Sparse Canonical Temporal Alignment With Deep Tensor Decomposition for Action Recognition

Chengcheng Jia, Ming Shao, *Member, IEEE*, and Yun Fu, *Senior Member, IEEE*

***Abstract*— In this paper, we solve three problems in action recognition: sub-action, multi-subject, and multi-modality, by reducing the diversity of intra-class samples. The main stage contains canonical temporal alignment and key frames selection. As we know, temporal alignment aims to reduce the diversity of intra-class samples; however, dense frames may yield misalignment or overlapped alignment and decrease recognition performance. To overcome this problem, we propose a sparse canonical temporal alignment (SCTA) method, which selects and aligns key frames from two sequences to reduce diversity. To extract better features from the key frames, we propose a deep non-negative tensor factorization (DNTF) method to find a tensor subspace integrated with SCTA scheme. First, we model an action sequence as a third-order tensor with spatiotemporal structure. Then, we design a DNTF scheme to find a tensor subspace in both spatial and temporal directions. Particularly, in the first layer, the original tensor is decomposed into two low-rank tensors by NTF, and in the second layer, each low-rank tensor is further decomposed by tensor-train for time efficiency. Finally, our framework composed of SCTA and DNTF could solve the three problems and extract effective features for action recognition. Experiments on synthetic data, MSRDailyActivity3D, and MSRActionPairs data sets show that our method works better than competitive methods in terms of accuracy.**

***Index Terms*— Sparse canonical temporal alignment, key frames, deep non-negative tensor factorization, tensor-train.**

I. Introduction

HUMAN action recognition in realistic scenarios has attracted an increasing amount of attention in recent years and contemporary developments have shown promising performance even with complex backgrounds [1]–[3].

Manuscript received January 24, 2016; revised June 20, 2016 and September 27, 2016; accepted September 27, 2016. Date of publication October 25, 2016; date of current version December 9, 2016. This work was supported in part by NSF IIS under Award 1651902, in part by NSF CNS under Award 1314484, in part by ONR under Award N00014-12-1-1028, in part by ONR Young Investigator under Award N00014-14-1-0484, and in part by the U.S. Army Research Office Young Investigator under Award W911NF-14-1-0218. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Xiaochun Cao.

C. Jia is with the Department of Electrical and Computer Engineering, Northeastern University, Boston, MA 02115 USA (e-mail: cjia@coe.neu.edu).

M. Shao is with the Department of Computer and Information Science, College of Engineering, University of Massachusetts Dartmouth, North Dartmouth, MA 02747 USA (e-mail: mshao@umassd.edu).

Y. Fu is with the Department of Electrical and Computer Engineering, College of Engineering and the College of Computer and Information Science, Northeastern University, Boston, MA 02115 USA (e-mail: yunfu@ece.neu.edu).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org., provided by the author. The material includes software and instructions for Canonical Time Warping (CTW) and Generalized Time Warping (GTW). The total size of the file is 0.97 MB. Contact jia.ch@husky.neu.edu for further questions about this work.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2016.2621664

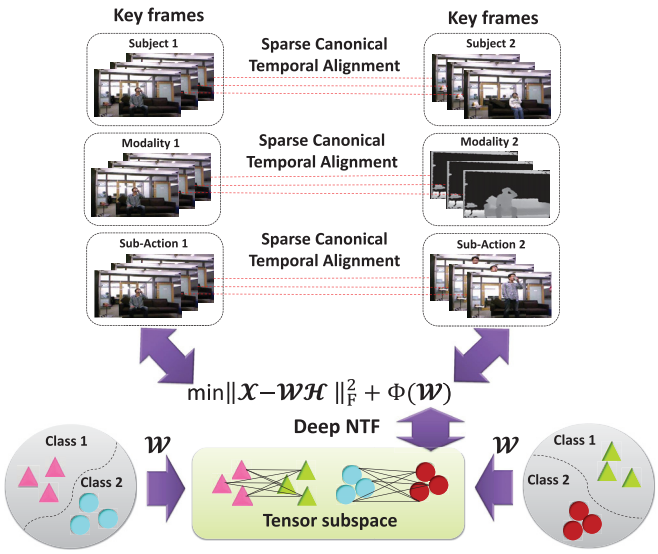


Fig. 1. Illustration of the proposed SCTA framework. Suppose $\mathcal{X}$ is the video dataset which can be decomposed into two low-rank tensors: $\mathcal{W}$, $\mathcal{H}$, where $\mathcal{W}$ is low-dimensional projection tensor, and $\mathcal{H}$ is the corresponding coefficients. The sparse key frames selected by SCTA in the tensor space make sure that two intra-class video sequences are well-aligned without irrelevant variance, which is able to further boost the action recognition performance in the tensor subspace sought out by our deep NTF mechanism.

To mitigate the impacts of noise and diversity, key frames are extracted to better describe actions in a video, meanwhile to alleviate the high-dimensional problem [4]–[6]. Key frames are sufficiently informative to represent action videos, and are usually obtained by clustering [7] or based on shots [8] containing the first, middle and last frame. However, there is an unexplored problem of action recognition that all the frames may contain intra-class variance due to sub-action (different scales), multi-subject and multi-modality shown in Fig. 1. This may severely affect the accuracy of action recognition.

Temporal alignment of two action sequences can alleviate the intra-class variance, and therefore address the multi-view, multi-subject, and multi-modality problems above [9]–[11]. To name a few: coupling two sequences with trajectories [12], [13], aligning two motions by Dynamic Time Warping (DTW) [14], warping different sequences dynamically on a manifold with spatial information [15], aligning action and facial sequences via Canonical Temporal Warping (CTW) [9], and proposing a probabilistic CTW with extra annotations [16]. However, these methods consider neither selecting key frames from a temporal series nor addressing different variances of the same action. To the best of our knowledge, how to jointly select key frames

and mitigate the intra-class action variance among different scales (e.g., stand or sit to drink), different subjects and different modalities (e.g., RGB and depth data) is still unclear. To reduce the diversity of intra-class samples, we perform temporal alignment only on the key frames, which guides the learning process of a discriminant tensor subspace for recognition.

We, inspired by the facts above and the flexible representation of tensor structure, propose a tensor based generic *Sparse Canonical Temporal Alignment* (SCTA) approach for action recognition, shown in Fig. 1. We aim to solve three challenges caused by the diversity of intra-class samples through SCTA and discriminant tensor subspace learning, where SCTA includes two components: (1) key frames selection and (2) spatiotemporal alignment. Key frames represent a temporal sequence well and could be obtained through sparse learning, which compiles unique or limited reconstructions of a dataset [17]–[19]. Meanwhile, temporal alignment of two sequences aims to reduce the intra-class diversity and is usually optimized by Canonical Correlation Analysis (CCA) [20], which is used in CTW [9].

Tensor representation has been explored in recent years for human action representation [22], [23], where a tensor is a multi-dimensional array. In an action video, the first and second directions (modes) of a tensor indicate the row and column of a frame, and the third mode conveys temporal knowledge. Tensor representation can preserve the spatiotemporal structure of an action video, and overcome the "curse of dimensionality" problem [24] through the learned discriminant subspaces. Considering tensor representation preserves the spatiotemporal structure of data, we develop a novel Deep Non-negative Tensor Factorization (DNTF) along with the SCTA to find the discriminant tensor subspace. Since considerable redundant spatiotemporal information exists in action videos, we employ low-rank decomposition to obtain a more concise representation of a tensor structure, which contains two main parts. First, taking the positive property of the real-world data into account, Non-negative Tensor Factorization (NTF) is introduced to achieve the goal of low-rankness as well as positive coefficient values. Second, the action features are further refined in a deep structure with NTF in the first layer, followed by a Tensor-Train (TT) decomposition in the second layer, which can be learned in an efficient manner. The deep structure of decomposition could eliminate unexpected factors, such as intra-class diversity, as we progressively demonstrated in our tensor scheme.

Our framework is able to tackle three problems: sub-action, multi-subject, and multi-modality by key frame alignment in a new tensor subspace. Extensive experiments on a synthetic dataset, MSRDailyActivity3D action, and MSRActionPairs action datasets show that our method works better than competitive methods. Our contributions are threefold:

- SCTA framework is proposed to tackle three challenges: sub-action, multi-subject and multi-modality in action recognition, due to intra-class diversity.
- Key frames of pairwise sequences are extracted in a sparse canonical correlation analysis fashion. Our algorithm encourages zero values on weight vectors and maintains sparse non-zero values for key frames, which automatically selects appropriate key frames from pairwise intra-class sequences.
- A DNTF scheme is designed to find the discriminant tensor subspace from a deep structure including NTF and TT building blocks. The designed structure not only ensures a low-rank tensor decomposition with positive values, but also significantly reduces the time complexity.

The rest of paper is organized as the followings. In Section II, we review relevant works of key frames selection, temporal alignment and tensor subspace learning. Second, we highlight the motivation of this paper in Section III. Then, we introduce the details of SCTA and our DNTF model in Section IV and Section V. We illustrate the temporal alignment results on both synthetic and real-world datasets in Section VI before drawing conclusions in Section VII.

II. Related Work

In this section, we briefly review related action recognition/alignment methods in three lines: (1) key frames selection, (2) temporal alignment, and (3) tensor subspace learning.

Key frames selection is able to describe action sequences regardless of noise as it rules out irrelevant frames subject to diverse impact factors. Assa et al. [25] extracted the key poses from a skeleton sequence via an affinity matrix. Zhao and Elgammal [26] utilized the bag-of-words model to select neighbor key frames. Junejo et al. [14] extracted trajectories and calculated the Self-Similarity Matrix (SSM) for measurement sequences. Vijayanarasimhan and Grauman [27] selected key frames based on optical flows from the whole sequence. Most recently, Liu et al. [28] extracted optical flow of key frames via Adaboost and calculated co-occurrence probability of all the frames for action recognition. Different from theirs, in this work, we extract sparse key frames from a pair of action sequences for joint temporal alignment and action recognition.

Temporal alignment is promising in tackling multi-view, multi-subject and multi-modality problems [10]. Recently, it has sparked research attention in action sequences and facial expression sequence alignment. Rao et al. [12] and Gritai et al. [13] aligned the trajectories of different videos. Junejo et al. [14] adopted DTW to synchronize multi-view human actions. Wang and Mahadevan [29] solved manifold alignment by analyzing a subspace and preserving the local geometry. Zhou and De la Torre [9] proposed a CTW framework to align sequences according to both spatial and temporal correspondence, and to address multi-modal and multi-dimensional problems. Compared to these works, our method tackles not only the multi-subject and multi-modality problems, but also an unexplored sub-action problem related to different motion scales, e.g., stand or sit to drink. In addition, the selected key frames by our model are able to boost the action recognition performance, which will be demonstrated in the experimental section.

