Consistency and Diversity induced Human Motion Segmentation

Tao Zhou, Member, IEEE, Huazhu Fu, Senior Member, IEEE, Chen Gong, Ling Shao, Fellow, IEEE, Fatih Porikli, Fellow, IEEE, Haibin Ling, Jianbing Shen, Senior Member, IEEE

Abstract—Subspace clustering is a classical technique that has been widely used for human motion segmentation and other related tasks. However, existing segmentation methods often cluster data without guidance from prior knowledge, resulting in unsatisfactory segmentation results. To this end, we propose a novel Consistency and Diversity induced human Motion Segmentation (CDMS) algorithm. Specifically, our model factorizes the source and target data into distinct multi-layer feature spaces, in which transfer subspace learning is conducted on different layers to capture multi-level information. A multi-mutual consistency learning strategy is carried out to reduce the domain gap between the source and target data. In this way, the domain-specific knowledge and domain-invariant properties can be explored simultaneously. Besides, a novel constraint based on the Hilbert Schmidt Independence Criterion (HSIC) is introduced to ensure the diversity of multi-level subspace representations, which enables the complementarity of multi-level representations to be explored to boost the transfer learning performance. Moreover, to preserve the temporal correlations, an enhanced graph regularizer is imposed on the learned representation coefficients and the multi-level representations of the source data. The proposed model can be efficiently solved using the Alternating Direction Method of Multipliers (ADMM) algorithm. Extensive experimental results on public human motion datasets demonstrate the effectiveness of our method against several state-of-the-art approaches.

Index Terms—Subspace clustering, human motion segmentation, transfer learning, multi-level representation.

1 INTRODUCTION

HUMAN motion segmentation has received widespread interest in industry and research communities due to its extensive applications in video retrieval, virtual reality, and smart surveillance for user interfaces and human action analysis [2], [3], [4]. The main goal of human motion segmentation is to partition temporal data sequences that depict human activities and actions into a set of non-overlapping and internally temporal segments. More importantly, it is often used as a preprocessing step before motion-related analytical tasks. Currently, there are several significant challenges for human motion segmentation, including the lack of well-defined activity fragments, and the ambiguity in motion primitives caused by temporal variability among different actions [5].

Human motion data contains temporal information, which is critical for motion segmentation. However, due to the high-dimensional structure of visual representations and complexity of temporal correlations, it is still challenging to capture such discriminative temporal information from time-series data [6]. Several algorithms have been proposed to address this challenge, including model-based [7], temporal proximity-based [6], and representation-based [8]. Moreover, since temporal data occurs in the form of consecutive frame samples, human motion segmentation can be treated as an unsupervised clustering task [9]. Among clustering-based motion segmentation methods, subspace clustering-based approaches [8], [10] have attracted increasing attention and obtained promising segmentation performance.

Subspace clustering is a popular technique to partition a...
Our model over other state-of-the-art approaches. Multiple benchmark datasets demonstrate the superiority of the ADMM algorithm. Experimental results on temporal correlations. Finally, we show that our model can feature representations of the source data set to preserve the imposition on the learned representation coefficients and further, an enhanced graph regularizer (HSIC) to enhance the diversity of the learned representation coefficients. We present a novel human motion segmentation algorithm, which integrates multi-level subspace learning and transfer learning into a unified framework. Our model aims to transfer knowledge from related source data to boost the performance of target data in the human motion segmentation task. To the best of our knowledge, this work is the first to develop multi-level subspace learning for human motion segmentation.

A deep NMF structure is built to capture the hidden information by leveraging the benefits of strong interpretability from the NMF model. Through this deep NMF structure, we obtain multi-level non-negative representations based on different dimensionalities of the dictionary atoms in multi-layer feature spaces, which reduce the distribution difference between the source and target domains.

Our model explores domain-invariant properties by using a multi-mutual consistency learning strategy while preserving domain-specific knowledge. The main advantage of our approach is to automatically balance the two aspects for enhancing the ability of transfer subspace learning.

We propose a novel constraint term to ensure that the learned multi-level subspace representation coefficients are diverse, which can help explore the complementary information from multi-level feature spaces to boost the segmentation performance.

This paper significantly extends our previous work in the conference paper [1], with multi-fold aspects. First, we reformulate a new multi-level transfer subspace learning framework with three key components, i.e., multi-mutual consistency learning, diversity cross multi-level representation, and temporal correlation preservation, for human motion segmentation. Second, we propose to utilize the HSIC as a diversity constraint for explicitly encouraging the learned multi-level representations to be of sufficient diversity, which can enhance complementary information to boost transfer subspace learning performance. Third, we provide deeper insight into the proposed multi-level transfer subspace learning framework. In this regard, our model is a general approach, as both hand-crafted HOG features and deep CNN features can be fed into our model. Last but not least, more experiments and additional ablation studies are carried out to further investigate the effectiveness of the proposed model and different key components.

The rest of this paper is arranged as follows. Related works, including subspace clustering, human motion analysis, and transfer learning, are briefly introduced in Section 2. We present the details of the proposed method in Section 3. We provide the experimental settings, experimental results, and model study in Section 4. Finally, we conclude this paper in Section 5.

2 Related Work

There are four types of works that are most related to the proposed human motion segmentation method, including 1) subspace clustering, 2) human motion analysis, 3) temporal data clustering, and 4) transfer learning.
Subspace clustering [15], [16], [17], [18], [19] holds the assumption that data points can be drawn from multiple subspaces corresponding to different clustering groups. Currently, self-representation based subspace clustering is increasing attention, in which each data point is expressed using a linear combination of other data points. For example, sparse subspace clustering (SSC) [20] aims to find the sparsest representation among the infinitely possible representations based on $\ell_1$-norm. Different from SSC, low-rank representation clustering (LRR) [21] attempts to uncover hidden structures with a low-rank representation. By introducing a graph regularizer, smooth representation clustering (SMR) [22] investigates the grouping effect of representation-based algorithms. Moreover, there are also several deep learning-based subspace clustering approaches [28], [29], [30], [32], [37]. However, these methods cannot be directly applied to motion segmentation since they do not consider the temporal correlations in successive frames.

2.2 Human Motion Analysis

Over the past decades, several methods have been developed for human motion analysis. For example, Zhong et al. [33] develop a bipartite graph co-clustering framework to segment unusual activities in video. Jenkins et al. [34] adopt the zero-velocity crossing points of the angular velocity to partition a stream of motion data into different sequences. Barbic et al. [35] develop to decompose human motion into multiple distinct actions using the probabilistic principal component analysis algorithm. Further, Beaudoin et al. [36] present a new framework to automatically distill a motion-motif graph from an arbitrary collection of motion data. Moreover, several clustering-based approaches have been proposed to segment a stream of human behavior into several activities [37].

