spatial, temporal, and person-level information. Su et al. [40] used an autoencoder to extract the features. To classify activities, encoded features were given to the KNN classifier. Zheng et al. [41] extracted deep features for classifying actions. They introduced a GAN autoencoder and extracted the dynamic motions from skeleton frames. Tang et al. [42] developed a new graph convolutional network to transfer the knowledge between data with different distributions. They fed the network with the source domain and target domain to extract features. The extracted feature from the source domain was used to train the label predictor under the supervision of source labels. After reducing the domain shift between the two domains, two weight matrices were fed into the relevant domain classifier. The issue with these approaches is that unsupervised learning was only used for feature extraction and ground truth was needed to categorize activities with supervised methods. Guo et al. [43] proposed a semi-supervised method that can recognize activities with a few examples to deal with scarce training data. They used an interactive graph structure to generate the representation of activities and match them with a few frames. Liu et al. [44] proposed a framework for one shot activity recognition. They examined the semantic significance between the activities and each body component based on their descriptions. Then, the framework stresses the important body parts for each class of activities for representation. The difference between these methods and the method of this paper is that they use a small amount of labelled data to categorize activities, but our method does not require any labelled data. Generally, the task of our proposed method is to do HAR fully unsupervised without using ground truth in any part of the HAR system.

A limited number of research attempted to address HAD using a fully unsupervised approach. An early HAD literature used incremental k-means clustering and a hand-crafted feature method to discover activities [45]. They also built a model for activities by using the mixture of the Gaussian Hidden Markov Model. These methods used all joints from all frames to extract the feature. Noisy data and outliers were not considered. Paoletti et al. [46] used subspace clustering to
discover human activities. To reduce the number of redundant frames, they introduced a trim method. Even though their approach has achieved promising results, they cannot handle complex scenarios because their inputs were well-segmented videos, and they required prior knowledge about the number of performed activities.

Hadjikhan et al. [8] presented a clustering method based on PSO that received unsegmented frames as inputs and clustered them. They presented feature extraction from informative skeleton joints by combining different techniques. They employed a kinetic energy-based method to solve the redundant data problem and extract effective frames. Most of these works used a single objective to cluster activities and did not achieve good results for all datasets. A single objective does not operate equally well for all datasets due to differences in data features. As a result, optimizing various objectives such as symmetric and connectivity at the same time to capture distinct dataset properties is more useful.

To solve HAD in a fully unsupervised way, a multi-objective clustering approach with a cluster validity index to estimate the number of activities was presented in [47]. In addition, a game theory method was applied to select the best solution in the multi-objective problem. Different from [47], in this paper, an efficient incremental technique is applied to multi-objective clustering to automatically detect the appropriate clusters instead of identifying the number of clusters through an exhaustive iterative process from maximum to minimum k values. In other words, for each number of clusters in [47], all data points are clustered together regardless of the quality of each cluster. However, in the proposed method, the best clusters are gradually selected during the clustering process by examining the quality of each cluster and considering the conditions of the data points in the selected cluster.

III. PROPOSED HUMAN ACTIVITY DISCOVERY

In this section, we introduce the proposed flexible multi-objective PSO clustering with the integration of game theory (FMOPG) to discover human activities. An overview of the proposed method is given in Fig. 2. In this method, after receiving the frames containing data of human skeletal joints in 3D space, keyframes are selected based on the kinetic energy of the joints. Based on informative joints selected from important parts of the body, the characteristics of the relevant skeletons are then extracted. Next, 3D skeleton sequences of frames are segmented into the activity instances using a fixed size sliding window technique. Finally, these instances are clustered using the proposed clustering method.