Tensor structure for action recognition has attracted lots of attention recently, as it can represent spatiotemporal information in a natural way. Considering local geometry of

action series, Lui [30] presented the action series as a *third-order* tensor on the Riemann manifold, and calculated the log-distance of two samples on the tangent space. It should be noted that it does not explicitly seek for a common subspace and therefore fails to adapt to unseen datasets. Jia et al. [22] proposed a tensor subspace learning method by transferring depth information from the well-established source domain to the incomplete target domain to improve the performance of missing modality recognition. To explore the positive properties of data, NTF is proposed to find a subspace for face detection [31], [32] and pose recognition [33]. Recently, inspired by deep structure to extract features [34], [35], deep semi non-negative matrix factorization (deep semi-NMF) [36], [37] is proposed for multi-view face recognition with negative values as hidden features. Different from their work, we design a novel DNTF scheme composed of NTF and TT layers, which runs faster than conventional Tucker decomposition while obtaining positive feature interpretation in the hidden layers for more realistic data.

This paper is based on our previous work [21], which proposes an SCTA framework based on key frames selection and temporal alignment to solve the three challenges in action recognition. Compared to [21], we have three improvements in this paper: (1) a DNTF mechanism is proposed for extracting features; (2) more experiments are added to evaluate the DNTF framework under the three challenges; (3) extra parameters such as signal-to-noise ratio and time complexity are analyzed.

III. Motivation of Our Work

A. Three Challenges in Action Recognition

1) Sub-Action Problem: There are some shape variations in the same class, for example, drinking action when people are standing or sitting on sofa. Considering this partial variation, we represent an action sequence as a hierarchical structure including a common part and an individual part called sub-action, and we aim to mitigate the diversity by taking individual part into account.

2) Multi-Subject Problem: Different people perform the same action in different manners, such as velocity and motion scale. We aim to reduce the variations between different people, and maximize the coherence of the same action.

3) Multi-Modality Problem: Different modalities may help to improve performance as a complement to each other. We employ *RGB* and *depth* data in the multi-modality setting.

STCA is proposed to solve these problems, including two main parts: temporal alignment and sparse learning. Temporal alignment is usually performed by CCA [20] to find the similar frames of two sequences and reduce the intra-class diversity. Different from that, our STCA is similar to Sparse CCA (SCCA), which selects related elements and discards others from two sequences. Sparse learning is employed in our model with two merits: (1) selecting key frames using non-zero weights, and (2) obtaining unique or limited reconstructions of data after regression.

Our framework integrates STCA with DNTF to eliminate unexpected factors such as intra-class diversity in a two-layer decomposition scheme.In the first layer, NTF is performed to obtain a low-rank dictionary and a data representation. In the second layer, TT is used to eliminate redundancy of the dictionary and data representation. Particularly, if there are some other factors such as illumination or view angle in an action dataset, DNTF could remove their unexpected effects on the result of recognition in a deep decomposition manner.

B. Three Scenarios in Action Recognition

Our model is designed to solve the three challenges mentioned above, by key frame selection and temporal alignment. On the other hand, we also set different scenarios to see different influences of key frame selection or temporal alignment on action recognition.

Scenario 1 (S_1): Neither key frames selection nor temporal alignment in our model.

Scenario 2 (S_2): Temporal alignment is adopted but no key frames selection.

Scenario 3 (S_3): Both key frames selection and temporal alignment are performed.

IV. Sparse Canonical Temporal Alignment

In this section, we use temporal alignment to discover key frames from two videos, and only use these key frames for discriminant tensor subspace learning. To that end, we first introduce the concept of Canonical Temporal Alignment (CTA), from which we develop SCTA.

Given two intra-class action sequences with label $l \in \{1, \ldots, L\}$, they are represented as two *third-order* tensors $\mathcal{X}_s, \mathcal{X}_t \in \mathbb{R}^{r \times c \times f}$, where r, c and f indicate the dimensions of row, column of a frame and number of frames, respectively. The corresponding *mode-3* unfolding matrices are $X_s, X_t \in \mathbb{R}^{f \times (rc)}$. Then, the objective function of CTA for two sequences X_s, X_t can be written as:

$$\min_{\substack{A_s, A_t, \\ W_s, W_t}} \|A_s X_s W_s - A_t X_t W_t\|_{\mathrm{F}}^2 + \Phi(A_s, A_t, W_s, W_t), \quad (1)$$

where $A_s, A_t \in \mathbb{R}^{f \times f}$ warp two sequences in temporal domain and $\Phi(A_s, A_t, W_s, W_t)$ is the additional regularizer w.r.t. warping functions A_s, A_t and projection matrices W_s, W_t. A combination of DTW and CCA is used for temporal alignment in [9], which updates one variable when fixing others.

However, in CTA, the alignment results that provide the correspondence between frames from two sequences may not be necessary for high-level tasks such as action recognition. As indicated by the previous work, sparse key frames from the video sequence will work better [38]. Therefore, in this section, we propose an SCTA framework that pursues sparse correspondences between two video sequences. To that end, we introduce the column-wise sparse constraint $\|\cdot\|_{2,1}$ to the Eq. (1):

$$\begin{aligned} \min_{A_s, A_t, W} \; & \lambda_1 \|A_s X_s W - A_t X_t W\|_{\mathrm{F}}^2 \\ & + \lambda_2 \|A_s\|_{2,1} + \lambda_3 \|A_t\|_{2,1} + \Phi(A_s, A_t, W), \end{aligned} \quad (2)$$

where λ_p $(p = 1, 2, 3)$ is a penalty factor. In our new framework, we seek for a *common discriminant* tensor space

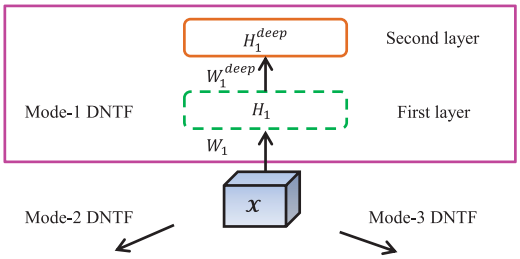


Fig. 2. Schematic illustration of *mode-1* DNTF. $\mathcal{X}$ is a *third-order* tensor in the first layer of DNTF. Two matrices W_1 and H_1 are obtained by *mode-1* NTF, followed by TT decompositions in the second layer to obtain W_1^{deep} and H_1^{deep}.

span W instead of two separated projections, and the introduced constraints on A_s and A_t aid in selecting important key frames from two sequences. We will detail the formulation of $\Phi(A_s, A_t, W)$ and the solutions of Eq. (2) in Section V.

V. Deep Non-Negative Tensor Factorization

In this section, we propose a DNTF method for high-dimensional action data decomposition, including the first NTF layer and the second TT layer. In this way, the deep structure can represent the multi-linear features with reasonable non-negative properties, while disentangling different challenges for discriminant feature learning in an efficient manner. We take *mode-1* DNTF as an example to illustrate this idea in Fig. 2. In our current two-layer structure, we have NTF in the first layer and TT in the second. In the first layer, our NTF model integrates with both CTA and sparse modeling, which is significantly different from the traditional NTF method. This step is very critical in finding key frames. Without the key frame selection in the first step, our method may still suffer from intra-class variations, similar to existing methods. The second layer TT is used to decompose W_1 and H_1 and compute the updated W_1^{deep} and H_1^{deep} with a lower rank. Then, we iterate the two steps until convergence. Next, we will introduce NTF and TT first, then give our objective function and solution, with time complexity analysis. Also, we compare our model with both the subspace alignment model and the temporal alignment model theoretically.

A. Non-Negative Tensor Factorization (NTF)

Conventional tensor decomposition methods including the Tucker decomposition [39] or CANDECOMP/PARAFAC (CP) decomposition [40] can obtain low-rank tensor structure which is useful for vision problems, such as human action analysis [22], [41], human brain image recovery and texture synthesis [42]. Considering the positive properties of action video representations [43], we propose to use NTF in the *first layer* of our deep structure.

Given an action dataset of m videos with L class labels represented by a *fourth-order* tensor $\mathcal{X} \in \mathbb{R}^{r\times c\times f\times m}$, we aim to find the decomposition $\mathcal{X} = \mathcal{W}\mathcal{H}$ by the following objective [32], [44]:

$$\arg\min_{\mathcal{W},\mathcal{H}} \|\mathcal{X} - \mathcal{W}\mathcal{H}\|_{\mathrm{F}}^2, \tag{3}$$

where $\mathcal{W}$ indicates the projection tensor, $\mathcal{H}$ indicates the reduced dimensional tensor, and $\mathcal{W} \geq 0$, $\mathcal{H} \geq 0$. The solution is obtained through two steps: (1) we perform *mode-n* unfolding of $\mathcal{X}$ to obtain matrix $X^{(n)}$, (2) NMF is employed to obtain *mode-n* projection matrix W_n and dimensionality reduced sample H_n. Accordingly, Eq. (3) is rewritten as:

$$\arg\min_{W_n,H_n} \|X^{(n)} - W_n H_n\|_{\mathrm{F}}^2, \tag{4}$$

and W_n and H_n are updated by:

$$W_n^{ij} \leftarrow W_n^{ij} \cdot \frac{(X^{(n)} H_n^{\mathrm{T}})^{ij}}{(W_n H_n H_n^{\mathrm{T}})^{ij}},$$
$$H_n^{ij} \leftarrow H_n^{ij} \cdot \frac{(W_n^{\mathrm{T}} X^{(n)})^{ij}}{(W_n^{\mathrm{T}} W_n H_n)^{ij}}, \tag{5}$$

where W_n^{ij} (H_n^{ij}) is an element of $W_n(H_n)$, and i, j indicates row and column respectively. According to the Tucker decomposition, the interaction of $\mathcal{W}$ and $\mathcal{H}$ is represented as:

$$\mathcal{W}\mathcal{H} = \mathcal{H} \otimes W_1 \otimes \ldots \otimes W_n \otimes \ldots W_N, \tag{6}$$

where $W_n \in \mathbb{R}^{I_n\times J_n}$ is the *mode-n* projection matrix, $\mathcal{H} \in \mathbb{R}^{J_1\times\ldots\times J_n\times\ldots\times J_N}$ is core tensor of $\mathcal{X}$, $I_n \in \{r, c, f\}$ is original feature dimension, and J_n is the reduced dimension in the tensor space. Note that in our problem, $N = 3$.