2.3 Temporal Data Clustering

The goal of the temporal data clustering task is to segment data sequences into a set of non-overlapping parts. It has a wide range of applications, from facial analytics, and speech segmentation to human action recognition. To achieve this, a semi-Markov K-means clustering [38] model is often used to exploit repetitive patterns. For instance, Zhou et al. [3] develop a K-means kernel associated with a dynamic temporal alignment framework. Temporal subspace clustering (TSC) [8] method learns expressive coding coefficients based on a non-negative dictionary learning algorithm, which also introduces a temporal Laplacian regularization term to exploit the temporal correlations. Transfer subspace segmentation (TSS) [26] borrows knowledge from relevant source data to boost the target tasks. Low-rank transfer subspace (LTS) [27] method borrows some of transfer learning models [40], [41], [42], [43], [44], [45] have been developed and obtain promising performance. Among these methods, domain-invariant feature learning is a popular strategy [45] to learn a common feature space where both the domain shift and distribution difference can be mitigated. Several works, such as dictionary learning [48], [49] and subspace learning [46],
The alignment of two domains. Moreover, deep learning has been introduced to integrate knowledge transfer and feature learning into a unified framework [50], [51], [52], [53], [54], [55]. However, most of these methods conduct the domain alignment using high-level features from top layers, while ignoring the low-level structural information. Besides, they mainly focus on domain-invariant feature learning without preserving domain-specific knowledge.

3 THE PROPOSED METHOD

In this section, we first introduce the motivation (§ 3.1) and provide details of the proposed method (§ 3.2). Moreover, we describe the optimization steps for solving the proposed model (§ 3.3). Finally, we construct a new affinity matrix for clustering-based motion segmentation (§ 3.4), and provide complexity analysis (§ 3.5).

3.1 Motivation

As previously mentioned, three main challenges remain for human motion segmentation using transfer subspace learning, i.e., (1) how to capture multi-level information to enhance the performance of transfer subspace learning; (2) how to decrease the distribution gap between the two domains while also preserving domain-specific knowledge; and (3) how to effectively preserve temporal correlations among motion data. To address these challenges, we propose a novel multi-level transfer subspace learning framework with three key components for human motion segmentation, and the details of each component will be provided in the following subsections.

Moreover, deep structure learning has proven its effectiveness in several machine learning and computer vision applications [56], [57], [58], [59]. To effectively capture multi-level structural information, we adopt a multi-layer decomposition process based on the deep NMF model, which can be formulated as:

\[
X \approx D^{(1)}H^{(1)} \\
\approx D^{(1)}D^{(2)}H^{(2)} \\
\vdots \\
\approx D^{(1)}D^{(2)} \cdots D^{(l)} \cdots D^{(L)}H^{(L)},
\]

where \(D^{(l)} \geq 0\) and \(H^{(l)} \geq 0\) (\(l = 1, \ldots, L\)) represent the base matrix and feature representation in the \(l\)-th layer, respectively. And \(L\) denotes the number of layers. Through this deep NMF structure, we obtain multi-level non-negative representations based on different dimensionalities of the dictionary atoms, which can reduce the influence of the distribution gap between the two domains in a layer-wise fashion. It can be worth noting that the learned dictionary (base matrix) is not arbitrary in our NMF-based model with the non-negativity constraint.

3.2 Multi-level Transfer Subspace Learning

Existing subspace clustering or subspace clustering-based motion segmentation approaches tend to reconstruct original data points using either shallow representations (e.g., handcrafted features) or high-level representations (e.g., features from the last layer of deep networks). However, the original data often contains complex structural information and hierarchical semantics, which are difficult to extract by only using a single-layer clustering strategy. In Fig. 3, we show the clustering results when constructing the affinity matrix using the coefficients from the first layer, last layer, and multi-layer fusion, respectively. As can be seen, the orange part exists in the results when using the first and last layer representation methods, which indicates that the two methods generate multiple fragments and cannot achieve meaningful or accurate segmentation. While the method using multi-level representations can accurately segment the current motion action. Motivated by these results, we propose a multi-level subspace learning strategy to effectively exploit the hierarchical semantics and structural information in a layer-wise fashion, which can be formulated as follows:

\[
\begin{align*}
\mathcal{L} \left(X_s, X_t; D^{(1)}, D^{(2)}, H^{(1)}, H^{(2)}, Z^{(l)}\right) = \\
\|X_s - D^{(1)} \cdots D^{(L)}H^{(L)}\|^2_F + \|X_t - D^{(1)} \cdots D^{(L)}H^{(L)}\|^2_F \\
+ \sum_{l=1}^{L} \|H^{(l)} - H^{(l)}Z^{(l)}\|_2, 1,
\end{align*}
\]

Well, \(X_s \in \mathbb{R}^{d \times n_s}\), \(X_t \in \mathbb{R}^{d \times n_t}\) denote the source and target data, respectively. \(D^{(l)} \geq 0\) and \(H^{(l)} \geq 0\) (\(l = 1, \ldots, L\)) denote the basis matrix and the feature representation matrix at the \(l\)-th layer for the source data, respectively (similar for the target data). In addition, \(d\) is the original feature dimension, and \(n_s\) and \(n_t\) are the number of the source and target data, respectively. Here, we have \(H_t \in \mathbb{R}^{d \times n_t}\) and \(H_t \in \mathbb{R}^{d \times n_t}\) with \(d_l\) denoting the feature dimension in the \(l\)-th layer, we obtain \(Z_l \in \mathbb{R}^{n_s \times n_s + n_t}, l = 1, 2, \ldots, L\). Moreover, \(Z_l\) denotes a column vector whose elements are all set to one. The first two terms aim to explore the multi-level structures in both the source and target data, and the third term is used to conduct the multi-level transfer subspace learning.
In addition, in Eq. (2), the non-negative constraint $Z^{(l)} \geq 0$ enhances the discriminative ability of the learned representations. The constraint $1^\top Z^{(l)} = 1^\top$ ensures the sum of each coefficient vector to be one, resulting in the suppression of representation coefficients from different subspaces. In Eq. (2), the feature representations of the source data (i.e., $H^{(l)}_s$) are adopted as the subspace learning dictionary, which is used to reconstruct the feature representations of both the two domains (i.e., the source and target). By using this reconstruction strategy, it enables that knowledge from the related source data to be effectively transferred to the target task. Besides, $\| \cdot \|_{2,1}$ denotes the $\ell_{2,1}$-norm, which is used to constrain the columns of a matrix to be zero [21], i.e., $\| E \|_{2,1} = \sum_{i=1}^{N} \sqrt{\sum_{j=1}^{M} |E_{ij}|^2}$, where $E \in \mathbb{R}^{M \times N}$. By using the $\ell_{2,1}$-norm, there is an underlying assumption that some corruptions could be sample-specific, i.e., some data points may be corrupted while the others are clean.

Remarks: Since diverse information often exists in different domains, it is difficult to transfer useful knowledge from the related source to the target by only using a single subspace. Thus, we carry out subspace learning on multi-level feature spaces to obtain multiple subspace representations and then fuse them to seek an optimal affinity matrix. Moreover, the $\ell_{2,1}$-norm has been shown to be more robust to outliers than the classic Frobenius norm [16], [21].

