An Efficient Framework for the Clustering of Human Activity Data using Kernelized Robust Covariance Descriptors

Guntru Prasanth Kumar*, M. S. Subodh Raj and Sudhish N. George

National Institute of Technology Calicut, India

Abstract

In this paper, a new method for the efficient clustering of human activity data is proposed. Unlike the traditional human activity clustering approaches, our method relies on the skeletal data recorded with the help of motion capture (mocap) systems to achieve the goal. The proposed method is structured around the kernel-based robust covariance descriptor. By introducing a data re-framing technique that efficiently utilizes the temporal properties of the human activity data, we have alleviated the data redundancy and insufficiency issues associated with action sequences. The optimization model developed encompasses the combined benefits of low-rank representation and least square regression. The formulation is strengthened by incorporating the temporal dependency of the human activity sequences with the help of a temporal Laplacian regularizer. With the proposed algorithm, a representation matrix is learned from the raw data, which is then used to perform subspace clustering. Experiments conducted on multiple human activity datasets reveal the ability of the proposed method to achieve better clustering results compared to state-of-the-art counterparts.

Keywords

Human activity data, Kernelized covariance descriptors, Temporal Laplacian regularization, Temporal subspace clustering.

1. Introduction

Human activity recognition (HAR) from action sequences remains a challenging research topic in computer vision due to its multifaceted applications [1, 2]. HAR finds application in visual surveillance, healthcare, human-machine interface, video retrieval, and entertainment industry, to name a few [3, 4]. Traditional approaches to HAR use RGB video sequences as the input. Handcrafted features are later extracted from the video sequences for the purpose of activity recognition [3, 4]. Because of the high-dimensional nature of the video sequences, high computational complexity is often associated with such HAR approaches [1]. Later, sensor-based HAR gained popularity. The focus of such methods were on the data obtained from sensors such as accelerometer and gyroscopes. In such cases, the subject itself needs to be in possession of the sensor so that the movements of the human body can be recorded. This along with the

The 11th Colour and Visual Computing Symposium, September 08-09, 2022, Gjøvik, Norway

*Corresponding author.

prasanthkumarguntru7@gmail.com (G. P. Kumar); subodhrajms@gmail.com (. M. S. S. Raj); sudhish@nitc.ac.in (S. N. George)

0000-0002-1111-9520 (. M. S. S. Raj); 0000-0002-0886-9478 (S. N. George)

influence of noise acts as a limiting factor in sensor based HAR approaches [5, 1, 6]. With the evolution of mocap systems, new modalities were introduced to represent the human activity information. Such modalities include the motion depth maps and the skeletal representations [7, 8]. With skeletal representations, time series of 3D joint positions of the human body are recorded by the mocap systems [9]. The spatio-temporal quality of such recordings are superior that they find application in multiple domains including gait analysis, medical rehabilitation, and computer animations [7, 3].

The common approaches to HAR with mocap data involve supervised learning methods. Though they guarantee good results, such methods are often susceptible to missing sample issues. Further, the requirement of a huge clean dataset for initial learning of the system poses a serious bottleneck to supervised HAR approaches [8, 10]. This paves a foundation for the need of having unsupervised HAR strategies. Though in unsupervised methods the aforementioned challenges faced by supervised methods are alleviated, they encounter other limitations. The main challenge in unsupervised HAR is posed by the fact that the task needs to be performed in a robust and accurate manner without any prior knowledge about the data samples [11].

Recent studies showcase the ability of subspace clustering algorithms in dealing with high-dimensional data clustering problems [12]. The key idea in subspace clustering approach is to identify multiple low-dimensional subspaces from which the data originates [13]. The subspaces so identified house a cluster of data. This approach is commonly termed as the *Union of Subspaces* (UoS) model [14]. The popularity of subspace clustering approach has increased with the introduction of sparse subspace clustering approach in which a sparsity constraint is imposed on the coefficients in order to learn a sparse representation of the raw data [14]. Low-rank representation learning (LRR) [15] is another technique used with subspace clustering wherein the global structure of the data is considered while learning the coefficients. In LRR based approaches, a given dictionary is utilized to learn a low-rank representation of the data samples. The clustering results obtained with such low-rank representations are usually better [16]. Least square regression (LSR) [17] based subspace clustering in which the grouping of data samples are performed with the help of Frobenius norm operator is another promising approach in subspace clustering. The aforementioned approaches when used independently will not be suitable for HAR as they do not consider the time series information associated with the skeletal data. As a solution to this problem we have developed an approach called the time series activity clustering (TSAC) which utilizes the combined advantages of LRR and LSR. We have incorporated a kernel-based robust covariance descriptor to extract features out of the raw input data with the aim of exploring the non-linearity present in the data. Further, the temporal Laplacian regularizer is employed to capture the temporal dependency among the data samples.