In addition, to better align pairwise intra-class neighbors, the action labels of of training samples are taken as *prior knowledge* to construct a graph in the manifold. We construct a pairwise graph $S \in \mathbb{R}^{m\times m}$ with the discriminant information, whose element can be defined as:

$$S_{ij} = \begin{cases} 1, & k \text{ nearest-neighbors of the same class;} \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

Here S is used to align pairwise sequences from the same class. Finally, Eq. (3) can be rewritten as:

$$\arg\min_{\mathcal{W},\mathcal{H}} \|(\mathcal{X} - \mathcal{W}\mathcal{H})S\|_{\mathrm{F}}^2, \tag{8}$$

where S performs on *mode-4* of $\mathcal{X}$ and $\mathcal{H}$. Next, we will introduce the TT decomposition to disentangle the hidden factors in $\mathcal{W}$ and $\mathcal{H}$.

B. TT Decomposition

Our deep mechanism aims to further find spatiotemporal factors and more precise representations of data. TT decomposition is used for the second layer of our DNTF model for its efficiency property explained in Section V-D. The execution of TT decomposition includes: (1) decompose feature representation $\mathcal{H}$ for different factors (spatial and temporal), (2) decompose classifier (projection tensor) $\mathcal{W}$ to reduce its dimensions, which is inspired by TensorNet [45].

Given an *N-order* tensor $\mathcal{A} \in \mathbb{R}^{I_1\times\ldots\times I_n\times\ldots\times I_N}$ where I_n is the dimension of *mode-n*, the TT format is written as follows:

$$\mathcal{A}(i_1, \ldots, i_N) = \mathcal{G}_1(\gamma_0, i_1, \gamma_1) \ldots \mathcal{G}_N(\gamma_{N-1}, i_N, \gamma_N), \tag{9}$$

where $\mathcal{G}_n(\gamma_{n-1}, i_n, \gamma_n)$ is an element of tensor core $\mathcal{G}_n \in \mathbb{R}^{r_{n-1}\times I_n\times r_{n+1}}$, r_n is *mode-n* rank, γ_n and i_n are *mode-n* auxiliary indices, and $r_0 = r_N = 1 (1 < n < N)$. Fig. 3 shows

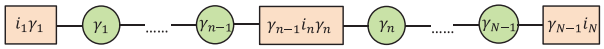


Fig. 3. Illustration of TT decomposition. Note rectangles indicate tensor cores, while the circles indicate auxiliary indices.

the TT format, the circles contain the auxiliary indices γ_{n-1} and γ_n which connect two cores $\mathcal{G}_{n-1}$ and $\mathcal{G}_n$ in the rectangles. The TT means we have to multiply all the elements of small core tensors and sum over all the indices. TT decomposition is fast compared with common tensor decomposition, e.g., the Tucker decomposition due to no recursion therein [46].

Our *mode-1* DNTF is illustrated in Fig. 2. For the *fourth-order* tensor $\mathcal{X}$, we decompose it in the first layer using NTF as $\mathcal{X} = \mathcal{W}\mathcal{H}$, where $\mathcal{W}$ is a transformation matrix and $\mathcal{H}$ is the feature representation. In the second layer, we apply TT decomposition on $\mathcal{H}$ and $\mathcal{W}$ as:

$$\mathcal{X} = \mathcal{W}\mathcal{H} = \overbrace{\mathcal{W}\prod_{n=1}^{N} U_n\mathcal{G}_n}^{\text{TT decomposition on } \mathcal{H}} = \underbrace{\mathcal{W}'\mathcal{G}}_{\text{TT decomposition on } \mathcal{W}'} = \prod_{n=1}^{N}\mathcal{C}_n\mathcal{G} = \mathcal{C}\mathcal{G}, \quad (10)$$

where $\mathcal{W}' = \mathcal{W}\prod_{n=1}^{N} U_n$, $\mathcal{G} = \prod_{n=1}^{N}\mathcal{G}_n$, $\mathcal{C} = \prod_{n=1}^{N}\mathcal{C}_n$, $N = 4$.

In DNTF, first, $\mathcal{H}$ is TT decomposed to obtain *mode-n* core $\mathcal{G}_n$ and matrix U_n, which contains spatiotemporal factors. Similar with deep semi-NMF [36] model, we integrate $\mathcal{W}$ and U_n to be new $\mathcal{W}'$, which contains spatial and temporal factors drawn from $\mathcal{H}$. Second, $\mathcal{W}'$ is TT decomposed to obtain new *mode-n* core $\mathcal{C}_n$, which is similar with TensorNet model [45] to obtain low-rank transformation. Finally we perform $\mathcal{W} \leftarrow \mathcal{C}$ and $\mathcal{H} \leftarrow \mathcal{G}$ in the second layer of our model. We perform NTF in the first layer on *mode-1* unfolding matrix $X^{(1)}$ to get two low-rank matrices W_1 and H_1, i.e., $X^{(1)} = W_1H_1$. Then W_1 and H_1 are further decomposed by TT in the second layer, i.e., $X^{(1)} = W_1H_1 \overset{\text{TT}}{=} W_1U_1G_1 = W_1'G_1 \overset{\text{TT}}{=} C_1G_1$. Finally we perform $W_1^{deep} \leftarrow C_1$ and $H_1^{deep} \leftarrow G_1$.

In the deep structure, the first layer NTF integrates with both canonical temporal alignment and sparse modeling, which is significantly different from the traditional NTF method suffering from intra-class variations. The second layer TT is used to decompose $\mathcal{W}$ and $\mathcal{H}$ further to a lower rank. We then iterate the two steps until convergence. In the future, additional decomposition could contribute to the deep model in the third or fourth layer, such as the Tucker decomposition or CP for other purposes. Next we will introduce our objective function and DNTF scheme in details.

C. Objective Function and Solutions

In the given dataset $\mathcal{X} \in \mathbb{R}^{r\times c\times f\times m}$ containing L class labels, $\mathcal{X}_s, \mathcal{X}_t \in \mathbb{R}^{r\times c\times f}$ are the s,t-th samples with the same label l ($l \le L$). We decompose $\mathcal{X}$ by Eqs. (3)$\sim$(4) to obtain *mode-n* projection matrix W_n ($n = 1, 2, 3$), and s,t-th low-dimensional intra-class samples are obtained by $\mathcal{D}_{s/t} = \mathcal{X}_{s/t} \times_1 W_1^{-1} \times_2 W_2^{-1} \times_3 W_3^{-1}$. $\mathcal{D}_{s/t}$ are *mode-3* unfolded to be $D_{s/t} \in \mathbb{R}^{J_3\times(J_1J_2)}$, where each row indicates one frame of the sequence. Considering spatial decomposition in Eq. (8) and temporal alignment in Eq. (1), our objective function is formulated as:

$$\begin{aligned}\min_{\mathcal{W},\mathcal{H}} &\|(\mathcal{X} - \mathcal{W}\mathcal{H})S\|_{\text{F}}^2 \\ &+ \sum_{l=1}^{L}\sum_{s,t\in l}\lambda_1\|A_sD_s - A_tD_t\|_{\text{F}}^2 + \lambda_2\|A_s\|_{2,1} + \lambda_3\|A_t\|_{2,1}, \\ &\text{s.t. } A_sD_sD_s^{\text{T}}A_s^{\text{T}} = \mathbf{I},\ A_tD_tD_t^{\text{T}}A_t^{\text{T}} = \mathbf{I}, \end{aligned} \quad (11)$$

where $A_s, A_t \in \mathbb{R}^{J_3\times J_3}$, $\mathbf{I} \in \mathbb{R}^{J_3\times J_3}$ is an identity matrix, and $\lambda_p (p = 1, 2, 3)$ is the penalty coefficient of each item. The constrains keep the solution non-trivial. In our objective function, the first item finds the subspace by performing NTF when spatial features have non-negative values in practice. The second item aligns the two series of key frames by CTA to handle sub-action, multi-subject and multi-modality problems. The third and fourth items are used to select the key frames of intra-class samples by sparse constraint to eliminate temporal redundancy. Next we introduce solutions to the function by jointly optimizing deep non-negative factorization and temporal sparse weight allocation.

As the learning problem in Eq. (11) is not jointly convex over all unknown variables, we propose to use the Lagrange Multiplier method [47] to optimize: $\mathcal{W}$, $\mathcal{H}$, A_s and A_t. Let us first write down the Lagrange Multiplier function:

$$\begin{aligned}F = &\|(\mathcal{X} - \mathcal{W}\mathcal{H})S\|_{\text{F}}^2 \\ &+ \sum_{l=1}^{L}\sum_{s,t\in l}\lambda_1\|A_sD_s - A_tD_t\|_{\text{F}}^2 + \lambda_2\|A_s\|_{2,1} + \lambda_3\|A_t\|_{2,1} \\ &+ \text{tr}\left(Y_1(A_sD_sD_s^{\text{T}}A_s^{\text{T}} - \mathbf{I})\right) + \text{tr}\left(Y_2(A_tD_tD_t^{\text{T}}A_t^{\text{T}} - \mathbf{I})\right), \end{aligned} \quad (12)$$

where Y_1 and Y_2 are Lagrangian multipliers, and all the variables are optimized iteratively. Next the solution is detailed in our two-layer decomposition framework.

1) First Layer of DNTF: The first order gradients of F with respect to different variables equal to 0, including: *mode-n* projection matrix W_n, low-dimensional sample H_n and warp matrices of temporal direction A_s, A_t.

Update W_n:

$$W_n^{ij} \leftarrow W_n^{ij} \cdot \frac{(X^{(n)}SS^{\text{T}}H_n^{\text{T}})^{ij}}{(W_nH_nSS^{\text{T}}H_n^{\text{T}})^{ij}}, \quad (13)$$

where i, j indicates the row and column of W_n.