### 3.2.1 Multi-Mutual Consistency Learning

To decrease the distribution gaps between the two domains as well as preserve the knowledge from different domains, we propose a multi-mutual consistency learning strategy as follows:

$$R_{1} = \sum_{l=1}^{L} F_{\text{con}} (D^{(l)}, D^{(l)}) ,$$

(3)

where the goal of the term $F_{\text{con}}(\cdot, \cdot)$ is to reduce the distribution gap between the two domains by penalizing the divergence of two basis matrices in different layers. In contrast, several methods directly project the original features from the source and target data into a common low-dimensional space using a domain-invariant projection matrix, which could result in a loss of an amount of domain-specific knowledge. Although there are different strategies to constrain the consistency between $D^{(l)}$ and $D^{(l)}$, a simple but effective strategy is adopted in our model, i.e., $F_{\text{con}}(D^{(l)}, D^{(l)}) = \| D^{(l)} - D^{(l)} \|_F^2$.

Remarks: The proposed multi-mutual consistency learning strategy can reduce the distribution between source and target domains. By using the strategy, our transfer subspace learning model can transfer useful information from the source domain to benefit the segmentation while also preserving some domain-specific knowledge.

### 3.2.2 Diversity across Multi-level Representation

In the formulation of Eq. (2), we have presented a multi-level subspace learning strategy to effectively exploit the structural information in a layer-wise fashion. Further, to enhance the complementarity of multi-level representations, the encoded representation coefficients (i.e., $Z^{(l)}$, $l = 1, 2, \ldots, L$) of different levels are encouraged to be of sufficient diversity. To achieve this, a diversity regularization term is proposed. For convenience, we define a mapping $\phi(x)$ from $x \in \mathcal{X}$ to kernel space $\mathcal{F}$, where $\phi(\cdot)$ is used to map original features into kernel space. In this kernel space, the inner product between vectors is given by a kernel function $k_1(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle$. Let $G$ be a second kernel space on $\mathcal{Y}$, and we define a mapping function $\phi(y)$ from $y \in \mathcal{Y}$ to $G$ with $k_2(y_i, y_j) = \langle \phi(y_i), \phi(y_j) \rangle$. Inspired by [60], we adopt an empirical version of HSIC to enhance the diversity of representation coefficients in the proposed model, which is given below.

**Definition 1.** Given a series of $n$ independent observations collected from $p_{xy}$, i.e., $Z = \{ (x_1, y_1), \ldots, (x_n, y_n) \} \subseteq \mathcal{X} \times \mathcal{Y}$, an estimator of $\text{HSIC}(Z, \mathcal{F}, \mathcal{G})$ is defined by

$$\text{HSIC}(Z, \mathcal{F}, \mathcal{G}) = (n - 1)^{-2} \text{tr}(K_1K_2H),$$

(4)

where $k_{1,ij} := k_1(x_i, x_j)$ and $k_{2,ij} := k_2(y_i, y_j)$. Here, $h_{ij} := \delta_{ij} - 1/n$ centralizes the two Gram matrices (i.e., $K_1$ and $K_2$) in the feature space, ensuring that they have zero mean. Please refer to the details of HSIC in [60], [61].

Thus, to explore more complementary information from different feature spaces, we encourage the multi-level representation coefficients $Z^{(l)}$ and $Z^{(m)}$ ($l \neq m$) to be sufficiently diverse. The diversity regularization term can be defined by

$$R_{2} = \sum_{l=1}^{L} \sum_{m \neq l} \text{HSIC} (Z^{(l)}, Z^{(m)}) .$$

(5)

**Remarks:** It is worth noting that the diversity across multi-level representations can help to explore more knowledge from source data, which could boost the performance of human motion segmentation.

### 3.2.3 Temporal Correlation Preservation

Temporal correlation is important for accurate clustering since human motion data is sequential and consecutive. Thus, it is expected to preserve the temporal information in representation coefficients $Z$ and feature representations $H$. To achieve this, we regulate the $i$-th coefficient’s neighbors $[z_{i-\tau/2}, \ldots, z_{i-1}, z_{i+1}, \ldots, z_{i+\tau/2}]$ to be close to $z_i$, where $\tau$ denotes the number of sequential neighbors. Besides, to learn an effective reconstruction dictionary (i.e., $X$, in our method), we also regulate the $i$-th coefficient’s neighbors $[h_{i-\tau/2}, \ldots, h_{i-1}, h_{i+1}, \ldots, h_{i+\tau/2}]$ to be close to $h_i$ in the source data. Thus, this enhanced graph regularizer is imposed on both the learned representation coefficients and feature representations of the source dataset. By using it, our model can effectively uncover the temporal correlations residing in both the source and target data. To achieve this, we first build a weight matrix $S$ [8], [26], where each element of $S$ is given as:

$$s_{ij} = \begin{cases} 1, & \text{if } |i - j| \leq \tau, l(x_i) = l(x_j), \text{for source data;} \\ 1, & \text{if } |i - j| \leq \tau, \text{for target data;} \\ 0, & \text{otherwise,} \end{cases}$$

(6)

where $l(x_i)$ denotes the action label of the $i$-th sample $x_i$ from the source data. After that, we have the temporal correlation preservation term as follows:

$$R_{3} = \sum_{l=1}^{L} \left( \text{tr}(Z^{(l)}LZ^{(l)}\top) + \text{tr}(H^{(l)}_sH^{(l)}_s\top) \right) ,$$

(7)

where $\text{tr}(\cdot)$ represents the matrix trace. $L$ denotes the Laplacian matrix with $L = C - S$, where $C$ is a diagonal degree
matrix with $C_i = \sum_j S_{ij}$, and $L_s$ is the corresponding part to the source data.

**Overall formulation:** Finally, we formulate the proposed CDMS model (in Eq. 2) with three key components (in Eqs. (3)(5)(7)) into a unified framework as follows:

$$
\min_{\Omega} \mathcal{L} + \alpha \mathcal{R}_1 + \beta \mathcal{R}_2 + \gamma \mathcal{R}_3
= \|X_s - D^{(1)} \ldots D^{(L)} H^{(L)}\|_F^2 + \|X_t - D^{(1)} \ldots D^{(L)} H^{(L)}\|_F^2
+ \sum_{l=1}^L \|H^{(l)}_t - H^{(l)}_s Z^{(l)}\|_{2,1} + \alpha \sum_{l=1}^L \|D^{(l)}_t - D^{(l)}_s\|_F^2
+ \beta \sum_{l=1}^L \sum_{m \neq l} \text{HSIC}(Z^{(l)}, Z^{(m)})
+ \frac{1}{2} \gamma \sum_{l=1}^L \left(\frac{\text{tr}(Z^{(l)} LX^{(l)}_s) + \text{tr}(H^{(l)}_t H^{(l)}_s^\top)}{\text{tr}(H^{(l)}_t H^{(l)}_s)}\right),
\text{s.t.} \quad Z^{(l)} \geq 0, 1^\top Z^{(l)} = 1^\top, \forall l = 1, 2, \ldots, L,
$$

(8)

where $\Omega = \{D^{(l)} \geq 0, D^{(l)}_s \geq 0, H^{(l)} \geq 0, H^{(l)}_s \geq 0, Z^{(l)}, W^{(l)}\}$ ($l = 1, 2, \ldots, L$) is the variable set to be optimized, and $\alpha$, $\beta$, and $\gamma$ are trade-off parameters. For clarity, the main notations used in this paper are listed in Table 1.