A. Pre-processing

Not all frames recorded are useful. Redundant information contained reduces the efficiency of activity discovery and increases computation time. For this reason, the kinetic energy of each frame \((E_i)\) is calculated based on the joints using Eq. (1) based on [8]. The movement of joint \(j\) in frame \(i\) and \(i-1\) is calculated. This process is done for all joints \(j\). The sum of these movements for all joints is the current frame’s energy. The maximum and minimum local energy are then selected as keyframes to keep frames that represent the main contents of a video and have more valuable information [48] (the effect of selecting the keyframe based on kinetic energy is reported in Fig. 2 in supplementary).

\[
E(f_i) = \sum_{j=1}^{J} E(f_i^j) = 1/2 \sum_{j=1}^{J} (f_i^j - f_{i-1}^j)^2
\]  

To better display the raw data and obtain the relevant and important features of activities, we use three categories called displacement features, statistical features and orientation features to extract the features. Before extracting features, we select some joints as informative joints because not all joints and parts of the body like arms and legs provide useful information. Some joints have low mobility during the activities and only give information that may negatively affect the performance of activity discovery. The informative joints include both left and right hands, feet, hips, shoulders, elbows, and knees. Displacement features are divided into two groups including spatial displacement and temporal displacement. In spatial displacement, the Euclidean distance between joints of both hands, hands and head, and hip and feet at both sides in the same frame are computed which gives 5 features. In terms of temporal displacement, features are obtained based on the difference of each informative joints in the current frame from the same joints in the previous frame to represent the instantaneous changes. In addition, the difference between selected joints in current frame from the neutral frame is used to represent changes in activities over time giving a total of 2 (previous frame and neutral frame) \(\times\) 3 (coordinate of \(x,y,z\) of each joint) \(\times\) 12 (number of informative joints)=72 features. Mean and standard deviation differences features are obtained by computing the difference of each coordinate of the selected joints in the current frame from the mean and standard deviation of the same joint within a sequence including different activities to distinguish between the lower torso and upper torso activities that give 72 features (36 features for mean and 36 features for standard deviation). To describe human body posture, the orientation and angle features are extracted. For obtaining the orientation between the selected bones (left hip-knee with left Knee-foot, right hip-knee with right knee-foot, head-neck with right hand-elbow, head-neck with left hand-elbow, left hand-elbow with right hand-elbow, left elbow-shoulder with left hand-elbow, right elbow-shoulder with right hand-elbow), a rotation matrix is used. This matrix is calculated using the bones’ rotation angles relative to each other. These angles are computed with the assistance of internal and external products between the bones. The rotation angles relative to the \(x, y,\) and \(z\) axes are considered as orientation features (3\(\times\)7=21 features). Moreover, the angle features are computed based on the angles between the bones of elbow-wrist and shoulder-elbow at both sides and the angles between the bones of hip-knee and knee-ankle at both sides which give 4 features (Readers are referred to [8] for more details about the feature extraction method). After obtaining the features, PCA is applied to reduce the data dimension and computation time. The number of components is determined so that explain 85% of the variance of extracted
Preprocessing

Fig. 2: The proposed system is depicted in this diagram. Pre-processing based on [8] are performed, which includes choosing keyframes, extracting features from informative joints, dimension reduction, and segmentation of frames into fixed-length time overlapping windows before clustering the activities. In the clustering phase, the number of clusters starts from $k_{\text{max}}$ and continues until it reaches $k_{\text{min}}$. Initial centroids are chosen in the discovery process by a smart grid-based strategy to generate diverse solutions. After initialization, non-dominated solutions are obtained based on the assessment of each solution using the objective functions utilized. Next, Gaussian mutation is applied on non-dominated solutions to avoid getting stuck in the local optimum trap. Nash equilibrium is then employed to determine the global best solution. Particles update their position and velocity by adopting the mean-shift vector to deal with premature convergence and non-linear clusters. The steps from finding non-dominated solutions to updating the particles are repeated until the maximum iterations is reached. Using both objective functions, all global best clusters are then evaluated and the best cluster is selected by roulette wheel selection. If the condition of minimum members of the cluster is met, the selected cluster is collected as one of the final clusters and its members removed from the dataset. The number of clusters is recalculated based on the size of the trimmed dataset. The clustering process is repeated based on the new number of clusters.
features. The sequence of frames is then partitioned into overlapping fix-sized activity instances. The size of activity instances should be large enough to be meaningful in the discovery phase. In this paper, the size of each activity instance is 15. Overlap between sliding windows has a positive impact due to the retention of information when transferring between activities and leads to increase clustering performance [49]. Except for the first activity instance, the first frame of the rest of the activity instances overlaps with the last frame of the previous activity instance. These techniques of keyframe selection, feature extraction and dataset preparation briefly discussed here has been well-described in [8].