Update H_n:

$$H_n^{ij} \leftarrow H_n^{ij} \cdot \frac{(X^{(n)}SW_n^{\text{T}}S^{\text{T}})^{ij}}{(W_nH_nSW_n^{\text{T}}S^{\text{T}})^{ij}}. \quad (14)$$

Update A_s:

$$A_s^{ij} \leftarrow A_s^{ij} \cdot \frac{\left((\mathbf{I} + \frac{Y_1 + Y_1^{\text{T}}}{\lambda_1})^{-1}A_tD_tD_s^{\text{T}} - \lambda_2\|A_s\|_{2,1}\right)^{ij}}{(D_sD_s^{\text{T}})^{ij}}, \quad (15)$$

Algorithm 1 SCTA (Solving Problem Eq. (11))

Input: $\mathcal{X}, \lambda_1, \lambda_2, \lambda_3$

Initialize: $A_s = A_t = \mathbf{0}$.

while not converged **do**

for *Mode-n* alternation

1. First layer of DNTF and update W_n^{ij}, H_n^{ij}, A_s^{ij} and A_t^{ij} by Eqs. (13) $\sim$ (16).
2. TT decomposition by Eq. (10).
3. Second layer of DNTF and update W_n^{ij}, H_n^{ij}, A_s^{ij} and A_t^{ij} by Eqs. (13) $\sim$ (16).
4. Check the convergence conditions $\|\mathcal{X} - \mathcal{WH}\|_2 < \epsilon$.

end for

end while

Output: W_n, H_n.

Update A_t:

$$A_t^{ij} \leftarrow A_t^{ij} \cdot \frac{\left((\mathbf{I} + \frac{Y_2 + Y_2^{\mathrm{T}}}{\lambda_1})^{-1} A_s D_s D_t^{\mathrm{T}} - \lambda_3 \|A_t\|_{2,1}\right)^{ij}}{(D_t D_t^{\mathrm{T}})^{ij}}. \tag{16}$$

2) Second Layer of DNTF: We update W_n and H_n by TT decomposition analyzed in Eq. (10).

Update W_n: As $\mathcal{W}' = \prod_{n=1}^{N} \mathcal{C}_n$, we perform $W_n \leftarrow C_n$, then Eq. (13) is used to update W_n.

Update H_n: As $\mathcal{H} = \prod_{n=1}^{N} U_n \mathcal{G}_n$, we perform $H_n \leftarrow G_n$, then Eq. (14) is used to update H_n.

Update A_s and A_t: A_s and A_t are updated by Eq. (15) and (16). The updated W_n, H_n, A_s and A_t are taken as initial inputs of the first layer in an iterative manner, which is shown in Algorithm 1.

D. Time Complexity Analysis

Given an *N-order* tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times \ldots \times I_n \times \ldots \times I_N}$ where I_n and r_n are the *mode-n* dimension and rank, we discuss the time complexity of the key decomposition steps. For simplicity, we skip the subscript, i.e., $I_n \rightarrow I$ and $r_n \rightarrow r$. We mainly compare TT decomposition in our DNTF model with Tucker decomposition, which takes $O(NI^N r)$ operations. For a TT decomposition, each core $\mathcal{G}_n(r_{n-1}, I_n, r_{n+1})$ is unfolded to be a matrix $G_n \in \mathbb{R}^{(Ir)\times r}$ through Single Value Decomposition (SVD) needs and will take $O(Ir^3)$ operations for each mode. Therefore, there are in total $O(NIr^3)$ steps for the TT decomposition of $\mathcal{A}$. We can see that Tucker takes much more time than the TT decomposition when $N \gg 3$.

E. Model Comparison

The most related works to our model include: 1) Generalized canonical Time Warping (GTW) [10] for temporal alignment and 2) Subspace Alignment model (SA) [48] for recognition. We set two sequences $D_s, D_t \in \mathbb{R}^{f \times (rc)}$ and warping matrices $A_s, A_t \in \mathbb{R}^{f \times f}$ as Eq. (1) defines.

1) SA Model: The state-of-the-art subspace alignment methods are usually used for domain adaption, e.g., SA aligns the subspaces of two domains. Given source domain data D_s and target domain data D_t, first PCA is performed on both domains to find two subspaces P_S and P_T, then SA aligns the two subspaces by:

$$T_S \leftarrow D_s(P_S P_S^{\mathrm{T}} P_T), \quad T_T \leftarrow D_t P_T, \tag{17}$$

where T_S and T_T are the transformed data whose distances are measured in a new subspace.

Different from SA, our model aligns intra-class samples distributed in two domains element-by-element, particularly, frame-by-frame in the action sequences. Additionally, we find one shared subspace for both domains instead of two, by a DNTF mechanism:

$$\min_{\mathcal{W},\mathcal{H}} \|\mathcal{X} - \mathcal{WH}\|_{\mathrm{F}}^2 \; S + \sum_{c=1}^{C} \sum_{s,t=1}^{m} \lambda_1 \|A_s D_s - A_t D_t\|_{\mathrm{F}}^2, \tag{18}$$

where $\mathcal{W}$ is used for find a new common tensor subspace, and the second term is the frame-by-frame alignment of intra-class samples in two domains.

2) Temporal Alignment Model: GTW finds spatiotemporal correlations based on CCA, and adds a soft penalty on the warping path by minimizing:

$$\min_{\substack{W_s, W_t \\ A_s, A_t}} \sum_{s,t=1}^{m} \|W_s^{\mathrm{T}} D_s^{\mathrm{T}} A_s - W_t^{\mathrm{T}} D_t^{\mathrm{T}} A_t\|_{\mathrm{F}}^2 + \sum_{s}^{m} \eta \|F_l Q a_s\|_2^2, \tag{19}$$

where W_s and W_t are spatial transformations, $F_l \in \mathbb{R}^{l \times l}$ is the first order differential operator and $Qa_s \in \mathbb{R}^l$ is the warping path.

Compared to GTW, our model aligns two sequences by the key frames one-by-one, which eliminates the redundant frames or overlapping of sequences by sparse learning:

$$\min_{A_s, A_t} \lambda_1 \|A_s D_s - A_t D_t\|_{\mathrm{F}}^2 \; S + \lambda_2 \|A_s\|_{2,1} + \lambda_3 \|A_t\|_{2,1}, \tag{20}$$

where A_s and A_t are sparse warping matrices to select key frames by $L_{2,1}$ norm.

In a word, our model aims to find a subspace, using temporal alignment of key frames from pairwise sequences. In addition, our model is designed for three challenges: sub-action, multi-subject and multi-modality, which are not fully solved by the state-of-the-art.

VI. Experiment

This section includes three experiments: (1) temporal alignment of synthetic data to show the effectiveness of sparse learning, (2) generic SCTA for three problems: sub-action, multi-subject and multi-modality, and (3) systematic evaluations on different scenarios, which have different influences on action recognition via either temporal alignment or sparse learning. Additionally, we also analyze the parameters setting and time complexity.

A. Datasets & Experiment Setting

In this subsection, there are three experimental settings: (1) synthetic temporal alignment (Section VI-B); (2) DNTF mechanism with two layers for subspace alignment comparison (Section VI-C) to solve three challenges; (3) action recognition with first layer NTF under different scenarios (Section VI-D) to evaluate temporal alignment and sparse learning.

We evaluate two popular datasets: MSRDailyActivity3D action dataset,[1] and MSRActionPairs action dataset.[2] In both datasets, we explore RGB and depth image modalities for three challenges discussed in this paper. The three datasets are introduced below.

1) Synthetic Dataset: We generate three sequences randomly for comparison.

2) MSRDailyActivity3D Dataset: In this dataset, there are 16 different actions performed by ten subjects, each of which acts twice. We use the cropped depth data in our experiment. First, each action is sub-sampled to $80 \times 80 \times 10$, and then we use a Gabor filter to extract features from the sequence.

3) MSRActionPairs Dataset: This dataset includes 12 action categories in six pairs. Each action has ten instances, each of which is performed in three trials. This gives a total of 360 samples and each category contains 30 samples. In this experiment, we explore the HOG feature instead of Gabor to improve the performance of SSM method. Each action sample size is $84 \times 53 \times 20$ after extracting HOG features.

B. Synthetic Temporal Alignment

In this subsection, we generate three sets of signals to evaluate the proposed SCTA and others to demonstrate the performance of key frames selection incorporated with temporal alignment. Notably, both GTW and SCTA can align spatiotemporal features, but the main difference between them lies in that SCTA employs sparse constraint to select key frames and rules out noisy frames of a pair of action sequences. Next, we detail the competitive methods used in this experiment:

1) Procrustes Dynamic Time Warping (pDTW): pDTW is an extension of DTW, which is proposed for shape alignment [10]. pDTW aligns two sequences by minimizing:

$$J_{pDTW}(A_{s/t}) = \sum_{s,t=1}^{m} \frac{1}{2} \|D_s A_s - D_t A_t\|_{\mathrm{F}}^2, \tag{21}$$

where $A_{s/t} \in \{0, 1\}$ is the warping matrix and $D_{s/t}$ is s/t-th sequence drawn from m samples.

2) Procrustes Derivative Dynamic Time Warping (pDDTW): pDDTW is based on DDTW [49], which uses derivatives of features. pDDTW aligns two sequences by minimizing:

$$J_{pDDTW}(A_{s/t}) = \sum_{s,t=1}^{m} \frac{1}{2} \|D_s F_s^{\mathrm{T}} A_s - D_t F_t^{\mathrm{T}} A_t\|_{\mathrm{F}}^2, \tag{22}$$

where $F_{s/t}$ is the first order differential operator.

[1] http://users.eecs.northwestern.edu/~jwa368/my_data.html
[2] http://www.cs.ucf.edu/oreifej/HON4D.html

3) Procrustes Iterative Motion Warping (pIMW): IMW iteratively handles time warping and spatial transformation of two sequences [50], and pIMW is extended to align multiple sequences by minimizing:

$$\begin{aligned} &J_{pIMX}(A_{s/t}, R_{s/t}, O_{s/t}) \\ &\quad = \sum_{s,t=1}^{m} \frac{1}{2} \|(D_s \circ R_s + O_s) A_s - (D_t \circ R_t + O_t) A_t\|_{\mathrm{F}}^2 \\ &\qquad + \sum_{s=1}^{m} \left(\eta_s^a \|R_s F_s^{a\mathrm{T}}\|_{\mathrm{F}}^2 + \eta_s^b \|O_s F_s^{b\mathrm{T}}\|_{\mathrm{F}}^2 \right), \end{aligned} \tag{23}$$

where $R_{s/t}, O_{s/t}$ are scaling and translating parameters. $F_{s/t}^a, F_{s/t}^b$ are first order differential operators.