### 3.3 Optimization

Although the proposed objective function in Eq. (8) is not jointly convex for all variables, we can employ the ADMM algorithm to efficiently solve it. To adopt this strategy to our problem, we introduce two auxiliary variables $J^{(l)}$ and $E^{(l)}$ to replace $Z^{(l)}$ and $[H^{(l)}, H^{(l)}_s] - H^{(l)}_s Z^{(l)}$, respectively. Therefore, we have the following equivalent problem:

$$
\mathcal{L}(\Omega) = \|X_s - D^{(1)} \ldots D^{(L)} H^{(L)}\|_F^2 + \|X_t - D^{(1)} \ldots D^{(L)} H^{(L)}_s\|_F^2
+ \sum_{l=1}^L \|E^{(l)}\|_{2,1} + \alpha \sum_{l=1}^L \|D^{(l)} - D^{(l)}_s\|_F^2
+ \beta \sum_{l=1}^L \sum_{m \neq l} \text{HSIC}(Z^{(l)}, Z^{(m)})
+ \frac{1}{2} \gamma \sum_{l=1}^L \left(\frac{\text{tr}(Z^{(l)} LX^{(l)}_s) + \text{tr}(H^{(l)}_t H^{(l)}_s^\top)}{\text{tr}(H^{(l)}_t H^{(l)}_s)}\right)
+ \sum_{l=1}^L \Phi \left(A_t^{(l)}, H^{(l)}_t - H^{(l)}_s Z^{(l)} + J^{(l)} - E^{(l)}\right)
+ \sum_{l=1}^L \Phi \left(A_s^{(l)}, Z^{(l)} - J^{(l)}\right),
\text{s.t.} \quad Z^{(l)} \geq 0, 1^\top Z^{(l)} = 1^\top, \forall l = 1, 2, \ldots, L,
$$

(9)

where $\Phi(A, Q) = \frac{1}{2} \|Q\|_F^2 + \langle A, Q \rangle$, with $\langle \cdot, \cdot \rangle$ denoting the matrix inner product. $A_t^{(l)}$ and $A_s^{(l)}$ ($l = 1, 2, \ldots, L$) are Lagrangian multipliers, and $\mu$ is a penalty scalar. We describe the optimization steps for each subproblem below.

**$D_s$-subproblem:** The optimization problem with respect to $D_s$ is formulated as

$$
\min_{D_s \geq 0} \|X_s - D_s^{(1)} \ldots D_s^{(L)} H^{(L)}\|_F^2 + \alpha \sum_{l=1}^L \|D_s^{(l)} - D_s^{(l)}_s\|_F^2, \quad \forall l = 1, 2, \ldots, L.
$$

(10)

We can update $D_s^{(l)}$ in each layer one by one. By taking the derivative of Eq. (10) with respect to $D_s^{(l)}$ and using the Karush-Kuhn-Tucker (KKT) condition [63], we have the updating rule:

$$
D_s^{(l)} \leftarrow D_s^{(l)} \circ \frac{\Theta_s^{(l-1)} X_s H^{(L)}_s^\top + \alpha D_s^{(l)}}{\Theta_s^{(l-1)} + \alpha D_s^{(l)}},
$$

(11)

where $\circ$ denotes element-wise product, and $\Theta_s^{(l-1)} = D_s^{(1)\top} \ldots D_s^{(l-1)\top}$ and $\Omega_s^{(l+1)} = \Theta_s^{(l-1)} D_s^{(l)\top} \ldots D_s^{(L+1)\top}$.

**$H_s$-subproblem:** To update $H_s$, the following objective function should be optimized:

$$
\min_{H_s \geq 0} \|X_s - D_s^{(1)} \ldots D_s^{(L)} H^{(L)}_s\|_F^2 + \sum_{l=1}^L \Phi \left(A_t^{(l)}, H_t^{(l)} - H_s^{(l)} Z^{(l)} - E^{(l)}\right),
$$

(12)

where $\Theta_t^{(l)} = D_s^{(1)\top} \ldots D_s^{(l-1)\top}$ and $\Omega_t^{(l+1)} = \Theta_s^{(l-1)} D_s^{(l)\top} \ldots D_s^{(L+1)\top}$.

**$H_t$-subproblem:** To update $H_t$, the following objective function should be optimized:

$$
\min_{H_t \geq 0} \|X_t - D_t^{(1)} \ldots D_t^{(L)} H_t^{(L)}\|_F^2 + \frac{\mu}{2} \|H_t - H_s Z^{(l)} - E^{(l)} + A_t^{(l)\top} / \mu\|_F^2.
$$

(13)

It can be observed that the updates of $H_t^{(l)} (l = 1, \ldots, L)$ in each layer are independent. Thus, $H_t^{(l)}$ can be updated one by one. For $H_t^{(l)}$, we optimize the following problem:

$$
\min_{H_t^{(l)} \geq 0} \|X_t - \Theta_t^{(l)} H_t^{(l)}\|_F^2 + \frac{\mu}{2} \|H_t^{(l)} - H_s Z^{(l)} - E^{(l)} + A_t^{(l)\top} / \mu\|_F^2.
$$

(14)

Denoting $E^{(l)} = [E^{(l)}_t, E^{(l)}_s], Z^{(l)} = [Z^{(l)}_t, Z^{(l)}_s]$, and $A_t^{(l)} = [A_t^{(l)}_1, A_t^{(l)}_2]$, it is equivalent to optimizing

$$
\min_{H_t^{(l)} \geq 0} \|X_t - \Theta_t^{(l)} H_t^{(l)}\|_F^2 + \frac{\mu}{2} \|H_t^{(l)} - H_s Z^{(l)} - E^{(l)} + A_t^{(l)} / \mu\|_F^2.
$$

(15)

By taking the derivative of Eq. (15) w.r.t. $H_t^{(l)}$ and using the KKT condition [63], we have the updating rule:

$$
H_t^{(l)} \leftarrow H_t^{(l)} \circ \frac{2 \Theta_t^{(l)} X_t + \mu (E^{(l)} - A_t^{(l)\top} / \mu) (I - Z^{(l)} \circ I)}{2 \Theta_t^{(l)} X_t + \mu H_t^{(l)} (I - Z^{(l)} \circ I) - E^{(l)} + 2 \gamma H_t L_t},
$$

(16)

where I denotes an identity matrix.