The following are the main contributions of this work:

1. An unsupervised optimization model with improved performance is formulated for the clustering of human activity data. To effectively utilize the temporal dependencies of human activity sequences, a temporal Laplacian regularizer is introduced in the proposed model.
2. We have blended the LRR and LSR based subspace clustering approaches to achieve better clustering results. A clean dictionary is learned along with the representation matrix in

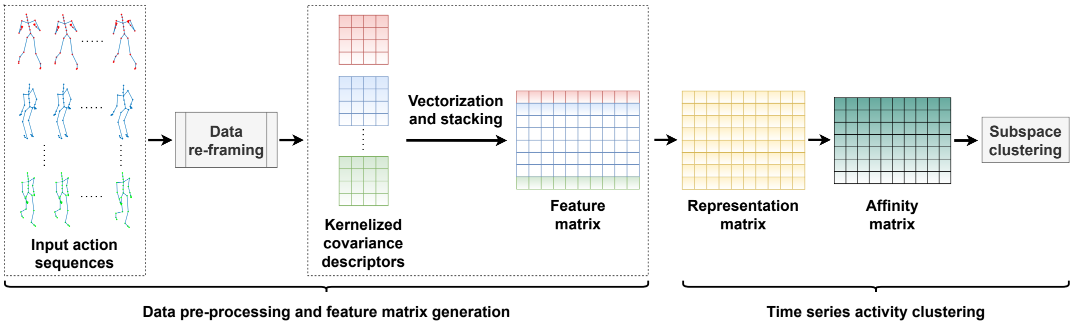

Input action sequences
Data re-framing
Kernelized covariance descriptors
Vectorization and stacking
Feature matrix
Representation matrix
Affinity matrix
Subspace clustering
Data pre-processing and feature matrix generation
Time series activity clustering

Figure 1: Overview of the proposed time series activity clustering

order to facilitate for the efficient use of the LRR model.

3. Data re-framing techniques are introduced to deal with data redundancy and insufficiency issues associated with the human activity timestamps. With the help of a robust kernel-based covariance descriptor, the underlying non-linear dependencies of the human activity timestamps are well utilized.
4. A representation matrix is generated by solving the formulated optimization problem using the Alternating Direction Method of Multipliers (ADMM) approach in an iterative manner. The representation matrix so obtained is later used to perform subspace clustering of the action sequences. The performance of the proposed model is verified against the state-of-the-art counterparts using multiple human activity datasets.

The rest of the work is outlined as follows. The proposed method, the problem formulation, and the solutions obtained are presented in Section 2. Section 3 explains the experimental validation done using the proposed method. Finally, conclusions are drawn in Section 4.

2. Proposed Method

In this section, we give a detailed outline of the proposed TSAC approach. The workflow of the proposed approach is shown in Fig. 1.

2.1. Data Re-framing

A sequence of human action is a collection of action timestamps evolved over a period of time. Two important observations can be made with reference to a given collection of action sequences as mentioned below:

1. The number of timestamps corresponding to each of the action sequences may not necessarily be the same. This disparity often appears as a bottleneck in generalizing any algorithm dealing with human activity recognition.
2. As the number of timestamps in an action sequence increases, it introduces additional computation overhead. That will lead to a rise in demand for resource utilization.

The aforementioned challenges can be addressed by standardizing the number of timestamps for all the action sequences under consideration. If the number of timestamps is standardized to be 'N', then data pruning methods need to be employed on action sequences having more number of timestamps and data augmentation needs to be performed on action sequences having lesser number of timestamps. The temporal smoothness property of human action sequences can be conveniently utilized to achieve this goal

Human action sequences are temporally highly correlated. As far as activity recognition is concerned, this correlation leads to redundant information. Often, the complete set of frames of a recorded action sequence is not essential to perform activity recognition. Thus, we introduce a pruning technique, termed *succession pruning*, in which the alternative frames of the action sequence are removed to eliminate redundancy while maintaining the temporal properties of the action sequence. That will drastically reduce the amount of data to be processed and will also result in reduced computation overhead and resource utilization. Whereas in action sequences experiencing insufficiency of timestamps, we perform *timestamp augmentation.* In this process, the trailing end of the action sequence gets augmented with the terminal timestamps of the same action sequence. That is in line with the temporal smoothness property of the human action data.

2.2. Feature Matrix Generation using Kernel-based Robust Covariance Descriptor

Human actions are represented in the form of skeletal structures with the help of modern motion capture systems. Each timestamp of an action sequence can be represented as a collection of '*n*' joints. Thus, the time stamp '*t*' of an action sequence can be represented as $\mathbf{e}(t) \in \mathbb{R}^{3\times n}$ of 3D positions $\{\mathbf{e}_1(t), \ldots, \mathbf{e}_n(t)\}$. The 3D coordinates of the i^{th} joint of the skeletal structure corresponding to the t^{th} timestamp can be denoted as, $\mathbf{e}_i(t) \in [x_i(t), y_i(t), z_i(t)]^\top$.

Once the data re-framing process is completed, the raw action sequences with fixed number of timestamps are represented in the form of a feature matrix. It involves a two step process. In the first step, we use the concept of covariance to obtain the covariance descriptor of each action sequence. The use of covariance will help us to capture the changes pertaining to each joint of the skeletal structure [18]. If $\boldsymbol{\mu}$ represents the temporal average of the timestamps of an action sequence, then the corresponding action sequence can be represented in the form of a covariance matrix as shown below.

$$\boldsymbol{\Psi} = \frac{1}{N-1}\sum_{t=1}^{N}[\mathbf{e}(t) - \boldsymbol{\mu}][\mathbf{e}(t) - \boldsymbol{\mu}]^\top \tag{1}$$

This process is repeated for each action sequence, resulting in a unique covariance descriptor for each input sequence.