B. Smart grid-based initialization

Initialization of particles has a significant task that influences on diversity and convergence. For this reason, a new smart grid-based initialization method is proposed. This method can detect density areas that are suitable area for initialization and producing diverse solutions. First, the lower and upper bounds of the dataset are computed. Feature space is divided into the \( k \times k \) non-overlapping regular hyper square grid cells. The positions of each cell are selected and the most number of data points are assigned to each cell. Next, the activity instances are mapped to grid cells with the most number of data points. After assigning the data points to grid cells, a data point is assigned to the cell on the top-right. If a data point is on the edge of grid cells, it is assigned to the top-right cell. After assigning the data points to grid cells, the rest of the activity instances overlaps with the last frame of the previous activity instance. Finally, the cluster centers are placed in one particle. The process of initialization is shown in Fig. 3 (see Algorithm 1).

C. Flexible multi-objective clustering

PSO is a population-based algorithm. The position of each individual particle in the population represents a solution. Each of the particle updates its position using its velocity to reach the optimum solution [50]. Position and velocity of each particle are obtained from the Eq.(2) and Eq.(3). The parameters in the velocity update in Eq.(3) are adjusted in Eq.(4), (5), and (6) to keep the balance between local and global search.

\[
x_i(t+1) = x_i(t) + v_i(t)
\]  

(2)

Algorithm 1: Smart grid-based initialization algorithm

| Input: \( D=\{d_1,d_2,\ldots,d_n\} \) //Set of data points  
|  \( k \leftarrow \) Current number of clusters  
| Output: \( C=\{c_1,c_2,\ldots,c_k\} \) // A set of centroids  
| 1 Find the maximum and minimum values in each dimension of datapoints  
| 2 Divide phenotype space into \( k \times k \) non-overlapping regular hyper square grid cells  
| 3 for each datapoint do  
| 4 Map \( d_i \) to a grid cell according to its feature values  
| 5 if datapoint \( d_i \) is on the edge (coordinates of the cells) then  
| 6 Assign \( d_i \) to the top-right cell  
| 7 Choose one datapoint randomly as a centroid \( c_1 \)  
| 8 Collect all centroids into a set called \( C \)  
| \[
v_i(t+1) = w \times v_i(t) + c_1 \times rand_1 \times (pbest_i(t) - x_i(t)) + c_2 \times rand_2 \times (gbest(t) - x_i(t))
\]  

(3)

\[
w = \frac{w_{\text{max}} + t \times (w_{\text{max}} - w_{\text{min}})}{t_{\text{max}}}
\]  

(4)

\[
c_1(t+1) = (c_{1\text{max}} - c_{1\text{min}}) \times \frac{t}{t_{\text{max}}} + c_{1\text{max}}
\]  

(5)

\[
c_2(t+1) = (c_{2\text{max}} - c_{2\text{min}}) \times \frac{t}{t_{\text{max}}} + c_{2\text{max}}
\]  

(6)

where \( x_i(t) \) is the position and \( v_i(t) \) is the velocity of the particle of particle \( i \) at time \( t \). \( pbest_i \) and \( gbest \) are the local best position of particle \( i \) and \( gbest \) is the global best position of the population, respectively. \( rand_1 \) and \( rand_2 \) are random value parameters with range \([0,1]\). The parameter \( w \) is a constant weight of inertia. \( w_{\text{max}} \) is initial weight. \( c_1 \) and \( c_2 \) are acceleration coefficients [51].

D. Multi-objective clustering and the global best selection

In one-objective clustering, datasets are clustered based on only one property. Other characteristics like compactness and separation are not taken into account. As a result, multi-objective clustering is used to investigate multiple criteria for clustering datasets in order to optimize solutions at the same time. The following is a formula for a multi-objective problem:

\[
\text{Minimize } f_i(x) : i = 1, \ldots, N
\]  

(7)

where \( f_i \) is the \( i^{th} \) objective function, \( N \) is the number of objective functions, and \( x \) is the decision vector. This paper presents a multi-objective method for PSO. Here, several optimal solutions called pareto-optimal sets (non-dominated solutions) are obtained. These solutions are stored in a predefined size repository and whenever the repository is full, the less important solutions are removed with the help of the roulette wheel selection method.
Since the ability of the algorithm to explore or exploit is continually changing, the diversity of solutions is critical. For this purpose, Gaussian mutation proposed in [8] is applied on non-dominated solutions as follow:

\[ v'_{\text{non-dominated}}(d) = v_{\text{non-dominated}}(d) \times G(0, h) \times (x_{\max}(d) - x_{\min}(d)) \]

\[ x'_{\text{non-dominated}}(d) = x_{\text{non-dominated}}(d) + G(0, h) \times v_{\text{non-dominated}}(d) \]