**$L$-subproblem:** To update $L$, the following objective function should be optimized:

$$
\min_{L \geq 0} \|X_t - D_t^{(1)} \ldots D_t^{(L)} H_t^{(L)}\|_F^2 + \frac{\mu}{2} \|H_t^{(l)} - H_s Z^{(l)} - E^{(l)} + A_t^{(l)} / \mu\|_F^2.
$$

(17)

We also update $H_t^{(l)} (l = 1, \ldots, L)$ one by one. For $H_t^{(l)}$, it is equivalent to optimizing the following problem:

$$
\min_{H_t^{(l)} \geq 0} \|X_t - \Theta_t^{(l)} H_t^{(l)}\|_F^2 + \frac{\mu}{2} \|H_t^{(l)} - H_s Z^{(l)} - E^{(l)} + A_t^{(l)\top} / \mu\|_F^2.
$$

(18)
By taking the derivative of Eq. (18) w.r.t. $H_{l}^{(i)}$ and using the KKT condition [63], the updating rule is given as:

$$H_{l}^{(i)} \leftarrow H_{l}^{(i)} \ominus \frac{2\Theta_{l}^{(i)} X_{t} + \mu (H_{l}^{(i)} Z_{l}^{(i)} + E_{l}^{(i)} - A_{l}^{(i)}/\mu)}{2\Theta_{l}^{(i)} / \Theta_{l}^{(i)} H_{l}^{(i)} + \mu H_{l}^{(i)}}. \quad (19)$$

**J-subproblem:** Dropping the unrelated terms, the optimization for $Z^{(i)}$ yields

$$\min_{l} \Phi \left( A_{l}^{(i)}, [H_{l}^{(i)}, H_{l}^{(i)} - H_{l}^{(i)} J^{(i)} - E_{l}^{(i)}] \right) + \Phi (A_{l}^{(i)}, Z^{(i)} - J^{(i)}). \quad (20)$$

This is equivalent to optimizing the following problem:

$$\min_{l} \|H_{l}^{(i)}, H_{l}^{(i)} - A_{l}^{(i)}/\mu - E_{l}^{(i)} + H_{l}^{(i)} J^{(i)}\|_{F}^{2} + \|J^{(i)} - Z^{(i)} - A_{l}^{(i)}/\mu\|_{F}^{2}. \quad (21)$$

By taking the derivative of (21) w.r.t $J^{(i)}$ and setting it to zero, its closed-form solution is given as follows:

$$J^{(i)} = \left(H_{l}^{(i)\top} H_{l}^{(i)} + \mathbf{I}\right)^{-1}$$

$$H_{l}^{(i)\top} \left( [H_{l}^{(i)}, H_{l}^{(i)} + A_{l}^{(i)}/\mu, E_{l}^{(i)}] + (Z^{(i)} - A_{l}^{(i)}/\mu) \right). \quad (22)$$

**Z-subproblem:** By dropping the unrelated terms, we can optimize the following problem for updating $Z^{(i)}$:

$$\min_{z_{l}} \beta \sum_{m \neq l} \text{HSIC} (Z^{(i)}, Z^{(m)}) + \gamma \left( tr(J^{(i)} L Z^{(i)}) \right)$$

$$+ \Phi (A_{l}^{(i)}, Z^{(i)} - J^{(i)}), \quad \text{s.t.} \quad Z^{(i)} \geq 0, 1^{\top} Z^{(i)} = 1^{\top}. \quad (23)$$

In this study, we adopt the inner product kernel for the HSIC constraint, i.e., $K_{l} = Z^{(l)}$, $Z^{(l)}$. Then, we have $\text{HSIC}(Z^{(l)}, Z^{(m)}) = tr(Z^{(l)} K Z^{(m)})$ with $K = \sum_{m \neq l} MK_{m} M$, where $m_{ij} = \delta_{ij} - 1/n$. Finally, by taking the derivative of (23) w.r.t $Z^{(i)}$, we can obtain

$$\frac{\partial J}{\partial Z^{(i)}} = \beta Z^{(i)} K + Z^{(i)} L + \frac{\mu}{2} H_{s}^{(i)}\top H_{s}^{(i)} - A_{l}^{(i)}/\mu, H_{s}^{(i)}\top Z^{(i)} - H_{s}^{(i)}\top H_{s}^{(i)} + A_{l}^{(i)}/\mu - E_{l}^{(i)}). \quad (24)$$

By setting $\frac{\partial J}{\partial Z^{(i)}}$ to zero, it has its closed-form solution. After that, an iterative algorithm [64] is used to obtain the optimal solution of $Z^{(i)}$.

**E-subproblem:** Updating the error term $E^{(i)}$ is equivalent to solving the following problem:

$$\min_{E^{(i)}} \frac{1}{\mu} \|E^{(i)}\|_{2,1} + \frac{1}{2} \|E^{(i)} - G\|_{F}^{2}, \quad (25)$$

where $G = [H_{l}^{(i)}, H_{l}^{(i)}] - H_{l}^{(i)} Z_{l}^{(i)} + A_{l}^{(i)}/\mu$. This optimization problem is solved using the algorithm in [65].

**Multipliers updating:** The multipliers $A_{l}^{(i)}$ and $A_{l}^{(i)}$ ($l = 1, 2, \ldots, L$) can be updated by

$$\left\{ \begin{array}{l} A_{l}^{(i)} := A_{l}^{(i)} + \mu (H_{l}^{(i)}, H_{l}^{(i)} - H_{l}^{(i)} Z_{l}^{(i)} - E_{l}^{(i)}), \\ A_{l}^{(i)} := A_{l}^{(i)} + \mu (Z_{l}^{(i)} - J^{(i)}). \end{array} \right. \quad (26)$$

**Initialization and implementation:** Following previous deep NMF-based works [56, 57], we first pretrain a deep NMF model to obtain initial approximations for $D_{l}^{(i)}, D_{l}^{(i)}, H_{l}^{(i)}$, and $H_{l}^{(i)}$ ($l = 1, 2, \ldots, L$) in each layer. This pretraining process often reduces the training time of our model, and its effectiveness has also been proven in deep auto-encoder networks [66]. Let us consider the source data as an example. We decompose $X_{s} \approx D_{l}^{(i)} H_{l}^{(i)}$ and further decompose $H_{l}^{(i)} \approx D_{l}^{(i)} H_{l}^{(i)}$, until all layers are initialized. Then, we carry out the optimization steps for all variables and repeat this until convergence. The detailed steps for optimizing the proposed framework in Eq. (8) via the ADMM algorithm are summarized in Algorithm 1.