Although covariance descriptors finds application in multiple domains, they cannot capture non-linearity present in the data. For making the feature matrix robust, different approaches are adopted to incorporate additional statistical information along with the covariance descriptor. This includes the entropy based approaches, the mutual information based approaches, and the kernel-based approaches. Among others, the kernel-based approaches have been used to simulate more complex models. The use of kernels improves the descriptive power of the covariance matrices. The work proposed by Cavazza *et al.* [18] showcases the benefits of using kernels in works related to human activity recognition. Motivated by this observation, we modify the expression given in Eq. (1) to incorporate the kernel function and obtain the following robust covariance descriptor.

$$\mathbf{\Psi} = \frac{1}{N-1}\sum_{t=1}^{N}\left[\mathcal{K}(\mathbf{e}(t)) - \boldsymbol{\mu}_{\kappa}\right]\left[\mathcal{K}(\mathbf{e}(t)) - \boldsymbol{\mu}_{\kappa}\right]^{\top} \tag{2}$$

Here, $\mathcal{K}(.)$ represents the kernel function and $\boldsymbol{\mu}_{\kappa}$ is the temporal average of the kernel entries. The choice of kernel function is application specific. We have used two kernel functions namely the polynomial kernel and the exponential kernel, out of which the later one have produced promising results.

The exponential kernel is defined as:

$$\mathcal{K}(\mathbf{e}(t)) = \exp\left\{\frac{\mathbf{e}(t)}{(\sigma+b)^2}\right\} \tag{3}$$

where $b > 0$ and σ is the kernel bandwidth.

After obtaining the robust covariance descriptors for each action sequence, in the second step we generate the feature matrix. The covariance descriptor $\mathbf{\Psi}$ contains redundant information as it is symmetric along the main diagonal. In order to reduce the amount of data to be processed, we vectorize each covariance descriptor by retaining the upper triangular values alone. Later, we stack (as columns) each of the vectors so obtained to form the feature matrix $\mathbf{X} \in \mathbb{R}^{p\times k}$. Here, '$p$' is the length of the individual vectors and 'k' is the total number of action sequences.

2.3. Temporal Subspace Clustering (TSC) with Laplacian Regularization

Subspace clustering is performed by generating a representation matrix out of the feature matrix in state-of-the-art approaches. This is accomplished by using the self representation property of the feature matrix, resulting in the computation of a set of unique coefficients $\mathbf{Y} \in \mathbb{R}^{k\times k}$ for the feature matrix $\mathbf{X} \in \mathbb{R}^{p\times k}$. This can be mathematically expressed as $\mathbf{X} = \mathbf{XY}$. The limitation of such an approach is that they tend to produce sub-optimal results if the sampling done is not sufficient. Instead of using the self representation property, if we learn an efficient dictionary $\mathbf{W} \in \mathbb{R}^{p\times k}$, we can overcome the aforementioned problem. Given a dictionary $\mathbf{W}$, the set of data samples $\mathbf{X}$ can be expressed as $\mathbf{X} \approx \mathbf{WY}$. The set of coefficients $\mathbf{Y}$ can be efficiently obtained in an iterative manner by utilizing the underlying low-rank nature of $\mathbf{Y}$. The low-rank property of $\mathbf{Y}$ can be capture with the help of LRR [15]. Since the rank minimization problem is inherently NP-hard, nuclear norm can be used as a substitute for rank minimization [17]. It is also important to obtain the intra-cluster correlation among the data samples to obtain better clustering results. To this end, we can use the principle of LSR [17]. By yielding the concepts of LRR and LSR we formulate an optimization problem as follows.

$$\min_{\mathbf{Y},\mathbf{W}} \frac{1}{2}\|\mathbf{X}-\mathbf{WY}\|_F^2 + \frac{\lambda_1}{2}\|\mathbf{Y}\|_F^2 + \lambda_2\|\mathbf{Y}\|_* \quad s.t. \quad \mathbf{Y} \geq \mathbf{0}, \mathbf{W} \geq \mathbf{0} \tag{4}$$

where, the term $\frac{1}{2}||\mathbf{X} - \mathbf{WY}||_F^2$ captures the reconstruction error, $||\cdot||_F$ represents the Frobenius norm operator, $|| \cdot ||_*$ denotes the nuclear norm operator, and λ_1 and λ_2 are the balancing parameters.

Manifold regularization methods are efficient in incorporating the temporal dependency of data in the problem formulation [19]. Thus, by modifying the general Laplacian regularizer, which captures the spatial dependency of data, we have developed a temporal Laplacian regularizer $\mathcal{L}(\cdot)$ as our interest is in the temporal information of the action sequences.

For a representation matrix **Y**, the temporal Laplacian regularization function can be defined as [20]:

$$\mathcal{L}(\mathbf{Y}) = \frac{1}{2}\sum_i \sum_j z_{ij}||y_i - y_j||_2^2 = \text{tr}\ (\mathbf{YL_TY}^\top), \tag{5}$$

where $\mathbf{L_T} = \widetilde{\mathbf{W}} - \mathbf{Z}$ is a temporal Laplacian matrix, $\widetilde{\mathbf{W}}_{\mathbf{ii}} = \sum_{j=1}^m z_{ij}$, and **Z** is a weight matrix that finds the successive similarities in **X**.