Where \( x'_{\text{non-dominated}}(d) \) and \( v'_{\text{non-dominated}}(d) \) represent the position and velocity of \( i^{\text{th}} \) non-dominated particle in \( d^{\text{th}} \) dimension. \( x_{\max} \) and \( x_{\min} \) are the maximum and minimum value in \( d^{\text{th}} \) dimension. \( G \) is Gaussian distribution with the mean \( 0 \) and the variance \( h \). \( h \) is linearly decreased during each iteration according to Eq.(10) to ensure that the capability of exploration and exploitation are strong at the initial and later steps accordingly.

\[ h(t + 1) = h(t) - (1/t_{\text{max}}) \]

To find the global best solution, Game Theory (GT) is used to make decisions on pareto-optimal sets (See Fig. 4). GT is the study of how to calculate the optimum strategy for a given set of circumstances to maximize the outcome. A game space contains three key features: players on opposing sides of the game, strategies, which are taken actions by the player at various stages of the game, and payoff, which is the consequence of each player’s strategies. To map the multi-objective problem to the GT, each objective function is defined as a player in the game, particles are represented as players’ strategies, and the fitness value of each objective function can be regarded as a payoff. Since Strategic equilibrium is used to find a game’s solution, Nash Equilibrium (NE) is used to find the global solution from pareto-optimal set. In NE, each player chooses the best feasible strategy based on its interests, without collusion or assistance with other players. To pick the best global optimal solution from the pareto-optimal set, we use the following equations based on [52] to compute the NE.

\[ NashE_j = \sum_{i=1}^{N} OBJ_{ji} \]

\[ OBJ_{ji} = \frac{\text{currentOBJ}_{ji} - \text{BestOBJ}_{i}}{\text{BestOBJ}_{i}} \]

\( N \) is the size of the particle population. \( NashE_j \) is the NE criterion of the \( j^{\text{th}} \) individual and should be minimized. \( \text{currentOBJ}_{ji} \) is the \( i^{\text{th}} \) objective of the \( j^{\text{th}} \) individual. \( \text{BestOBJ}_{i} \) is the best fitness value of the \( i^{\text{th}} \) objective (the results of selecting the global best solution based on game theory is reported in Fig. 1 in supplementary).

E. Updating particles by adopting the mean-shift vector

Although PSO has a good ability to find a global area in the search space, it slows down as it approaches the global optimum. As shown in Fig. 5, PSO in its early stages moves with large steps towards the global optimum. However, over time the search process slows down dramatically as it becomes closer to the final solution. On the other hand, Mean-shift clustering has a rapid convergence in finding the local optimal. It is also able to find clusters of any size and shape due to its kernel function [53]. Therefore, we adopt Mean-shift clustering in PSO and update each particle based on the concept of mean-shift vector. In this way, before updating the particles, the radius of the clusters (\( \sigma \)) in the particles is calculated and considered as a bandwidth parameter in the mean-shift vector. To compute shifted centroid \( M(x) \), the weight of nearby data points is determined by kernel function \( kf(x) \) in Eq.(13). The \( M(x) \) in the obtained radius is determined by Eq.(14).

\[ kf(x) = \frac{1}{N} \sum_{i=1}^{N} exp\left(\frac{\text{centroid} - d_i}{\sigma}\right) \]

\[ M(x) = \frac{\sum_{i=1}^{N} kf(x) \times x_i}{\sum_{i=1}^{N} kf(x)} \]

where \( N \) is the number of data points in the cluster \( \text{centroid} \) and \( d_i \) is the \( i^{\text{th}} \) data point in cluster \( \text{centroid} \). After obtaining \( M(x) \) for each centroid, Eq.(3) is modified as follow:

\[ v'_i(t+1) = w \times v_i + c_1 \times rand_1 \times (\text{best}_i(t) - x_i(t)) + c_2 \times rand_2 \times (M(x) - x_i(t)) \]

In Eq.(15), the shifted vector \( M(x) - x_i(t) \) is calculated, which specifies in which direction and to what extent the particle should move towards the global solution. The pseudocode of the particle update process is given in Algorithm 2.