4) Procrustes Canonical Time Warping (pCTW): pCTW minimizes the distance of two sequences in low dimensional space, and aligns the warping paths of them by:

$$\begin{aligned} &J_{pCTW}(W_s, W_t, A_s, A_t) \\ &\quad = \sum_{s,t=1}^{m} \|W_s^{\mathrm{T}} D_s A_s - W_t^{\mathrm{T}} D_t A_t\|_{\mathrm{F}}^2 + \phi(W_s) + \phi(W_t), \end{aligned} \tag{24}$$

where $\phi(W) = \frac{1}{1-\eta} \|W\|_{\mathrm{F}}^2$, and W_s, W_t satisfy the orthogonal constraints:

$$\begin{cases} W_s^{\mathrm{T}} \left((1-\eta) D_s A_s A_s^{\mathrm{T}} D_s^{\mathrm{T}} + \eta \mathbf{I} \right) W_s = \mathbf{I}, \\ W_t^{\mathrm{T}} \left((1-\eta) D_t A_t A_t^{\mathrm{T}} D_t^{\mathrm{T}} + \eta \mathbf{I} \right) W_t = \mathbf{I}, \end{cases} \tag{25}$$

where $\eta \in \{0, 1\}$ is a penalty between the error and regularization terms.

Fig. 4 shows the results of temporal alignment of triple sequences of synthetic data. We can see that pDTW fails because of distorted spatial sequences. The feature derivatives of pDDTW do not well capture the structure of sequences. pIMW overfits the sequences and the noise (third spatial component), whereas pCTW and GTW can successfully select features therefore removing the noisy dimension. SCTA performs feature (key frames) selection not only spatially but also temporally, and yields small alignment error.

C. Three Challenges in Temporal Alignment

In this subsection, we design three experiments to demonstrate the capability of our method to address the three challenges in temporal alignment. A subset of MSRActionPairs dataset is used for the evaluations where three body appearances, two modalities of ten subjects are selected. Here we briefly introduce the competitive methods in this section:

- Transfer Joint Matching (TJM) [51] minimizes the variance between source and target data in a new subspace by assigning less penalty on source data irrelevant to target data by a kernel mapping.
- GFK [52] learns many intermediate subspaces on a manifold to align source and target domains for transfer learning.
- SA [48] finds the subspaces of source and target domains and transforms source subspace by an affinity matrix to couple target subspace.

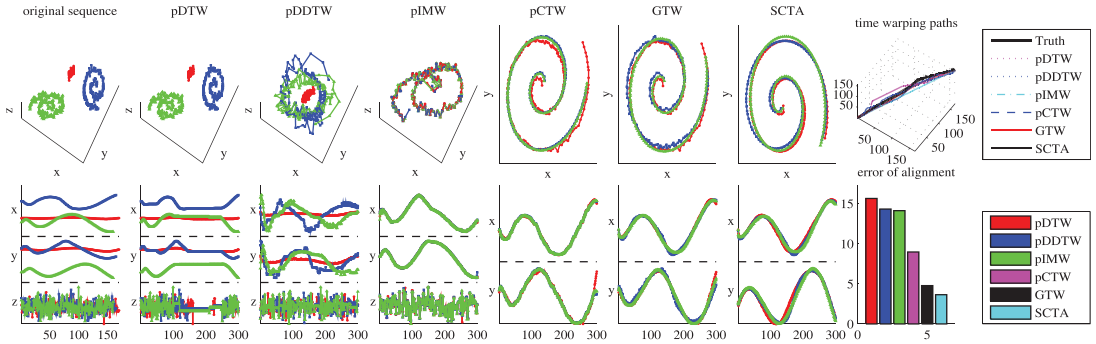


Fig. 4. Synthetic data evaluations. Original triple sequences $X_i, i \in \{1, 2, 3\}$ are generated first, with additional Gaussian noises in the third dimension. Spatiotemporal warping functions are calculated by pDTW, pDDTW, pIMW, pCTW, GTW and SCTA, respectively. pCTW, GTW and SCTA are based on CCA to align the homogeneous resources, and rule out the noises from the third dimension. Sub-figure on upper right shows different warping paths, while that on bottom right indicates mean alignment errors.

TABLE I

ACCURACY OF SUB-ACTION AND MULTI-SUBJECT PROBLEMS

Problem	Feature	JTM	GFK	LSSA	$\lambda_p = 0$	Ours-I	Ours-II
Sub-action	HOG	0.79	0.86	**0.90**	0.80	0.89	**0.90**
	Gabor	0.84	0.88	0.89	0.84	0.90	**0.93**
Multi-subject	HOG	0.77	0.79	0.84	0.84	**0.87**	**0.87**
	Gabor	0.75	0.79	0.81	0.65	0.73	**0.82**

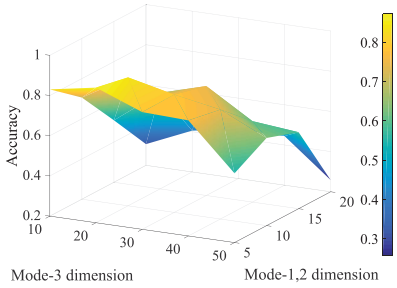


Fig. 5. Accuracy of DNTF on MSRActionPairs dataset.

- LSSA [53] performs kernelized SA based on selecting landmarks from source and target domains.

Table I shows the performances of sub-action and multi-subject problems, where $\lambda_p = 0 (p = 1, 2, 3)$ indicates the degenerated model of our method, which means neither sparse learning nor temporal alignment. "Ours-I" indicates our single layer model by NTF, and "Ours-II" means our two layers model DNTF with both NTF and TT. Since we focus on the performances of different models instead of individual features, we employ two common features extracted from action videos for comparison, i.e., HOG and Gabor. Fig. 5 shows the accuracies of DNTF under different dimensions in the multi-subject problem, and we can see that DNTF obtains higher accuracy in lower dimensional space on each mode.

We create four different Training-Testing settings for this problem: (1) RGB-Depth modalities of ten subjects. Particularly, RGB data are used for training and reference and we evaluate the labels of new depth data. (2) Depth-RGB modalities of ten subjects. (3) RGB-Depth modalities of three sub-actions. RGB data are used for training and reference and depth data for testing. (4) Depth-RGB modalities of three sub-actions. We employ some recent subspace alignment methods for comparison in the experiment. Table II shows the accuracy of subspace alignment methods for cross-modality experiments, i.e., different modalities for training and testing. We can see that LSSA performs better than SA, which verifies that the landmark based method is reasonable. Both our method with key frames selection Ours-I and deep structure Ours-II obtain better accuracy in most cases, which demonstrates the temporal alignment of key frames scheme is able to extract more discriminant features for action recognition. Fig. 6 shows the alignment results for multi-subject and multi-modality problems. We can see that the key poses of an action are captured and aligned properly.

TABLE II

ACCURACY OF MULTI-MODALITY PROBLEM, TRAIN-TEST: RGB-DEPTH & DEPTH-RGB

RGB-Depth	TJM	GFK	SA	LSSA	$\lambda_p = 0$	Ours-I	Ours-II
Multi-subject	0.08	0.11	0.05	0.17	0.26	**0.37**	**0.37**
Sub-action	0.08	0.05	0.06	0.19	0.24	0.25	**0.33**
Depth-RGB	JTM	GFK	SA	LSSA	$\lambda_p = 0$	Ours-I	Ours-II
Multi-subject	0.08	0.07	0.10	0.17	0.19	0.31	**0.37**
Sub-action	0.08	0.07	0.12	0.16	0.20	**0.36**	0.25

D. Action Recognition of Different Scenarios

1) Competitive Methods: In this subsection we introduce two competitive methods, and three scenarios with different parameters setting of our model for comparisons.

- Discriminant Non-Negative Tensor Factorization (DsNTF) [54] integrates the Fisher criterion into the NTF for discriminant feature learning.
- SSM [14] can measure two action sequences frame-by-frame, and is insensitive to multi-view problem and individual diversity.
- Scenario 1 (S_1): $\lambda_p = 0$ $(p = 1, 2, 3)$. Neither sparse learning nor temporal alignment in our model.
- Scenario 2 (S_2): $A_{s/t} = \mathbf{I}$. Temporal alignment is adopted but no sparse constraint in our model. Here we note it as $\Phi(\cdot)$ for simplicity.

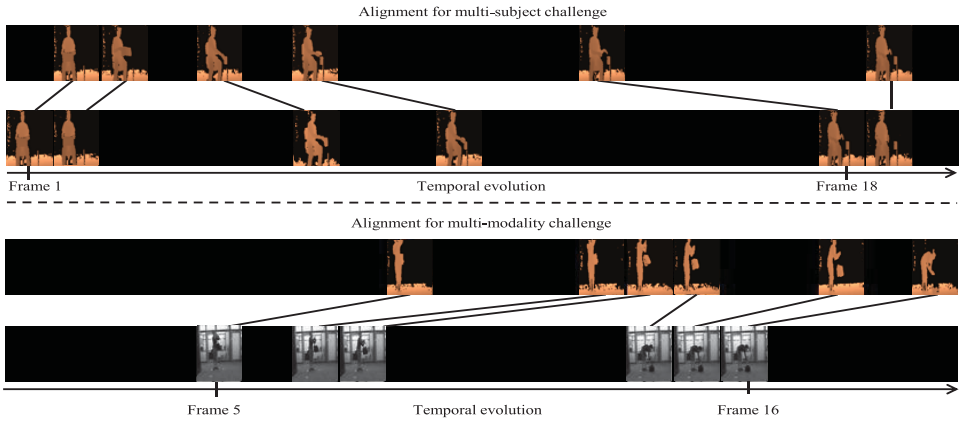


Fig. 6. Visualization of temporal alignment of key frames from two intra-class action sequences with 20 frames. The first two rows show the result for multi-subject challenge in "depth" modality, from which we can see that the key frames of "putting things on chair" actions are well aligned. The last two rows show the result for multi-modality challenge, from which we can see that the key frames of "putting things on floor" actions are aligned as well.

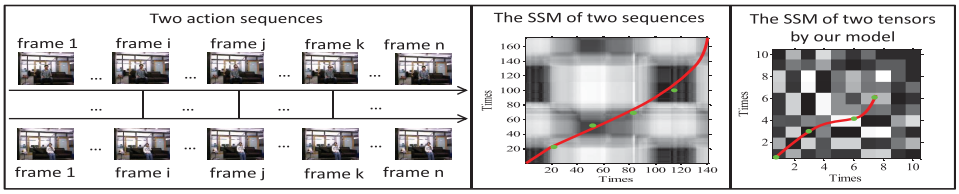


Fig. 7. Illustration of temporal key frames alignment on MSRDailyActivity3D dataset. Left: Sit action. Solid lines connect the key frames between two sequences. Middle: SSM of two action sequences by [14]. The red curve connects the realistic aligned frames along time while the green dots indicate the aligned key frames. Right: SSM by our method after sampling. The green dots indicate the key frames while the red curve shows the aligned path.