Each element of **Z** is found as [21],

$$z_{ij} = \begin{cases} 1 & \text{for } |i-j| \leq \frac{\gamma}{2} \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where γ denotes empirically defined threshold value.

With the introduction of $\mathcal{L}(\mathbf{Y})$, Eq. (4) is modified as,

$$\min_{\mathbf{Y},\mathbf{W}} \frac{1}{2}||\mathbf{X} - \mathbf{WY}||_F^2 + \frac{\lambda_1}{2}||\mathbf{Y}||_F^2 + \lambda_2||\mathbf{Y}||_* + \lambda_3\mathcal{L}(\mathbf{Y}) \quad s.t.\ \ \mathbf{Y} \geq \mathbf{0}, \mathbf{W} \geq \mathbf{0} \tag{7}$$

2.4. Solution

The optimization problem given in Eq. (7) can be solved using the ADMM approach under the ALM framework. ADMM finds solution for the unconstrained optimization problem by splitting the problem into multiple sub-problems. As a first step, we will introduce three auxiliary variables **E**, **F**, and **G** to decouple the terms present in the formulation. This results in a formulation as shown below.

$$\min_{\mathbf{Y},\mathbf{W},\mathbf{E},\mathbf{F},\mathbf{G}} \frac{1}{2}||\mathbf{X} - \mathbf{WY}||_F^2 + \frac{\lambda_1}{2}||\mathbf{E}||_F^2 + \lambda_2||\mathbf{F}||_* + \lambda_3\mathcal{L}(\mathbf{G})$$

$$s.t. \quad \mathbf{Y} = \mathbf{E}, \mathbf{Y} = \mathbf{F}, \mathbf{Y} = \mathbf{G}, \mathbf{Y} \geq \mathbf{0}, \mathbf{W} \geq \mathbf{0} \tag{8}$$

The Augmented Lagrangian corresponding to Eq. (8) is given as:

$$\begin{aligned} \mathfrak{L}(\mathbf{E}, \mathbf{F}, \mathbf{G}, \mathbf{Y}, \mathbf{W}) = {} & \tfrac{1}{2}||\mathbf{X} - \mathbf{WY}||_F^2 + \frac{\lambda_1}{2}||\mathbf{E}||_F^2 + \lambda_2||\mathbf{F}||_* + \lambda_3\text{tr}(\mathbf{GL}_\text{T}\mathbf{G}^\top) \\ & + \langle \mathbf{\Phi}_1, \mathbf{Y} - \mathbf{E} \rangle + \langle \mathbf{\Phi}_2, \mathbf{Y} - \mathbf{F} \rangle + \langle \mathbf{\Phi}_3, \mathbf{Y} - \mathbf{G} \rangle \\ & + \tfrac{\beta}{2}(||\mathbf{Y} - \mathbf{E}||_F^2 + ||\mathbf{Y} - \mathbf{F}||_F^2 + ||\mathbf{Y} - \mathbf{G}||_F^2) \end{aligned} \tag{9}$$

2.4.1. Updating E:

The update expression for **E** is obtained by solving the following sub-problem.

$$\mathbf{E}^{[l+1]} = \underset{\mathbf{E}}{\operatorname{argmin}} \quad \lambda_1 \|\mathbf{E}\|_F^2 + \langle \mathbf{\Phi}_1, \mathbf{Y} - \mathbf{E} \rangle + \tfrac{\beta}{2} \|\mathbf{Y} - \mathbf{E}\|_F^2 \tag{10}$$

By differentiating Eq. (10) with respect to **E** and equating it to zero, the **E** update is given as,

$$\mathbf{E}^{[l+1]} = \frac{1}{\lambda_1 + \beta} \big(\mathbf{\Phi}_1^{[l]} + \beta \mathbf{Y}^{[l]} \big) \tag{11}$$

2.4.2. Updating F:

The **F** sub-problem is given as,

$$\mathbf{F}^{[l+1]} = \underset{\mathbf{F}}{\operatorname{argmin}} \quad \lambda_2 \|\mathbf{F}\|_* + \langle \phi_2, \mathbf{Y} - \mathbf{F} \rangle + \tfrac{\beta}{2} \|\mathbf{Y} - \mathbf{F}\|_F^2 \tag{12}$$

The update expression for **F** is found using the singular value thresholding (SVT) operator as follows [16],

$$\mathbf{F}^{[l+1]} = \mathbf{SVT}_{\frac{\lambda_2}{\beta}} \left[\mathbf{Y}^{[l]} + \frac{\mathbf{\Phi}_2^{[l]}}{\beta} \right] \tag{13}$$

2.4.3. Updating G:

The update expression for **G** is obtained by solving the following sub-problem.

$$\mathbf{G}^{[l+1]} = \underset{\mathbf{G}}{\operatorname{argmin}} \quad \lambda_3 \mathrm{tr}(\mathbf{G} \mathbf{L}_{\mathrm{T}} \mathbf{G}^\top) + \langle \mathbf{\Phi}_3, \mathbf{Y} - \mathbf{G} \rangle + \tfrac{\beta}{2} \|\mathbf{Y} - \mathbf{G}\|_F^2 \tag{14}$$