### 3.4 Clustering-based Motion Segmentation

With Algorithm 1, we can obtain the learned multi-level representations $Z^{(i)}$ ($l = 1, 2, \ldots, L$). Then, we extract the corresponding target representations $Z^{(i)} \in \mathbb{R}^{n_{s} \times n_{t}}$ from $Z^{(i)} = [Z^{(i)}_{l}, Z^{(i)}]$. Inspired by [8], we develop a new similarity measurement to construct an affinity matrix $A$ for our multi-level transfer subspace learning, which can explore intrinsic relationships among within-cluster samples from human motion data. Specifically, each element of $A$ can be defined as the distance between a pair of the learned representation coefficients for target data, which is given by

$$a_{ij} = \frac{1}{L} \sum_{l=1}^{L} \frac{z_{ij}^{(i)\top} Z_{l}^{(i)}}{\|z_{ij}^{(i)\top} Z_{l}^{(i)}\|_{2}}, \quad (27)$$

where $z_{ij}^{(i)}$ and $z_{ij}^{(i)}$ denote the $i$-th and $j$-th columns of $z_{ij}^{(i)}$, respectively. Note that we compute each element $a_{ij}$ by averaging the pair distance among multi-level feature spaces. After that, the Normalized Cuts [67] algorithm is applied to the learned affinity matrix $A$ to produce the clustering results.

### 3.5 Complexity Analysis

The overall optimization procedure consists of two stages, i.e., pretraining and model updating, thus we analyze the computational burden of the two parts separately. For clarity, we define $p$ as the maximal layer size in all layers and $n$ ($n = n_{s} + n_{t}$) as the total number of the source and target data. For the pretraining part, the computational complexity is of order $O(L_{p}(n_{s}p^{2} + n_{s}p^{2} + n_{t}p^{2} + n_{t}p^{2}))$, where $t_{p}$ is the number of iterations. For the model updating part, the
computational cost lies in updating $D_l^{(l)}, D_l^{(i)}, H_l^{(i)}, H_l^{(o)}, J_l^{(i)}, Z_l^{(i)},$ and $E_l^{(i)}$. Thus the computational complexity is of order $O((L_u(n_s^2p+n_s^2p^2+n_sp^3+p^3+n_s^2n_s+n_sn_p))$, where $L_u$ is the number of iterations in this stage. Finally, considering $n_s, n_r > p$ in our task, the overall computational cost is $O(L((t_p+L_u)(n_s^2p+n_sp^2+n_sp^2p+n_p^3)+t_u(n_s^3+n_sn_p)))$.

4 EXPERIMENTAL RESULTS AND ANALYSIS

In this section, we first introduce the human motion datasets used in our experiments (§ 4.1), and provide the details of the experimental setup (§ 4.2). Then we show the comparison results with several state-of-the-art methods (§ 4.3) and conduct the model study (§ 4.4).

4.1 Human Motion Datasets

To comprehensively evaluate the proposed model, we conduct experiments on five benchmark human motion datasets (see Fig. 4 for some example frames). • Keck Gesture Dataset (Keck) [68] consists of 14 different actions from military signals, in which each subject is carried out 14 actions and gestures. Besides, the videos in this dataset were obtained by a fixed camera when these subjects stand out in a static background. • Multi-Modal Action Detection Dataset (MAD) [69] consists of actions captured from various modalities using a Microsoft Kinect V2 system, which includes RGB images, depth cues, and skeleton formats. Specifically, the RGB images and 3D depth cues are of a size of $240 \times 320$. Moreover, each subject performs 35 different actions within two indoor scenes. • Weizmann Dataset (Weiz) [70] is composed of 90 video sequences with 10 actions (running, walking, skipping, bending, etc.) captured by nine subjects in an outdoor environment. All videos have a size of $180 \times 144$ with 50 fps. • UT-Interaction Dataset (UT) [71] is composed of 20 videos, each of which includes six different action types of human-human interactions (such as punching, pushing, pointing, hugging, kicking, and handshaking). • YouTube Dataset [72] consists of 11 action categories, which are diving, biking/cycling, horseback riding, golf swinging, soccer juggling, swinging, walking, trampoline jumping, volleyball spiking, tennis swinging, and basketball shooting. In our experiment, we construct four sequences, each of which includes the first 10 actions.

4.2 Experimental Setup

4.2.1 Dataset Settings

Following the dataset preprocessing and settings in [27], in our experiments, we use the extracted 324-dimensional HOG features [73] for each frame. Due to the fact that different datasets consist of different numbers of actions, we randomly choose ten motions from the Keck, MAD, and Weiz datasets in our experiments. In addition, we extract deep CNN features from the Keck, Weiz, and Youtube datasets using the pretrained VGG-16 [74] model and obtain a 1000-dimension feature vector for each frame. For model evaluation, we select one dataset as the source and the remaining datasets as the target, and we obtain the mean results of motion segmentation on the target dataset.

4.2.2 Compared Methods

We compare the proposed model with several state-of-the-art approaches, which include (1) Spectral clustering (SC) [12]. The feature vectors of target samples are directly fed into the standard spectral clustering algorithm [12] to obtain clustering results. (2) K-medoids (KMD) [75] selects target samples as centers and partitions them into different groups using the Manhattan Norm to measure the distance between data points. (3) Low-rank representation (LRR) [21] conducts subspace clustering by imposing a low-rank constraint on the target data. (4) Ordered subspace clustering (OSC) [76] utilizes a temporal correlation constraint to force representations of the temporal data to be closer. (5) Sparse subspace clustering (SSC) [20] assumes that there exists a dictionary that can be used to linearly represent all data points, and a sparse constraint is imposed on the representation coefficients. (6) Least square regression (LSR) [23] introduces the $\ell_2$ regularization on the learned coefficient matrix, in which the grouping effect of LSR enables to group mostly correlated data together. (7) Temporal subspace clustering (TSC) [8] employs a non-negative dictionary learning algorithm to obtain an expressive coefficient matrix for temporal clustering, and a temporal Laplacian regularization is imposed on the coefficient matrix. (8) Transfer subspace segmentation (TSS) [26] is a transferable subspace clustering method that explores useful information from relevant source data to boost clustering performance on target human motion data. (9) Low-rank transfer subspace (LTS) [27] employs a graph regularizer to capture temporal correlation in both the source and target data as well as uncovers clustering structures within data by introducing a weighted low-rank constraint. For convenience, we denote the preliminary of our method, proposed in [1], as MCSTL.

4.2.3 Evaluation Metrics and Parameter Settings

To evaluate the model performance, we utilize two popular metrics, including i.e., normalized mutual information (NMI) and accuracy (ACC). Specifically, NMI evaluates the mutual information between the ground truth and the recovered cluster labels, and ACC computes the classification score with the best map. Note that, higher values for both metrics indicate better performance. In addition, for our approach, we tune three trade-off parameters $\alpha$, $\beta$, and $\gamma$ in the range of $[10^{-5}, 10^{-4}, \ldots, 10^2]$. Furthermore, the number of layers for our CDMS model is set as 3, and the number of sequential neighbors is tuned from the set of $\{9, 11, \ldots, 21\}$.
### 4.3 Performance Comparison

In all comparisons, we set one sequence as the source and another one as the target. As we use four datasets for our evaluations, we report the segmentation results when testing on one dataset at one time, using the remaining three as the source domains. Besides, since SC, KMD, LRR, OSC, SSC, and LSR are not designed to utilize source information, we only employ target videos as input for these methods. For the TSC, TSS, and LRT models, both source and target videos are fed for segmentation. The clustering results of different methods using HOG features are shown in Table 2, where bold and underlined denote the best two results.