F. Detecting clusters automatically with incremental approach

In the real scenario, the number of activities is not known in the input videos. To tackle this, we present an incremental technique for multi-objective clustering. In this technique, unlike conventional methods that use cluster validity to estimate the number of clusters and are time consuming, not all data
Algorithm 2: Update particles

Input: Particle\(=\{p_1, p_2, \ldots, p_n\}\) //Set of particles
Output: Updated particles
1 for each particle do
2   for each cluster do
3       Calculate the cluster’s radius (\(\sigma\))
4       Calculate the weight of each datapoint in the
5       cluster using Gaussian kernel (Eq. (13))
6       Compute mean weight of the cluster by Eq. (14)
7       and save it as shifted centroid
8   Update velocity and position of each particle by
9       Eq. (15)
10  Update particle personal best (pbest)
11 Return updated particles

Fig. 5: Illustration of the motion of a particle that suffers from slowing down convergence as it approaches the global best solution and the effect of mean-shift vector in improving this problem.

are clustered at once like in Fig. 6(b). Instead, clusters are detected over time as shown in Fig. 6(a). As summarised in Algorithm 3, for each value of \(k\), clustering is done on the dataset. After obtaining the global best particle, each cluster is evaluated based on both objective functions. The plane is mapped into a grid with equal units and the centroids are placed in the cells based on their fitness values. Using the roulette method, a cell is selected. If more than one centroid is in the selected grid, one centroid is randomly selected. To guarantee that the clusters have enough observations to represent the activities, the selected cluster should have a minimum membership parameter (MinPt) which is set up based on \(\frac{n}{k_{max}}\). If the number of members in the selected cluster meets the MinPt constraint, the cluster is chosen as one of the final clusters, and its members are removed from the dataset. A new value for \(k\) is calculated from new population \(n^*\) of the trimmed dataset based on \(\sqrt{n}^*\). Otherwise, it is ignored and the value of \(k\) is decremented. This process continues until the value of \(k\) reaches \(k = 2\). Finally, the selected clusters are aggregated during the clustering process and are considered as the final result. The proposed clustering algorithm that integrates Algorithm 1, 2 and 3 is shown in

Algorithm 4.

Algorithm 3: Incremental technique to detect clusters automatically

Input: Global best solution
Output: Collect a potential cluster
1 for each cluster of global best solution do
2   Compute fitness values based on both objectives in
3       Eq. (17) and (18)
4   Use the roulette wheel selection method to choose a
5       cluster as \(c_i^*\)
6   if size of \(c_i^*\) > MinPt then
7       Add \(c_i^*\) to cluster set C
8       Remove member of \(c_i^*\) from dataset by
9       \(Dataset^* = Dataset - c_i^*\)
10  Recalculate the number of clusters by \(k = \sqrt{n}^*\)
11 else
12     Recalculate the number of cluster by \(k = k - 1\)

IV. EXPERIMENTS

Seven challenging datasets including Cornell Activity Dataset (CAD 60) [54], Kinect Activity Recognition Dataset (Kard) [55], MSR DailyActivity3D (MSR) [56], UTKinect-Action3D (UTK) [57], Florence3D (f3D) [58], NTU RGB+D Dataset (NTU-60) [59] and NTU RGB+D 120 (NTU-120) [44] were used to evaluate the proposed method. These datasets are challenging as they contain large-scale performed activities, high overlap between activities and many disruptions in capturing activities. We evaluated the performance of the proposed method based on these challenges. The details of each dataset are given in Table I. To show the effectiveness, we compared the proposed method with non-automatic methods (having prior knowledge of the number of clusters) including KM (k-means), SC (spectral clustering) and SSC (Sparse Subspace
algorithm, we have chosen them to show that the proposed method has improved their performance. Moreover, we have compared FMOPG with prior related supervised methods applied to CAD-60 dataset including Dynamic Bayesian Network [54], Spatio-Temporal Interest Point [65], SVM+Hidden Markov Model [55], Atomic motion+naive Bayes+nearest neighbor [66], Bags of visual words+Fisher vectors+SVM [67] and Convolutional Neural Networks+SVM [68] to demonstrate the performance of our method against methods that have been trained and have prior knowledge about activities. To evaluate the performance of the incremental technique, which we proposed for determining the number of clusters, we compared it with cluster validity index methods including Silhouette Index (Sil) [69], Calinski-Harabasz (Ch) [70], Davies-Bouldin (Db) index[71], Dunn index [72], Gap statistic [73], Elbow [74], Hartigan (Ha) index [75], Krzanowski-Lai (Ki) [76], Slope [77] and Jump [78].