By differentiating Eq. (14) with respect to **G** and equating it to zero, the **G** update is given as,

$$\mathbf{G}^{[l+1]} = \big(\mathbf{\Phi}_3^{[l]} + \beta \mathbf{Y}^{[l]} \big) \big(\lambda_{\mathbf{3}} (\mathbf{L}_{\mathbf{T}} + \mathbf{L}_{\mathbf{T}}{}^\top) + \beta \mathbf{I} \big)^{\mathbf{-1}} \tag{15}$$

2.4.4. Updating Y:

By solving the following sub-problem, the update expression for **Y** can be obtained.

$$\begin{aligned} \mathbf{Y}^{[l+1]} = \underset{\mathbf{Y}}{\operatorname{argmin}} \quad & \frac{1}{2} \|\mathbf{X} - \mathbf{W}\mathbf{Y}\|_F^2 + \langle \mathbf{\Phi}_1, \mathbf{Y} - \mathbf{E} \rangle + \langle \mathbf{\Phi}_2, \mathbf{Y} - \mathbf{F} \rangle \\ & + \langle \mathbf{\Phi}_3, \mathbf{Y} - \mathbf{G} \rangle + \frac{\beta}{2} \big(\|\mathbf{Y} - \mathbf{E}\|_F^2 + \|\mathbf{Y} - \mathbf{F}\|_F^2 + \|\mathbf{Y} - \mathbf{G}\|_F^2 \big) \end{aligned} \tag{16}$$

By equating the gradient of Eq. (16) to zero, the update expression for **Y** is given as,

$$\begin{aligned} \mathbf{Y}^{[l+1]} = \Big[\big(\mathbf{W}^{[l]} \big)^\top \mathbf{W}^{[l]} + 3\beta \mathbf{I} \Big]^{-1} \Big[& \big(\mathbf{W}^{[l]} \big)^\top \mathbf{X}^{[l]} + \beta \big(\mathbf{E}^{[l+1]} + \mathbf{F}^{[l+1]} + \mathbf{G}^{[l+1]} \big) \\ & - \big(\mathbf{\Phi}_1^{[l]} + \mathbf{\Phi}_2^{[l]} + \mathbf{\Phi}_3^{[l]} \big) \Big] \end{aligned} \tag{17}$$

2.4.5. Updating W:

The $\mathbf{W}$ sub-problem is given as follows.

$$\mathbf{W}^{[l+1]} = \underset{\mathbf{W}}{\text{argmin}} \quad \frac{1}{2}||\mathbf{X} - \mathbf{W}\mathbf{Y}||_F^2 \tag{18}$$

Solution of the above equation can be found as,

$$\mathbf{W}^{[l+1]} = \left[\mathbf{X}^{[l]}\big(\mathbf{Y}^{[l+1]}\big)^{\top}\right]\left[\mathbf{Y}^{[l+1]}\big(\mathbf{Y}^{[l+1]}\big)^{\top}\right]^{-1} \tag{19}$$

Finally, the Lagrange multipliers are updated as follows:

$$\mathbf{\Phi}_1^{[l+1]} = \mathbf{\Phi}_1^{[l]} + \beta\big(\mathbf{Y}^{[l+1]} - \mathbf{E}^{[l+1]}\big) \tag{20}$$

$$\mathbf{\Phi}_2^{[l+1]} = \mathbf{\Phi}_2^{[l]} + \beta\big(\mathbf{Y}^{[l+1]} - \mathbf{F}^{[l+1]}\big) \tag{21}$$

$$\mathbf{\Phi}_3^{[l+1]} = \mathbf{\Phi}_3^{[l]} + \beta\big(\mathbf{Y}^{[l+1]} - \mathbf{G}^{[l+1]}\big) \tag{22}$$

Convergence of the algorithm is ensured if,

$$\max\left\{\begin{array}{ll} \left\|\mathbf{Y}^{[l+1]} - \mathbf{E}^{[l]}\right\|_\infty, & \left\|\mathbf{E}^{[l+1]} - \mathbf{E}^{[l]}\right\|_\infty \\ \left\|\mathbf{Y}^{[l+1]} - \mathbf{F}^{[l]}\right\|_\infty, & \left\|\mathbf{F}^{[l+1]} - \mathbf{F}^{[l]}\right\|_\infty \\ \left\|\mathbf{Y}^{[l+1]} - \mathbf{G}^{[l]}\right\|_\infty, & \left\|\mathbf{G}^{[l+1]} - \mathbf{G}^{[l]}\right\|_\infty \end{array}\right\} < \epsilon \tag{23}$$

The overall process involved in the proposed time series activity clustering algorithm is summarized in Algorithm 1.

Once the representation matrix $\mathbf{Y}$ is obtained, an affinity matrix $\mathbf{Q} \in \mathbb{R}^{k\times k}$ is calculated. The accuracy of clustering is highly dependent on the affinity matrix $\mathbf{Q}$. A usual approach in obtaining the affinity matrix is as shown below [14, 15].

$$\mathbf{Q} = \frac{|\mathbf{Y}| + |\mathbf{Y}^{\top}|}{2} \tag{24}$$

But, the graph so constructed do not take into account the intrinsic relationships of the within-cluster data points. But for data containing temporal information, the within-cluster data points are highly correlated. In order to take advantage of this information, an affinity matrix $\mathbf{Q}$ is calculated as follows.

$$\mathbf{Q}(i,j) = \frac{y_i^{\top} y_j}{||y_i||_2||y_j||_2} \tag{25}$$

where, $||.||_2$ represents the ℓ_2 norm operator.