From the results in Table 2, we make the following observations: (1) Compared with SC, KMD, LRR, OSC, SSC, and LRR, our method transfers more useful information from the source data to learn distinctive representations of the target data, resulting in improved segmentation performance. (2) Compared with transfer clustering-based segmentation methods (including TSC, TSS, and TSS), our method also obtains much better performance. This is because our approach explores domain-invariant features and preserves domain-specific knowledge simultaneously. These two aspects are equally important for transfer learning. More importantly, our model fuses multi-level representations to construct the affinity matrix for motion segmentation, which effectively exploits hierarchical semantics and structural information in a layer-wise fashion. Thus, these results validate the effectiveness of the proposed method against other state-of-the-art models in human motion segmentation.

In Fig. 5, we visualize the clustering results obtained by our model and other state-of-the-art methods on a sample video of the Weiz dataset. Different colors denote different clusters in the action data. From the results, it can be observed that the LRR and SSC models generate multiple fragments and cannot achieve meaningful temporal segmentation. The main reason is likely that they do not consider the temporal correlations when conducting subspace clustering. Compared with the LRR and SSC methods, TSS obtains a relatively better segmentation but it still suffers from many unexpected fragments. LTS and TSS obtain much better performance in most cases; however, they occasionally generate some unexpected fragments. Overall, our methods (CDMS and MCTSL) achieve continuous and meaningful segments, providing much better segmentation results than...
other comparison methods.

**Performance on Deep Features.** As deep models have demonstrated promising performance in several computer vision tasks, we further evaluate the performance of the proposed model using deep features as inputs. In this study, deep features are extracted using the pretrained VGG-16 [74] model for each frame in the source and target datasets, and then we test the performance using the same settings as the previous experiments. We compare the proposed models (MCSTL and CDMS) with two state-of-the-art subspace transfer methods (i.e., TSS and LTS), and the comparison results are shown in Table 3. From this experiment, we have the following observations: (1) our MSTL model performs better than other comparison methods in most cases; (2) the models achieve similar or slightly better performance using deep features rather than HOG features. Moreover, it can be noted that there is a difference in computation cost between using the two types of features. When using deep features, it needs a large-scale dataset to train a pre-trained model, and then we can extract deep features as the inputs. The dimension of deep features is often high (e.g., 1000 in this study), which requires many computation resources. While using hand-crafted (e.g., HOG) features, it does not need extra training and requires fewer computation resources to process low-dimension features. Regardless of the input, our model is a general approach that can take hand-crafted features extracted from raw data or deep features.

### 4.4 Model Study

#### 4.4.1 Parameter Sensitivity

In our model, \( \tau \) denotes the number of sequential neighbors in the temporal correlation preservation term. To investigate the effect of this parameter, we test our model on the MAD dataset with Keck as the source for different values of \( \tau \). Fig. 6 shows the clustering performance for various values within \( \{3, 5, \ldots, 21\} \). From the results, it can be observed that our model obtains relatively better performance when \( \tau \in [15, 21] \). In addition, three key regularization parameters, i.e., \( \alpha, \beta \) and \( \gamma \), need to be manually tuned. To investigate the effects of the three parameters on the model output, we fix the value of one parameter and change the other two.

#### 4.4.2 Convergence Analysis

We compute the errors (i.e., \( \|H^{(l)}(I) - H^{(l)}(I)\|_\infty \) and \( \|W^{(l)}(I) - J^{(l)}\|_\infty \)) to demonstrate the convergence of our optimization algorithm. We report the mean values of different layers, and the convergence curves on the MAD dataset (Keck as source) are presented in Fig. 7 (d). Note that, for better presentation, the errors are normalized into the range \([0, 1]\). As can be observed, our model converges within about 50 iterations.

#### 4.4.3 Source Data Analysis

To evaluate the effectiveness of the source information for boosting the segmentation performance, we carry out an
experiment on the MAD dataset (with Keck as the source) in which different numbers of actions are used. The results are shown in Fig. 9 (a). As can be observed, the performance increases in general when the number of actions increases. This suggests that diverse source data is helpful for boosting the segmentation performance in the target domain. The main reason could be that more actions in the source data can transfer more useful knowledge to ensure that our model learns the distinctive representations of the target data. In addition, we also test the effect of using different numbers of each action. To achieve this, we utilize the frames with different ratios \((i.e., 0.1, 0.2, \cdots, 1)\) in each source action, while keeping the total number of actions fixed. We evaluate the performance on the MAD dataset, and the comparison results are shown in Fig. 9 (b). From the results, it can be observed that the performance of our model increases when the ratio of frames increases. This demonstrates that using more frames of each action in the source data can transfer useful knowledge to learn distinctive and effective representations of the target data. Thus, the effectiveness of the source data in improving the segmentation performance can be well validated.

It can be also noted that the results of our method with different source datasets are similar. If it needs to select a good source dataset to further improve the segmentation performance on the target data, we suggest selecting one source dataset that has a similar scenario to the target dataset. For example, we can obtain the best performance on MAD by using Keck as the source, since all these two datasets are under an indoor scenario.

Moreover, we have conducted an experiment by using two source datasets, and the comparison results are shown in Table 4. From the results, comparing using Weiz and Weiz & Keck as the sources, our method obtains similar performance in the two settings. Comparing using UT and UT & Weiz as the sources, our method gets a slightly better performance when using two datasets as sources than using single. Overall, our model can obtain similar and slightly better performance via using two datasets as the source than a single source, but the computation cost will increase when using more datasets. Thus, to balance the performance and computation cost, we just use one dataset as the source in this study. Additionally, our model is able to flexibly explore the complementarity among multi-level feature spaces, which can also be extended to other multi-modal learning tasks. In future work, we believe that the data size can be reduced by performing sampling techniques or replacing the original data set with a small number of points \([77],[78]\), and binary representations \([79]\) can also be considered to accelerate the computation speed. Moreover, another potential strategy is finding the most diverse and representative frames based on clustering methods, and then we can only use these frames as the source data for reducing the complexity.

Table 4: Comparisons on MAD when using one and two datasets as the source.

| Source Datasets | ACC   | NMI  |
|-----------------|-------|------|
| Weiz (source)   | 0.6392| 0.8238|
| Weiz + Keck (source) | 0.6345| 0.8325|
| UT (source)     | 0.6371| 0.8238|
| UT + Weiz (source) | 0.6440| 0.8289|

4.4.4 Dictionary Dimensionality Analysis

In the proposed model, we adopt a multi-layer deep matrix factorization structure, thus we need to set different dimensions for dictionary atoms at each layer. To investigate the effects of dimensionality of the dictionary atoms, we conduct a comparison experiment using different dimension sets. Specifically, we set ten cases in the dimension set, \(i.e., \{128-32;128-16;64-32;64-16;32-16;128-64-16;128-32-16;64-32-16;128-64-32-16\}\). Fig. 10 shows the segmentation results obtained based on using different numbers of layers and dimensionalities of the dictionary atoms on MAD using Keck as the source data. From the results, the ACC results vary when using different dimensionalities of the dictionary atoms, while the NMI results are robust to dimensionalities of the dictionary atoms. Thus, in this study, we adopt a three-layer deep NMF structure \(i.e., 128-64-16\) for all comparisons as the default setting.