- Scenario 3 (S_3): $\lambda_p > 0$ and $A_s, A_t \neq \mathbf{I}$, which indicates both key frames selection and temporal alignment are performed in our model. To evaluate the key frames selection, we set $\lambda_p \in \{[0, 1], 1000\}$, to show the effects of different weights in the recognition task.

2) MSRDailyActivity3D Dataset: To better illustrate the correspondence between frames, we introduce the concept of SSM. SSM is an $f \times f$ matrix indicates the pairwise distances of all f frames, and each element is calculated by $\|(X)^i - (X)^j\|$, where $(X)^i$ and $(X)^j$ indicate the features of the i- and j-th frames, respectively. The entry (i, j) in SSM tends to be larger if the two frames are significantly different. A few SSMs drawn from MSRDailyActivity3D dataset shows the *selected key frames* from two sequences (Fig. 7). The left subfigure is the schematic diagram of our STCA method, which selects the key frames of two intra-class action sequences (drinking) for alignment. The middle subfigure illustrates the SSM of two sequences frame-by-frame. Note that the green dots are key frames aligned manually, and the red curve is the corresponding aligned path by [13]. The right subfigure shows the key frames selected automatically and aligned path on SSM by our method. In brief, most of the key frames locate in the dark areas with lower SSM values, which indicates the frame pairs from two sequences (x,y-axis) with large similarity.

As there are ten subjects in each category, we use *five, six, seven, eight, nine* subjects for training each time, and the rest for testing. In this experiment, we select the dimension settings $[10, 10, 10]$ and $[40, 40, 10]$ to see the performance under relative lower and higher dimensional spaces. Corresponding results are shown in Table III and Table IV. Here we introduce the parameter $\lambda_p (p = 1, 2, 3)$ setting.

TABLE III
ACCURACY (%) OF MSRDAILYACTIVITY3D WITH SETTING [10, 10, 10]

	Dimension=[10,10,10]						
#Subs	DsNTF	SSM	$\lambda_1 = 0$	$\Phi(\cdot)$	$\lambda_1 = 0.1$	$\lambda_1 = 1$	$\lambda_1 = 10^3$
5	27.50	25.63	36.88	33.13	29.38	31.88	**51.25**
6	44.38	28.13	36.88	**54.38**	44.38	45.00	51.88
7	57.50	25.00	50.00	56.25	47.50	60.00	**61.25**
8	58.13	23.44	65.63	69.38	64.38	**72.50**	60.63
9	80.00	34.38	80.63	78.75	82.50	**83.75**	80.63

- When $\lambda_1 = 0$, we set $\lambda_2 = \lambda_3 = 0$, which is noted as $\lambda_p = 0$ for simplicity.
- When $\lambda_1 \in \{(0, 1], 1000\}$, we set $\lambda_2 = \lambda_3 = 1$.

From Table III we can see that the better performance is obtained for each method with the increasing training number. It can be concluded that the best parameter is obtained at $\lambda_p > 0$. In addition, we can see that $\Phi(\cdot)$ performs better than $\lambda_p = 0$ in most cases, which means temporal alignment has a positive effect on accuracy. The corresponding result of parameter tuning is shown in Fig. 8(a). In general, $\lambda_p > 0$ performs better than $\lambda_p = 0$. The mean accuracy curve calculated under all the dimensions is increasing and reaches the peak at $\lambda_1 = 1$, which means temporal alignment has a higher weight in DNTF. In Table IV, DsNTF is competitive with ours, while SSM performs worse in most cases. We believe the reason is

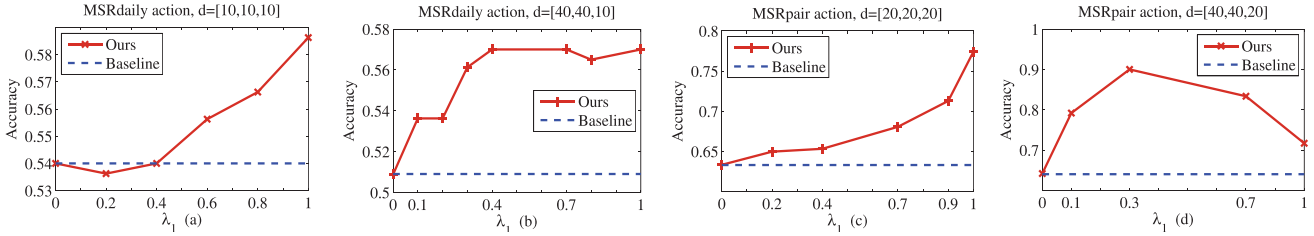


Fig. 8. Mean accuracy of the proposed method with $\lambda_1 \in [0, 1]$, $\lambda_2 = \lambda_3 = 1$ under different dimension settings on two datasets. Note $\lambda_1 = 0$ is baseline, and we can see the accuracy under $\lambda_1 > 0$ is higher than that of $\lambda_1 = 0$.

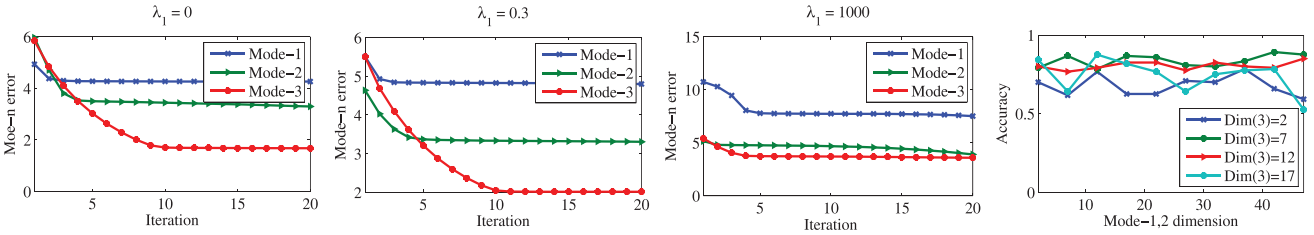


Fig. 9. First three sub-figures: *mode-n* error in different iterations on MSRActionPairs dataset. Fourth sub-figure: accuracy under *mode-n* ($n = 1, 2$) dimensions. We can see that *mode-3* dimension Dim(3) $= 7$ achieve relatively good results.

TABLE IV
ACCURACY (%) OF MSRDAILYACTIVITY3D WITH SETTING [40, 40, 10]

	Dimension=[40,40,10]						
#Subs	DsNTF	SSM	$\lambda_1 = 0$	$\Phi(\cdot)$	$\lambda_1 = 0.1$	$\lambda_1 = 1$	$\lambda_1 = 10^3$
5	33.75	31.25	**40.63**	28.75	22.50	35.63	**40.63**
6	**53.13**	32.03	35.00	43.75	51.25	33.75	40.00
7	39.38	35.42	33.13	58.13	59.38	**63.13**	55.63
8	66.88	34.38	68.13	65.00	59.38	**71.25**	58.75
9	78.75	46.88	77.50	80.63	75.63	**81.25**	80.63

TABLE V
ACCURACY (%) OF MSRACTIONPAIRS WITH SETTING [20, 20, 20]

	Dimension=[20,20,20]						
#Train	DsNTF	SSM	$\lambda_1 = 0$	$\Phi(\cdot)$	$\lambda_1 = 0.1$	$\lambda_1 = 1$	$\lambda_1 = 10^3$
16	67.26	72.62	64.88	46.43	67.26	**78.57**	64.88
17	75.64	76.28	60.26	**83.33**	76.28	80.77	65.38
18	73.61	73.61	63.89	74.31	72.92	**79.86**	78.47
19	70.45	75.00	49.24	68.94	58.33	71.21	**77.27**
20	73.33	74.17	**78.33**	57.50	67.50	76.67	60.83

that the size of dimensions or frames length is insufficient for SSM to find the similarity of two sequences. From both tables we can see that our performance is better than others given the increasing number of subjects for training.

3) MSRActionPairs Dataset: As there are 30 samples in each category, we use $16 \sim 20$ samples for training, and the rest for testing. We use the dimension settings $[20, 20, 20]$ and $[40, 40, 40]$ for evaluations. The corresponding results are shown in Table V and Table VI. We have the same λ_p setting with the last experiment. In Table V, we can see $\lambda_1 = 1$ is comparative to other methods, slightly worse than $\lambda_p = 0$ ($\approx 2\%$) under 20 training samples. However, the mean accuracy of the former is consistently higher than the latter in Fig.8(c). In addition, we can find that $\Phi(\cdot)$ is better than $\lambda_p = 0$ in most cases, meaning temporal alignment plays a positive role for accuracy. On the other hand, the SSM method is improved compared to the results in the last experiment. We believe the proper features and dimension are critical for SSM.

In Table VI, we can see that SSM performs worse along with increasing number of training data. The main reason is that it does not have a training process, and therefore its accuracy is not necessarily related to the number of training samples. Fig. 8(d) shows the accuracy under $\lambda_1 \in [0, 1]$ with 20 training samples, which indicates that the best performance is obtained at $\lambda_1 = 0.3$. From Table V and VI we can see that our accuracy is higher than others given the increasing number of training samples at most cases.

TABLE VI
ACCURACY (%) OF MSRACTIONPAIRS WITH SETTING [40, 40, 20]

	Dimension=[40,40,20]						
#Train	DsNTF	SSM	$\lambda_1 = 0$	$\Phi(\cdot)$	$\lambda_1 = 0.1$	$\lambda_1 = 1$	$\lambda_1 = 10^3$
16	55.95	**83.33**	70.83	75.00	51.79	58.93	56.55
17	64.74	**86.54**	64.10	68.59	51.28	30.77	**86.54**
18	45.14	**85.42**	25.00	76.39	77.08	57.64	81.94
19	84.09	81.82	42.42	**87.88**	84.09	68.94	74.27
20	57.50	80.83	64.17	67.50	**90.00**	83.33	75.83

Fig. 9(a) $\sim$ 9(c) show *mode-n* error along different iterations, when $\lambda_1 = 0$, 0.3 and 1000 respectively. We can see that the error is stable within a few iterations, which indicates that our method converges well on realistic data. Fig. 9(d) shows the accuracy under different dimensions of *mode-1,2*, from which we can see that the better result is obtained by *mode-3* with dimension Dim(3) $= 7$. In summary, the results above indicate that the performance is optimized by proper *mode-n* dimensions. Either insufficient or redundant information will affect the performance.