Algorithm 1 Time Series Activity Clustering

```
Require: Skeletal data and parameters $\lambda_1$, $\lambda_2$, $\lambda_3$, $\eta$, $\gamma$ and $\beta$
Ensure: $\mathbf{Y} \in \mathbb{R}^{k \times k}$
Find $\mathbf{\Psi}$ using Eq. (1)
Find $\mathbf{X}$ using $\mathbf{\Psi}$
Generate matrices $\mathbf{Z}$, $\widetilde{\mathbf{W}}$, and $\mathbf{L_T}$
while not converged do
    Update $\mathbf{E}^{[l+1]}$ with Eq. (11)
    Update $\mathbf{F}^{[l+1]}$ with Eq. (13)
    Update $\mathbf{G}^{[l+1]}$ with Eq. (15)
    Update $\mathbf{Y}^{[l+1]}$ with Eq. (17)
    Update $\mathbf{W}^{[l+1]}$ with Eq. (19)
    Update $\mathbf{\Phi}_1^{[l+1]}$ with Eq. (20)
    Update $\mathbf{\Phi}_2^{[l+1]}$ with Eq. (21)
    Update $\mathbf{\Phi}_3^{[l+1]}$ with Eq. (22)
    Update $\beta^{[l+1]} = \eta\beta^{[l]}$
    $l = l + 1$
    Use (23) to check the convergence
end while
```

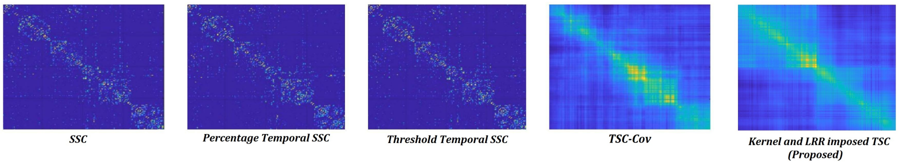
3. Experimental Results and Analysis

3.1. Dataset and Parameter Settings

To verify the performance of the proposed algorithm, it was tested on multiple human activity datasets. The datasets considered include the Gaming 3D (G3D) [22], Florence 3D (F3D) [23], UTKinect-Action 3D (UTK) [24], MSRC-Kinect12 (MSRC) [25], MSR Action 3D (MSRA) [26], and HDM14 [27] datasets. By means of observation, the parameters γ, λ_1, λ_2 and η are set to 5.2, 0.03, 18, and 0.7 respectively. The program development was done on a system with Intel Core i7 processor and a RAM of 16 GB, operating on 64-bit windows operating system with a clock frequency of 2.90 GHz.

3.2. Experimental Results

Experimental validation was done on the following methods.



SSC

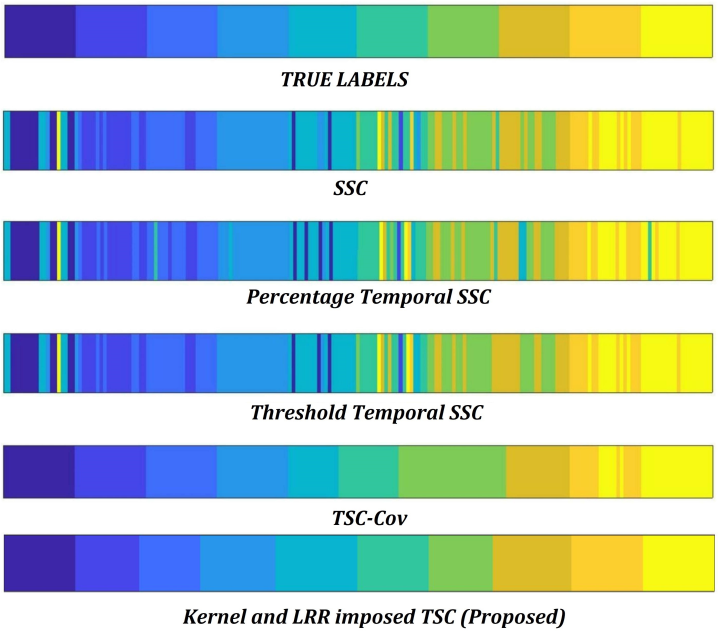
Percentage Temporal SSC

Threshold Temporal SSC

TSC-Cov

Kernel and LRR imposed TSC (Proposed)

Figure 2: Affinity graphs of SSC, Percentage Temporal SSC, Threshold Temporal SSC, TSC-Cov, and Kernel and LRR imposed TSC (Proposed) approaches on UTK Dataset.



TRUE LABELS

SSC

Percentage Temporal SSC

Threshold Temporal SSC

TSC-Cov

Kernel and LRR imposed TSC (Proposed)

Figure 3: Clustering Results on UTK Dataset

Subspace clustering approaches based on self representation: In this method, clustering using state-of-the-art clustering approaches are performed on the generated affinity matrix. The clustering methods considered include Spectral Clustering (SC) [28], Orthogonal Matching Pursuit (OMP) [29], K-means (Km) [28], SSC [14], and Elastic Net Subspace Clustering (EnSC) [29]. The results obtained using SSC is found to be superior to that of the counterparts [14].