<table>
<thead>
<tr>
<th>Dataset</th>
<th>CAD 60</th>
<th>KARD</th>
<th>MSR</th>
<th>UTk</th>
<th>F3D</th>
<th>NTU-60</th>
<th>NTU-120</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activity</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Subjects</td>
<td>14</td>
<td>18</td>
<td>16</td>
<td>10</td>
<td>9</td>
<td>60</td>
<td>120</td>
</tr>
<tr>
<td>Videos</td>
<td>60</td>
<td>2160</td>
<td>320</td>
<td>200</td>
<td>215</td>
<td>56880</td>
<td>114480</td>
</tr>
</tbody>
</table>

A. Evaluation measures

We compared clustering algorithms using accuracy based on [24] to investigate their performance across 30 runs. F-score and confusion matrix have been used to show the performance of each method in categorizing the activities and the confusion between activities. Moreover, the overall error (OE) of estimating the number of clusters was computed for each automatic method as given in Eq.(16).

\[
\text{Overall error} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} k^i - k^i_p}
\] (16)

Where \(k^i\) is the actual \(k\), for subject \(i\), \(k^i_p\) is the predicted \(k\) for subject \(i\), and \(n\) is all subjects in the dataset.

B. Set parameters and implementation

The experiment was repeated 30 times. The number of iterations and the size of the population (particles) in methods PSO, HPGMK and MOPGMGT, FMOPG were equal to 20 and 50, respectively. The number of components in the GMM Algorithm was set to the number of clusters. Bandwidth in MS clustering was equal to 2. The minimum number of samples and maximum distance between two samples in DBSCAN were equal to 0.5 and 2, respectively. The initial value of parameters \(c_{1\text{max}}\) and \(c_{2\text{max}}\) in the proposed method were set to 2.5, \(c_{1\text{min}}\) and \(c_{2\text{min}}\) set to 0, and \(w_{\text{max}}\) set to 0.9. Stop criteria for all algorithms was based on the number of iteration. Since the videos have not been segmented in the proposed method, all compared methods in this paper received unsegmented videos for the same evaluation. All methods were applied on segmented videos produced by using the sliding window technique used in [8]. All automatic methods used the
Jump method [78] to estimate the number of clusters. They were performed for different number of clusters in the range of \( k = 2 \) to \( K_{max} \). \( K_{max} \) was chosen based on \( \sqrt{n} \) where \( n \) is the number of data points. For each value of \( k \), the Jump value was calculated for them. Finally, the estimated number of clusters was determined based on the minimum value of Jump. The objective function SSE (sum square error) in Eq.(17) was used for all single and multi objectives clustering algorithms which should be minimized to achieve proper clustering.

\[
SSE = \sum_{k=1}^{K} \sum_{x \in c_k} ||x - \mu_k||^2
\]

\( x_i \) is a data point belonging to the cluster \( c_k \) and \( \mu_k \) is the mean of the cluster \( c_k \). The second objective function for multi-objectives clustering algorithms was Conn-index [18] which should be minimized. It is calculated as follow:

\[
Conn = \frac{\sum_{i=1}^{k} \min \sum_{j=1}^{n} d(p_i^j, m_j)}{n(\min_{i,j=1,i\neq j} d(m_i, m_j))}
\]

\[
m_i = \min_i^n (\sum_{j=1}^{k} d(p_i^j, p_j^i) / n)
\]

where \( n \) is number of objects in cluster \( c_i \) and \( p_j^i \) is the \( j \)th object of cluster \( i \).