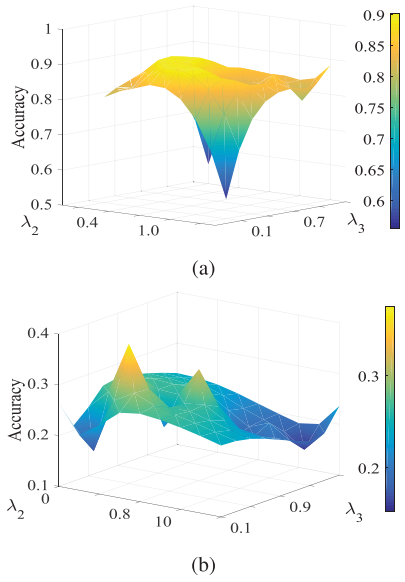


Fig. 10. Accuracy with different penalty factors λ_2 and λ_3 for top: sub-action and bottom: multi-modality challenges. We can see that the accuracy under $\lambda_2, \lambda_3 > 0$ is better than that of $\lambda_2 = 0$. a Sub-action. b Multi-modality.

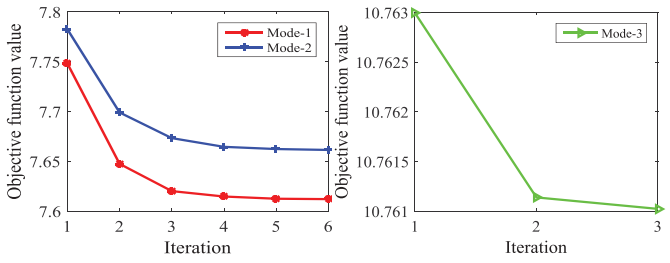


Fig. 11. Objective function value (OFV) of our model on MSRActivityPairs dataset. Left: *mode-1,2* OFV. Right: *mode-3* OFV. The OFVs of all modes will not change after a few iterations.

E. Parameters Analysis & Time Complexity

In this subsection, we systematically analyze four factors of our DNTF model with two layers on MSRActionPairs dataset, including (1) penalty parameters $\lambda_p (p = 1, 2, 3)$, (2) Objective Function Value (OFV), (3) Signal-to-Noise Ratio (SNR), and (4) time complexity comparison.

1) Penalty Parameters λ_p: We consider two problems to illustrate the role of λ_p, i.e., (1) sub-action problem, (2) multi-modality problem, by 10-fold multi-subject tests with *RGB-depth* as the Train-Test setting. Here we evaluate the settings: $\lambda_1 = 1$, $\lambda_2 \in \{0, 0.2, 0.4, 0.6, 0.8, 1.0, 10, 100, 1000\}$ and $\lambda_3 = \lambda_2$. Fig. 10(a) illustrates the results of the first problem, and we can see that higher accuracy is obtained when λ_2, $\lambda_3 > 0$, which outperforms the performance when $\lambda_2 = \lambda_3 = 0$ (no key frame selection). Fig. 10(b) illustrates the similar trends. The result indicates that key frames selection aids in improving the performance of recognition.

2) OFV: For the subspace dimension setting [10, 10, 5], we calculate the OFV of each mode as shown in Fig. 11. We can see that OFVs of all modes become stable within ten iterations, which indicates that DNTF model converges well.

3) SNR: Root Relative Squared Error (RRSE) is used to reveal how DNTF is affected by *mode-n* dimensions.

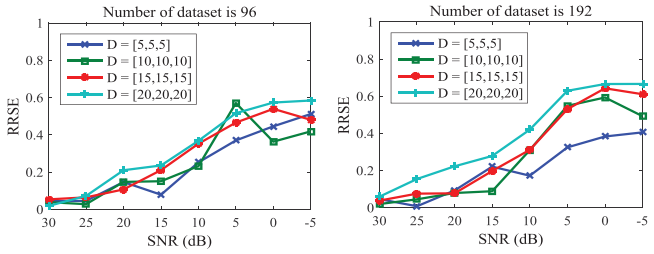


Fig. 12. RRSE under various noise levels (SNR). Top: dataset size is 96. Bottom: dataset size is 192.

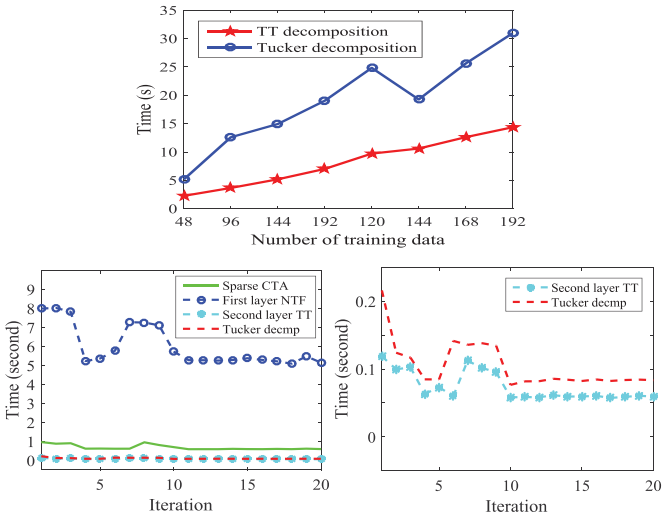


Fig. 13. Running time comparisons. Top: TT and Tucker decomposition given different numbers of training data. Bottom left: sparse CTA, first layer NTF, second layer TT and Tucker decomposition in 10-fold cross-validation. Bottom right: TT and Tucker decomposition.

Specifically, given a tensor $\mathcal{X}$, we have $\mathcal{X} = \mathcal{W}\mathcal{H}$. Then, we add different levels of Gaussian noises (30dB ~ -5dB), so the decomposition with contamination is $\widetilde{\mathcal{X}} = \mathcal{W}\widetilde{\mathcal{H}}$, where $\widetilde{\mathcal{H}}$ is the perturbed tensor with noises. We define:

$$RRSE = \frac{\|\mathcal{H} - \widetilde{\mathcal{H}}\|_{\mathrm{F}}}{\|\mathcal{H}\|_{\mathrm{F}}}. \tag{26}$$

Fig. 12 shows the RRSE under different dimensions, with 96 and 192 samples, respectively. We can see that increasing dimensions yield higher RRSE in most cases while smaller dimensions lead to lower RRSE. This consists with the phenomenon of higher accuracy under small dimensions shown in Fig. 5.

4) Time Complexity Comparison: We compare the running time of two tensor decomposition methods in our model: the Tucker decomposition and TT in the second layer. The running time under different data scales is shown in top of Fig. 13. The results confirm that using TT for DNTF reduces the running time compared with the Tucker decomposition.

Besides, we compare the running time of sparse CTA, first layer NTF, second layer TT and Tucker decomposition in bottom of Fig. 13. We can see that NTF uses more time than the Tucker decomposition and TT, and the Tucker uses more than TT, which is theoretically analyzed already. NTF needs to calculate several variables, therefore it costs more time than

TT which only decomposes $\mathcal{W}$ and $\mathcal{H}$. Since there are only 20 frames in one sequence in this dataset, key frames selection by sparse operation is fast.

VII. Conclusions

In this paper, we proposed a discriminant deep tensor decomposition method applicable to sub-action, multi-subject, and multi-modality problems in action recognition. We temporally aligned the key frames of intra-class action sequences using a sparse learning technique, then we designed a DNTF mechanism to find a subspace for key-frame action recognition. Additionally, we set different scenarios to evaluate the performances of key frame selection and temporal alignment on action recognition. In the experiment section, we conducted extensive experiments on both synthetic and realistic datasets to demonstrate the effectiveness of our method. We also analyzed key parameters for a better understanding of the proposed model.

References

[1] B. Singh, T. K. Marks, M. Jones, O. Tuzel, and M. Shao, "A multi-stream bi-directional recurrent neural network for fine-grained action detection," in *Proc. CVPR*, 2016, pp. 1961–1970.

[2] H. Wang, D. Oneata, J. Verbeek, and C. Schmid. (2015). "A robust and efficient video representation for action recognition." [Online]. Available: https://arxiv.org/abs/1504.05524

[3] H. Pazhoumand-Dar, C.-P. Lam, and M. Masek, "Joint movement similarities for robust 3D action recognition using skeletal data," *JVCIR*, vol. 30, pp. 10–21, Jul. 2015.

[4] N. P. Cuntoor and R. Chellappa, "Key frame-based activity representation using antieigenvalues," in *ACCV*. Berlin, Germany: Springer, 2006, pp. 499–508.

[5] L. Shao and L. Ji, "Motion histogram analysis based key frame extraction for human action/activity representation," in *Proc. IEEE CRV*, May 2009, pp. 88–92.

[6] Z. Qiu-yu, L. Lu, Z. Mo-yi, D. Hong-xiang, and L. Jun-chi, "A dynamic gesture trajectory recognition based on key frame extraction and HMM," *Int. J. Signal Process. Image Process. Pattern Recognit. (IPPR)*, vol. 8, no. 6, pp. 91–106, 2015.

[7] A. Hanjalic and H. Zhang, "An integrated scheme for automated video abstraction based on unsupervised cluster-validity analysis," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 8, pp. 1280–1289, Dec. 1999.

[8] Y. Rui, T. S. Huang, and S. Mehrotra, "Exploring video structure beyond the shots," in *Proc. IEEE MCS*, Jul. 1998, pp. 237–240.

[9] F. Zhou and F. de la Torre, "Generalized canonical time warping," *IEEE Trans. Pattern Anal. Mach. Intell. TPAMI*, vol. 38, no. 2, pp. 279–294, Feb. 2016.

[10] F. Zhou and F. De la Torre, "Generalized time warping for multi-modal alignment of human motion," in *Proc. IEEE CVPR*. Jun. 2012, pp. 1282–1289.

[11] F. Zhou and F. Torre, "Canonical time warping for alignment of human behavior," in *Proc. NIPS*, 2009, pp. 2286–2294.

[12] C. Rao, A. Gritai, M. Shah, and T. Syeda-Mahmood, "View-invariant alignment and matching of video sequences," in *Proc. IEEE ICCV*, Oct. 2003, pp. 939–945.

[13] A. Gritai, Y. Sheikh, C. Rao, and M. Shah, "Matching trajectories of anatomical landmarks under viewpoint, anthropometric and temporal transforms," *Int. J. Comput. Vis. IJCV*, vol. 84, no. 3, pp. 325–343, 2009.

[14] I. N. Junejo, E. Dexter, P. Laptev, and I. Perez, "View-independent action recognition from temporal self-similarities," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 1, pp. 172–185, Jan. 2011.

[15] D. Gong and G. Medioni, "Dynamic manifold warping for view invariant action recognition," in *Proc. IEEE ICCV*, Nov. 2011, pp. 571–578.