SSC approaches with Data Pruning: In this method, the input skeletal data is pruned with strategies including min Φ [29], Temporal SSC [29], Threshold Temporal SSC [29], and Percentage Temporal SSC [29]. Later, a feature matrix is generated from the pruned data sequences, followed by the application of SSC [14] approach. This method converges quickly. Among others, the percentage temporal SSC approach gives better results.

TSC-Cov: This is a method that we had developed in one of our previous works. In this method, data re-framing was not performed and the kernel-based features were not incorporated in the covariance descriptor. But, we had used a new clustering approach named TSC-Cov for subspace clustering.

The performance of the proposed algorithm was evaluated against the above mentioned methods. The metrics used for quantitative evaluation include the accuracy, the Normalized Mutual Information (NMI), and the Adjusted Rand Index (ARI). The results obtained are tabulated in Tables (1-3).

For qualitative evaluation, the affinity matrix and clustering results obtained using the proposed method and the state-of-the-art methods are presented. Although the experiments were conducted on all the six datasets mentioned earlier, for the purpose of illustration, the results obtained with the UTK dataset [24] is given in this paper.

Table 1
Comparison of Clustering accuracy (%)

Dataset	**SSC** [14]	**Percentage Temporal SSC** [29]	**TSC-Cov**	**TSAC (Proposed)**
G3D [22]	65.16	66.04	90.04	**92.65**
F3D [23]	61.86	63.72	81.39	**81.58**
UTK [24]	74.37	72.36	89.44	**90.01**
MSRC [25]	73.42	80.60	81.09	**82.00**
MSRA [26]	58.35	60.86	89.76	**91.54**
HDM14 [27]	53.06	57.29	76.23	**79.51**

Table 2
Comparison of NMI

Dataset	**SSC** [14]	**Percentage Temporal SSC** [29]	**TSC-Cov**	**TSAC (Proposed)**
G3D [22]	0.719	0.708	0.953	**0.961**
F3D [23]	0.716	0.709	0.872	**0.875**
UTK [24]	0.709	0.672	0.899	**0.911**
MSRC [25]	0.720	0.762	0.887	**0.890**
MSRA [26]	0.700	0.720	0.958	**0.972**
HDM14 [27]	0.754	0.771	0.893	**0.901**

Table 3
Comparison of ARI

Dataset	**SSC** [14]	**Percentage Temporal SSC** [29]	**TSC-Cov**	**TSAC (Proposed)**
G3D [22]	0.499	0.479	0.847	**0.851**
F3D [23]	0.548	0.539	0.781	**0.787**
UTK [24]	0.547	0.499	0.804	**0.819**
MSRC [25]	0.551	0.714	0.773	**0.781**
MSRA [26]	0.435	0.456	0.881	**0.895**
HDM14 [27]	0.439	0.484	0.753	**0.772**

Fig. 2 visualizes the affinity matrices generated using SSC [14], Percentage Temporal SSC [29], Threshold Temporal SSC [29], TSC-Cov, and the proposed method while working on the UTK dataset. We can observe that among others, the affinity matrix generated using the proposed method have much denser block diagonal structure. This is an indication of quality of the clustering process.

Fig. 3 shows the clustering results obtained on the UTK dataset. For visual analysis, each cluster is assigned a unique color. The figure shows a comparison between SSC [14], Percentage Temporal SSC [29], Threshold Temporal SSC [29], TSC-Cov, and the proposed method with reference to the true labels. From Fig. 3 we can observe that clustering results obtained with the proposed method are comparatively better than the other methods.

4. Conclusions

The paper proposes a new method for clustering of human activity sequences in an efficient way. The proposed method involves the extraction of features from raw input data with the help of a kernel-based robust covariance descriptor. The optimization model developed uses the combined advantage of LRR and LSR based subspace clustering approaches. The concept of temporal Laplacian regularized dictionary learning is introduced in order to learn an effective representation matrix from the extracted data features. With the help of ADMM approach, the solution for the optimization problem is obtained. Performance of the proposed approach is compared with that of the state-of-the-art approaches in terms of accuracy, NMI, and ARI. Experimental results validate superiority of the proposed method in obtaining better clustering results as compared to that of the counterparts. Motion capture data often suffers from corruptions in the recorded information. To address this problem, robust human activity clustering algorithms can be developed in the future. Also, mocap information can be combined with other modalities of human activity data to achieve improved clustering results in challenging scenarios.

References

[1] P. Pareek, A. Thakkar, A survey on video-based human action recognition: recent updates, datasets, challenges, and applications, Artificial Intelligence Review 54 (2021) 2259–2322.

[2] H.-B. Zhang, Y.-X. Zhang, B. Zhong, Q. Lei, L. Yang, J.-X. Du, D.-S. Chen, A comprehensive survey of vision-based human action recognition methods, Sensors 19 (2019) 1005.

[3] C. Jobanputra, J. Bavishi, N. Doshi, Human activity recognition: A survey, Procedia Computer Science 155 (2019) 698–703.

[4] Y. Kong, Y. Fu, Human action recognition and prediction: A survey, International Journal of Computer Vision 130 (2022) 1366–1401.