C. Discussion and results

Fig. 7 compares the best, worst, and average accuracy obtained by the different methods (numerical results are reported in Table 1 in supplementary). As it is shown, except for the F3D dataset that SSC (knows the number of clusters) has achieved a better performance in terms of average accuracy, in other datasets, the proposed method has had a significant advantage over other methods. The overall average accuracy of the FMOPG was 80.99 % for CAD-60, 54.70 % for UTK, 55.76 % for F3D, 40.34 % for MSR, and 43.76 % for KARD, 36.70 % for NTU-60 and 19.82 % for NTU-120. FMOPG has performed better among all automatic and non-automatic algorithms. Because other methods do not have a suitable technique for initialization of clusters to produce various solutions and improve exploration that makes the distance between the minimum and maximum accuracy in FMOPG is the lowest compared to other methods. FMOPG also used the concept of MS clustering in refining the particle velocity update to deal with early convergence [53]. It enabled PSO’s ability to have a better search around discovered solutions and prevented particles from slowing down to reach the final solution. That is why better results have been obtained compared to other algorithms. The integration of the incremental technique to find the clusters dynamically has caused, instead of estimating the number of clusters, FMOPG finds suitable clusters with good connectedness and separation. This has improved the accuracy results compared to methods that use the cluster validity method to estimate the number of clusters. In deep clustering methods including N2D, DCN, SDCN and COMPLETER, despite using deep learning to better represent the activities, they could not obtain good results because the used networks in these methods are not able to extract spatial and temporal features that are very effective to discover activities. They also use shallow clustering methods, such as k-means that easily get stuck in the local optimization. In terms of subspace clustering including SC, ENSC, and SSC algorithms, knowing the number of clusters, they did not achieve good results. These methods are in lack of appropriate strategy to find clusters and make a good balance between exploration and exploitation [79], [80]. Density-based algorithms also did not achieve good results due to their strong dependency on adjusted parameters. These algorithms easily get stuck in local optimization because they do not have an alternative strategy for exploring the search space. For the rest of the comparisons, PSO, HPGMK, MOPGMGT and FMOPG have been used because clustering is done automatically in these methods. The numerical results of Fig. 7 have been presented in the supplementary.

Table II shows a comparison of FMOPG performance with supervised and unsupervised methods based on precision for CAD-60. As can be seen, FMOPG is superior to unsupervised methods. However, compared to supervised methods, in spite of outperforming some supervised method and and obtaining acceptable result by FMOPG, it was not able to obtain the best results. Because, supervised methods use ground truth and are trained using a large number of labelled activities. In other words, these methods are already aware of the number of activities and the class of activities. But in our method, ground truth is not used and there is no information about the number of activities.
Table III shows a comparison of clustering time in seconds (s) between methods PSO, HPGMK, MOPGMGT, and FMOPG. This demonstrates that using the incremental technique is very effective for reducing clustering time. The time spent for clustering by FMOPG in five datasets CAD-60, UTK, F3D, MSR, KARD, NTU-60 and NTU-120 were equal to 168.26s, 10.30s, 7.90s, 66.31s, 40.60s, 498.58s and 995.16s, respectively. Although FMOPG has to optimize two functions at the same time, in the F3D, Kard, NTU-60 and NTU-120 it took less clustering time than single-objective PSO and HPGMK. This indicates that not only FMOPG has solved the problem of finding clusters, but also using incremental technique shorten clustering time making it computationally competitive with single-objective methods. The convergence of the four methods PSO, HPGMK, MOPGMGT, and FMOPG has been investigated in Fig. 8. Apart from the fact that the use of smart grid-based initialization has led to better convergence compared to other methods in the early stages, the proposed method has avoided premature convergence in the final stages. Because in mean-shift vector, the kernel function transforms the data into a higher-dimensional space in which they are separable and enables the proposed method to deal with complicated clusters with non-convex shapes. However, PSO and MOPGMGT did not achieve the best possible solution due to the lack of a suitable strategy to deal with non-linear cluster. Regarding HPGMK, although the combination of PSO with KM has prevented the occurrence of premature convergence in the final stages, the converged value of HPGMK is not as good as the FMOPG because of its single objective approach. Table IV shows the total error for obtained number of clusters by different approaches for each dataset by the different approaches. Table V indicates the effect of PCA and keyframes on HAD performance. We can observe that combining both methods on how much the results have improved. However, cluster validity index methods not only have not performed well in detecting the number of clusters and have high overall error, but the method that works better than others, in one dataset, works worse other datasets. This indicates that their results are not stable.