[16] M. A. Nicolaou, V. Pavlovic, and M. Pantic, "Dynamic probabilistic CCA for analysis of affective behaviour," in *Computer Vision—ECCV*. Berlin, Germany: Springer, 2012, pp. 98–111.

[17] J. Li, T. Zhang, W. Luo, J. Yang, X. Yuan, and J. Zhang, "Sparseness analysis in the pertraining of deep neural networks," *IEEE Trans. Neural Netw. Learn. Syst. (TNNLS)*, no. 99, p. 114, Mar. 2016, doi: 10.1109/TNNLS.2541681.

[18] J. Li, H. Chang, and J. Yang, "Sparse deep stacking network for image classification," in *Proc. 29th Conf. Artif. Intell. AAAI*, 2015, pp. 3804–3810.

[19] L. Jun, W. Luo, J. Yang, and X. Yuan. (2013). "Unsupervised Pretraining Encourages ModerateSparseness." [Online]. Available: http//arXiv preprint arXiv:1312.5813

[20] S. Shariat and V. Pavlovic, "Isotonic CCA for sequence alignment and activity recognition," in *Proc. IEEE ICCV*, Nov. 2011, pp. 2572–2578.

[21] C. Jia, M. Shao, and Y. Fu, "Sparse canonical temporal alignment with NTF for RGB-D action recognition," in *Proc. IJCNN*, Jul. 2016, pp. 2260–2266.

[22] C. Jia, Y. Kong, Z. Ding, and Y. R. Fu, "Latent tensor transfer learning for RGB-D action recognition," in *Proc. ACMMM*, 2014, pp. 87–96.

[23] C. Sun, I. N. Junejo, M. Tappen, and H. Foroosh, "Exploring sparseness and self-similarity for action recognition," *IEEE Trans. Image Process.*, vol. 24, no. 8, pp. 2488–2501, Aug. 2015.

[24] D. Tao, X. Li, X. Wu, and S. J. Maybank, "General tensor discriminant analysis and Gabor features for gait recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 10, pp. 1700–1715, Oct. 2007.

[25] J. Assa, Y. Caspi, and D. Cohen-Or, "Action synopsis: Pose selection and illustration," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 667–676, Jul. 2005.

[26] Z. Zhao and A. M. Elgammal, "Information theoretic key frame selection for action recognition," in *Proc. BMVC*, Sep. 2008, pp. 1–10.

[27] S. Vijayanarasimhan and K. Grauman, "Active frame selection for label propagation in videos," in *Computer Vision—ECCV*. Berlin, Germany: Springer, 2012, pp. 496–509.

[28] L. Liu, L. Shao, and P. Rockett, "Boosted key-frame selection and correlated pyramidal motion-feature representation for human action recognition," *Pattern Recognit.*, vol. 46, no. 7, pp. 1810–1818, 2013.

[29] C. Wang and S. Mahadevan, "Manifold alignment without correspondence," in *IJCAI*, vol. 2. 2009, pp. 1273–1278.

[30] Y. M. Lui, "Tangent bundles on special manifolds for action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 6, pp. 930–942, Jun. 2012.

[31] M. Heiler and C. Schnörr, "Controlling sparseness in non-negative tensor factorization," in *Computer Vision—ECCV*. Springer, 2006, pp. 56–67.

[32] A. Shashua and T. Hazan, "Non-negative tensor factorization with applications to statistics and computer vision," in *Proc. ICML*, 2005, pp. 792–799.

[33] F. Wu, X. Tan, Y. Yang, D. Tao, S. Tang, and Y. Zhuang, "Supervised nonnegative tensor factorization with maximum-margin constraint," in *Proc. AAAI*, 2013, pp. 962–968.

[34] Q. V. Le, W. Y. Zou, S. Y. Yeung, and A. Y. Ng, "Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis," in *Proc. IEEE CVPR*, 2011, pp. 3361–3368.

[35] Y. Sun, X. Wang, and X. Tang, "Deep learning face representation from predicting 10,000 classes," in *Proc. IEEE CVPR*, 2014, pp. 1891–1898.

[36] G. Trigeorgis, K. Bousmalis, S. Zafeiriou, and B. Schuller, "A deep semi-nmf model for learning hidden representations," in *Proc. ICML*, 2014, pp. 1692–1700.

[37] G. Trigeorgis, K. Bousmalis, S. Zafeiriou, and B. W. Schuller. (2015). "A deep matrix factorization method for learning attribute representations." [Online]. Available: https://arxiv.org/abs/1509.03248

[38] M. Sajjad, I. Mehmood, and S. W. Baik, "Sparse representations-based super-resolution of key-frames extracted from frames-sequences generated by a visual sensor network," *Sensors*, vol. 14, no. 2, pp. 3652–3674, 2014.

[39] I. Kotsia and I. Patras, "Support tucker machines," in *Proc. IEEE CVPR*, Jun. 2011, pp. 633–640.

[40] X. Shi, H. Yuan, W. Hu, C. Yuan, and J. Xing, "Multi-target tracking with motion context in tensor power iteration," in *Proc. CVPR*, Jun. 2013, pp. 3518–3525.

[41] G. Zhong and M. Cheriet, "Large margin low rank tensor analysis," *Neural Comput.*, vol. 26, no. 4, pp. 761–780, 2014.

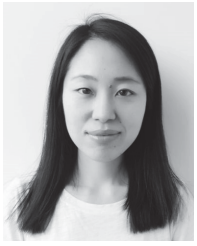
[42] Q. Wu, T. Xia, C. Chen, H. Y. S. Lin, H. Wang, and Y. Yu, "Hierarchical tensor approximation of multi-dimensional visual data," *IEEE Trans. Vis. Comput. Graph.*, vol. 14, no. 1, pp. 186–199, Jan. 2008.

[43] A. Efros, A. C. Berg, G. Mori, and J. Malik, "Recognizing action at a distance," in *Proc. IEEE ICCV*, Oct. 2003, pp. 726–733.

[44] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "A survey of multilinear subspace learning for tensor data," *Pattern Recognit.*, vol. 44, no. 7, pp. 1540–1551, 2011.

[45] A. Novikov, D. Podoprikhin, A. Osokin, and D. P. Vetrov, "Tensorizing neural networks," in *Proc. NIPS*, Sep. 2015, pp. 442–450.

[46] I. V. Oseledets, "Tensor-train decomposition," *SIAM J. Sci. Comput.*, vol. 33, no. 5, pp. 2295–2317, 2011.
[47] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 171–184, Jan. 2013.
[48] B. Fernando, A. Habrard, M. Sebban, and T. Tuytelaars, "Unsupervised visual domain adaptation using subspace alignment," in *Proc. IEEE ICCV*, Dec. 2013, pp. 2960–2967.
[49] E. J. Keogh and M. J. Pazzani, "Derivative dynamic time warping," in *Proc. SDM*, vol. 1. 2001, pp. 5–7.
[50] E. Hsu, K. Pulli, and J. Popović, "Style translation for human motion," *ACM TOG*, vol. 24, no. 3, pp. 1082–1089, Jul. 2005.
[51] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu, "Transfer joint matching for unsupervised domain adaptation," in *Proc. IEEE CVPR*, Jun. 2014, pp. 1410–1417.
[52] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in *Proc. IEEE CVPR*, Jun. 2012, pp. 2066–2073.
[53] R. Aljundi, R. Emonet, D. Muselet, and M. Sebban, "Landmarks-based kernelized subspace alignment for unsupervised domain adaptation," in *Proc. CVPR*, Jun. 2015, pp. 56–63.
[54] S. Zafeiriou, "Discriminant nonnegative tensor factorization algorithms," *IEEE Trans. Neural Netw.*, vol. 20, no. 2, pp. 217–235, Feb. 2009.



Chengcheng Jia received the B.E. degree from Northeastern Normal University, the M.E. and first Ph.D. degrees in computer science from Jilin University, China, in 2007, 2010, and 2013, respectively, and the second Ph.D. degree from the Electrical and Computer Engineering Department, Northeastern University, Boston, MA, in 2016. She has been an Intern with the Pacific Northwest National Laboratory, Richland, WA, since 2016. Her research interests include tensor factorization, machine learning, and deep learning for visual representation.

Ming Shao (M'16) received the B.E. degree in computer science, the B.S. degree in applied mathematics, and the M.E. degree in computer science from Beihang University, Beijing, China, in 2006, 2007, and 2010, respectively, and the Ph.D. degree in computer engineering from Northeastern University, Boston, MA, in 2016. He has been a tenure-track Assistant Professor with the College of Engineering, University of Massachusetts Dartmouth, since 2016. His current research interests include sparse modeling, low-rank matrix analysis, deep learning, and applied machine learning on social media analytics. He was a recipient of the Presidential Fellowship of State University of New York at Buffalo from 2010 to 2012 and the best paper award winner of IEEE ICDM 2011 Workshop on Large Scale Visual Analytics. He has served as the Reviewer for IEEE journals, such as IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, and the IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING.

Yun Fu (S'07–M'08–SM'11) received the M.S. degree in statistics and the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana–Champaign. He has been an interdisciplinary Faculty Member with the College of Engineering and the College of Computer and Information Science, Northeastern University, since 2012. His research interests are machine learning, computer vision, social media analytics, and big data mining. He has extensive publications. He is a fellow of IAPR, a Lifetime Senior Member of the ACM and the SPIE, a Lifetime Member of AAAI, OSA, and Institute of Mathematical Statistics, a member of Global Young Academy and INNS, and a Beckman Graduate Fellow from 2007 to 2008. He received seven Young Investigator Awards, such as the 2016 National Academy of Engineering Grainger Foundation Frontiers of Engineering Award, the 2016 IEEE CIS Outstanding Early Career Award, the 2016 UIUC ECE Young Alumni Achievement Award, the 2015 National Academy of Engineering U.S. Frontiers of Engineering, the 2014 ONR Young Investigator Award, the 2014 ARO Young Investigator Award, and the 2014 INNS Young Investigator Award, seven Best Paper Awards, such as the SPIE DSS 2016, the SIAM SDM 2014, the IEEE ICME 2014 candidate, the IEEE FG 2013, the IEEE ICDM-LSVA 2011, the IAPR ICFHR 2010, and the IEEE ICIP 2007, three Industrial Research Awards, such as the 2016 Samsung GRO Award, the 2015 Adobe Faculty Research Award, and the 2010 Google Faculty Research Award, and two Service Awards, the 2012 IEEE TCSVT Best Associate Editor and the 2011 IEEE ICME Best Reviewer. He serves as an Associate Editor, the Chair, a PC member, and a Reviewer of many top journals and international conferences/workshops.