[5] L. M. Dang, K. Min, H. Wang, M. J. Piran, C. H. Lee, H. Moon, Sensor-based and vision-based human activity recognition: A comprehensive survey, Pattern Recognition 108 (2020) 107561.

[6] S. K. Yadav, K. Tiwari, H. M. Pandey, S. A. Akbar, Skeleton-based human activity recognition using convlstm and guided feature learning, Soft Computing 26 (2022) 877–890.

[7] M. Barnachon, S. Bouakaz, B. Boufama, E. Guillou, Ongoing human action recognition with motion capture, Pattern Recognition 47 (2014) 238–247.

[8] S. Park, J. Park, M. Al-Masni, M. Al-Antari, M. Z. Uddin, T.-S. Kim, A depth camera-based human activity recognition via deep learning recurrent neural network for health and social care services, Procedia Computer Science 100 (2016) 78–84.

[9] J. K. Aggarwal, L. Xia, Human activity recognition from 3d data: A review, Pattern Recognition Letters 48 (2014) 70–80.

[10] M. Li, S. Chen, X. Chen, Y. Zhang, Y. Wang, Q. Tian, Symbiotic graph neural networks for 3d skeleton-based human action recognition and motion prediction, IEEE Tran. on PAMI (2021).

[11] A. Bagnall, J. Lines, A. Bostrom, J. Large, E. Keogh, The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances, Data Mining and Knowledge Discovery 31 (2017) 606–660.

[12] L. Parsons, E. Haque, H. Liu, Subspace clustering for high dimensional data: a review, Acm sigkdd explorations newsletter 6 (2004) 90–105.

[13] R. Vidal, P. Favaro, Low rank subspace clustering (lrsc), Pattern Recognition Letters 43 (2014) 47–61.

[14] E. Elhamifar, R. Vidal, Sparse subspace clustering: Algorithm, theory, and applications, IEEE Tran. on PAMI 35 (2013) 2765–2781.

[15] J. Xue, Y.-Q. Zhao, Y. Bu, W. Liao, J. C.-W. Chan, W. Philips, Spatial-spectral structured sparse low-rank representation for hyperspectral image super-resolution, IEEE Tran. on Image Processing 30 (2021) 3084–3097.

[16] J. Francis, A. Johnson, B. Madathil, S. N. George, A joint sparse and correlation induced subspace clustering method for segmentation of natural images, in: 2020 IEEE 17th India Council Int. Conf. (INDICON), 2020, pp. 1–7.

[17] Z. Wu, M. Yin, Y. Zhou, X. Fang, S. Xie, Robust spectral subspace clustering based on least square regression, Neural Processing Letters 48 (2018) 1359–1372.

[18] J. Cavazza, A. Zunino, M. S. Biagio, V. Murino, Kernelized covariance for action recognition, in: 2016 23rd Int. Conf. on Pattern Recognition (ICPR), 2016, pp. 408–413.

[19] Z. Zhang, K. Zhao, Low-rank matrix approximation with manifold regularization, IEEE Tran. on PAMI 35 (2013) 1717–1729.

[20] W. Liu, X. Ma, Y. Zhou, D. Tao, J. Cheng, p-laplacian regularization for scene recognition, IEEE Tran. on Cybernetics 49 (2019) 2927–2940.

[21] G. Casalino, N. D. Buono, C. Mencar, Part-based data analysis with masked non-negative matrix factorization, in: Int. Conf. on Computational Science and Its Applications, Springer, 2014, pp. 440–454.

[22] V. Bloom, V. Argyriou, D. Makris, Hierarchical transfer learning for online recognition of compound actions, Computer Vision and Image Understanding 144 (2015).

[23] L. Seidenari, V. Varano, S. Berretti, A. Del Bimbo, P. Pala, Recognizing actions from depth cameras as weakly aligned multi-part bag-of-poses, in: 2013 IEEE Conf. on CVPR - Workshops, 2013, pp. 479–485.

[24] L. Xia, C.-C. Chen, J. K. Aggarwal, View invariant human action recognition using histograms of 3d joints, in: 2012 IEEE Computer Society Conf.on CVPR - Workshops, 2012, pp. 20–27.

[25] S. Fothergill, H. M. M. , P. Kohli, S. Nowozin, Instructing people for training gestural interactive systems, in: CHI '12 Proc. of the SIGCHI Conf. on Human Factors in Computing

Systems, ACM, 2012, pp. 1737–1746.

[26] W. Li, Z. Zhang, Z. Liu, Action recognition based on a bag of 3d points, in: 2010 IEEE Computer Society Conf. on CVPR - Workshops, 2010, pp. 9–14.

[27] M. Müller, T. Röder, M. Clausen, B. Eberhardt, B. Krüger, A. Weber, Documentation Mocap Database HDM05, Technical Report CG-2007-2, Universität Bonn, 2007.

[28] Y. Lee, S. Choi, Minimum entropy, k-means, spectral clustering, in: 2004 IEEE Int. Joint Conf. on Neural Networks (IEEE Cat. No.04CH37541), volume 1, 2004, pp. 117–122.

[29] G. Paoletti, J. Cavazza, C. Beyan, A. Del Bue, Subspace clustering for action recognition with covariance representations and temporal pruning, in: 2020 25th Int. Conf. on Pattern Recognition (ICPR), 2021, pp. 6035–6042.