Fig. 9 shows the average OE of the clusters found in the proposed method compared with the number of clusters estimated in MS, DBSCAN, PSO, HPGMK, MOPGMGT. From the results, it is obvious that the proposed method was able to find and cluster the activities better than the others. For this reason, it has the lowest OE compared to density-based clustering and other methods that have used the Jump method to estimate the number of clusters. It is also important to note that due to the use of PSO as the core of their method, PSO, HPGMK, MOPGMGT and FMOPG obtained approximate average errors. But in general, FMOPG has less error than these methods because it has modified the search process of PSO algorithm using the concept of Mean-Shift (MS) clustering. In the other three methods, the random method is used to generate the population, while in FMOPG, it starts to search from a better position using smart grid-based initialization.

Table V indicates the effect of PCA and keyframes on HAD performance. We can observe that combining both methods
Fig. 9: Comparison of average overall error for detecting the number of clusters by methods MS, DBSCAN, PSO, HPGMK, MOPGMGT, and FMOPG for all datasets.

not only reduces the clustering time but also has a positive effect on the HAD performance due to the selection of frames that show the most differentiation and reducing the number of features to those most useful to HAD by keyframes and PCA, respectively. Fig. 10 indicates the effect of mean-

Fig. 10: Effect of Meanshift and Smart initialization on the performance of the FMOPG on subject one in F3D.

Table V: Impact of PCA and Keyframes on the performance of HAD and time of discovery in MSR.

<table>
<thead>
<tr>
<th>Method</th>
<th>Performance</th>
<th>Time (second)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FMOPG without PCA and Keyframes</td>
<td>41.70</td>
<td>371.39</td>
</tr>
<tr>
<td>PCA</td>
<td>46.23</td>
<td>203.48</td>
</tr>
<tr>
<td>Keyframes</td>
<td>44.07</td>
<td>187.59</td>
</tr>
<tr>
<td>PAC+Keyframes</td>
<td>48.02</td>
<td>90.35</td>
</tr>
</tbody>
</table>

Fig. 11: Comparison of confusion matrices between the three methods HPGMK (a), MOPGMGT (b) and FMOPG (c) on subject 8 in MSR dataset. Activity list: (1) Drink; (2) Eat; (3) Read book; (4) Call cellphone; (5) Write on a paper; (6) Use laptop; (7) Use vacuum cleaner; (8) Cheer up; (9) Sit still; (10) Toss paper; (11) Play game; (12) Lie down on sofa; (13) Walk; (14) Play guitar; (15) Stand up; and (16) Sit down. AVG is the average F-score for all activities.
the confusion between activities compared to the other two methods. The other two, use Euclidean distance for clustering activities and will have difficulty to find non-linear clusters.

V. CONCLUSION

In this paper, we have proposed a novel flexible multi-objective clustering based on game theory to address human activity discovery. A new smart grid-based initialization method was introduced to improve the initialization of the centroids. We modified particles’ velocity update by using a mean-shift vector to cope with premature convergence and non-linear clusters. Gaussian mutation was employed to generate diverse solutions. Besides, an incremental technique was presented to perform clustering dynamically and flexibly without prior information about the number of clusters. The effectiveness of the proposed method was shown on seven challenging datasets. The proposed method outperformed the state-of-the-art methods in all evaluation parameters and has improved the accuracy on the CAD-60, UTK, MSR, and NTU-60 by 8.66 %, 1.75 %, 3.56 %, and 1.27 %, respectively. Although FMOPG has shown great performance in HAD, hand-crafted features are very sensitive to parameter tuning and are not reliable for a real scenario. As future work, the hand-crafted method used can be replaced by a feature learning approach. Sliding window segmentation method can be enhanced using a learning approach to detect the length of the sliding window dynamically based on input streams of skeleton sequences. Lastly, the applications of FMOPGM to address real-world problems such as analyzing the driver behavior, finding the community in social networks, and face recognition are other possible future works.

ACKNOWLEDGMENT

This work was supported by Grant UBD/RSC/1.11/FICBF(b)/2019/001 from Universiti Brunei Darussalam.

REFERENCES