4.4.5 Ablation Study

Effectiveness of Multi-level Representations: To validate the effectiveness of fusing multi-level subspace representations from different feature spaces, we conduct an experiment to test the performance of our model when using the representations from the first layer, last layer, and multiple fused layers on the Weiz dataset. The results are shown in
5 Conclusion

We have proposed a novel CDMS framework for human motion segmentation. Our model first factorizes the original features of the source and target data into implicit multi-layer feature spaces, in which we carry out transfer subspace learning on different layers to capture multi-level structural information. A multi-mutual consistency learning strategy is carried out to reduce the distribution gap between the source and target domains. Further, we introduce a novel constraint term based on the HSIC to strengthen the diversity of multi-level subspace representations, which enables the complementarity of multi-level representations to be explored in order to boost the transfer learning performance. Moreover, we develop an enhanced graph regularizer term to preserve the temporal correlations. Finally, we obtain the optimized affinity matrix by fusing the multi-level subspace representation coefficients. Extensive experimental results on benchmark datasets demonstrate the effectiveness of the proposed model against several state-of-the-art human motion segmentation methods.

References

[1] T. Zhou, H. Fu, C. Gong, J. Shen, L. Shao, and F. Porikli, “Multi-mutual consistency induced transfer subspace learning for human motion segmentation,” in Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit., 2020.
[2] R. Poppe, “Vision-based human motion analysis: An overview,” Comput. Vis. Image Unders., vol. 108, no. 1-2, pp. 4–18, 2007.
[3] F. Zhou, F. De la Torre, and J. K. Hodgins, “Hierarchical aligned cluster analysis for temporal clustering of human motion,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 35, no. 3, pp. 582–596, 2012.
[4] M. Keuper, S. Tang, B. Andres, T. Brox, and B. Schiele, “Hierarchical aligned cluster analysis for temporal clustering of human motion,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 42, no. 1, pp. 140–153, 2018.
[5] Jonathan Feng-Shun Lin, Michelle Karg, and Dana Kulić, “Movement primitive segmentation for human motion modeling: A framework for analysis,” IEEE Trans. Man-Mach. Syst., vol. 46, no. 3, pp. 325–339, 2016.
[6] E. Keogh and S. Kasetty, “On the need for time series data mining benchmarks: a survey and empirical demonstration,” Data Min. Knowl. Discov., vol. 7, no. 4, pp. 349–371, 2003.
[7] Y. Xiong and D.-Y. Yeung, “Mixtures of arma models for model-based time series clustering,” in Proc. IEEE Int. Conf. Data Min., IEEE, pp. 717–720, 2002.
[8] S. Li, K. Li, and Y. Fu, “Temporal subspace clustering for human motion segmentation,” in Proc. IEEE Int. Conf. Comput. Vis., pp. 4453–4461, 2015.
[9] F. Zhou, F. De la Torre, and J. K. Hodgins, “Aligned cluster analysis for temporal segmentation of human motion,” in Proc. IEEE Int. Conf. Automatic Face & Gesture Recognit., pp. 1–7, 2008.
[10] G. Xia, H. Sun, L. Feng, G. Zhang, and Y. Liu, “Human motion segmentation via robust kernel sparse subspace clustering,” IEEE Trans. Image Process., vol. 27, no. 1, pp. 135–150, 2017.
[11] X. Cao, C. Zhang, C. Zhou, H. Fu, and H. Foroosh, “Constrained multi-view video face clustering,” IEEE Trans. Image Process., vol. 24, no. 11, pp. 4381–4393, 2015.
[12] A. Y. Ng, M. I. Jordan, and Y. Weiss, “On spectral clustering: Analysis and an algorithm,” in Proc. Adv. Neural Inf. Process. Syst., pp. 849–856, 2002.
[13] J. P. Costeira and T. Kanade, “A multibody factorization method for independently moving objects,” Int. J. Comput. Vis., vol. 29, no. 3, pp. 159–179, 1998.
[14] L. Lu and R. Vidal, “Combined central and subspace clustering for computer vision applications,” in Proc. Int. Conf. Mach. Learn., pp. 593–600, 2006.
[15] R. Vidal, “Subspace clustering,” IEEE Trans. Signal Process., vol. 58, no. 5, pp. 1557–1574, 2010.
[16] Z. Zhai, D. Zhu, F. Hu, Z. Zhang, and X. Cao, “Latent multi-view subspace clustering,” Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit., pp. 4279–4287, 2017.
[17] T. Zhou, C. Zhang, X. Peng, H. Bhaskar and J. Yang, “Dual shared-specific multiview subspace clustering,” IEEE Trans. Cybern., vol. 50, no. 8, pp. 3517–3530, 2019.
[18] J. Yang, J. Liang, K. Wang, P. L. Rosin and M.-H. Yang, “Subspace clustering via good neighbors,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 42, no. 6, pp. 1537–1544, 2019.
[69] D. Huang, S. Yao, Y. Wang, and F. De La Torre, “Sequential max-margin event detectors,” in Proc. Eur. Conf. Comput. Vis., pp. 410–424, 2014.

[70] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri, “Actions as space-time shapes,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 29, no. 12, pp. 2247–2253, 2007.

[71] M. S. Ryoo and J. K. Aggarwal, “Spatio-temporal relationship match: video structure comparison for recognition of complex human activities,” in Proc. IEEE Int. Conf. Comput. Vis., pp. 1593–1600, 2009.

[72] J. Liu, Y. Yang, I. Saleemi, and M. Shah, “Learning semantic features for action recognition via diffusion maps,” in Comput. Vis. Image Underst., vol. 116, no. 3, pp. 361–377, 2012.

[73] N. Danal and B. Triggs, “Histogram of oriented gradients for human detection,” in Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit., pp. 886–893, 2005.

[74] K. Simonyan, and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in Proc. Int. Conf. Learn. Representations, 2015.

[75] L. Rddusseeun and P.-J. Kaufman, “Clustering by means of medoids,” 1987.

[76] S. Tierney, J. Gao, and Y. Guo, “Subspace clustering for sequential data,” in Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit., pp. 1019–1026, 2014.

[77] X. Chen and D. Cai, “Large scale spectral clustering with landmark-based representation,” in Proc. AAAI Conf. Artif. Intell., pp. 313–318, 2011.

[78] N. Tremblay and A. Loukas, “Approximating spectral clustering via sampling: a review,” in Sampling Techniques for Supervised or Unsupervised Tasks, pp. 129–183, 2020.

[79] Z. Zhang, L. Liu, F. Shen, H. T. Shen, and L. Shao, “Binary multi-view clustering,” in IEEE Trans. Pattern Anal. Mach. Intell., vol. 41, no. 7, pp. 1774–1782, 2018.