Pattern Recognition 85 (2019) 50–59

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

Pseudo low rank video representation

Tingzhao Yu[a,b,*], Lingfeng Wang[a], Chaoxu Guo[a,b], Huxiang Gu[a], Shiming Xiang[a], Chunhong Pan[a]

[a] National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, China
[b] School of Computer and Control Engineering, University of Chinese Academy of Sciences, China

ARTICLE INFO

Article history:
Received 26 February 2018
Revised 10 July 2018
Accepted 31 July 2018
Available online 3 August 2018

Keywords:
Pseudo low rank
Data driven
Low resolution
Action recognition

ABSTRACT

Action recognition plays a fundamental role in computer vision and has drawn growing attention recently. This paper addresses this issue conditioned on extreme Low Resolution (abbreviated as eLR). Generally, eLR video is often susceptible to noise, thus extracting a robust representation is of great challenge. Besides, due to the limitation of video resolution, eLR video cannot be cropped or resized randomly, then it is inevitably complicated to design and to train a deep network for eLR video. This paper proposes a novel network for robust video representation by employing pseudo tensor low rank regularization. A new Video Low Rank Representation model (named VLRR) is first proposed to recover the inherent robust component of a given video, and then the recovered term is introduced to a convolutional Network (denoted pLRN) as an auxiliary pseudo Low Rank guidance. Benefitting from the auxiliary guidance, pLRN can learn an approximate low rank term end-to-end. Besides, this paper presents a new initialization strategy for eLR recognition neTwork based on Tensor factorization (dubbed TenneT). TenneT is data-driven and learns the convolutional kernels totally from the video distribution while without any back-propagation. It outperforms random initialization both in speed and accuracy. Experiments on benchmark datasets demonstrate the effectiveness and superiority of the proposed method.

1. Introduction

Video based human action recognition is currently a hot research topic [1,2] with wide applications, *e.g.*, video surveillance [3,4], event detection [5,6] and crowd analysis [7,8]. The task of video representation is to learn a discriminative feature transformation for robust video analysis. Early researches focus on detecting the trajectories of spatio-temporal interest points [9,10], while recently various deep models have been exploited for getting discriminative spatial-temporal descriptors in an end-to-end manner [11,12]. The main difference between a sequence of video frames and a series of images lies in the temporal correlation. Efforts have been contributed via 3D spatial temporal convolution [11], and further improvements are achieved through two-stream networks [12–14]. Nevertheless, extracting effective video representation is still of great challenge especially for extreme Low Resolution (eLR) videos due to the background motion, foreground occlusion, illumination changes, viewpoint variation and long-temporal duration.

As a promising research topic, eLR video analysis [15–17] has great potential primarily for three considerations. (***a***) It can protect privacy. An emerging challenge for computer vision is how to analyze human's activities without invading privacy. eLR analysis provides a potential solution. (***b***) It can be easily implemented to mobile devices. The advent of wearable devices raised the opportunity for eLR analysis. The transmission and storage are simple, fast and effective. (***c***) eLR video is widely existing in real-world video surveillance. Even when the video is of high resolution, the region of interest often occupies only a small proportion, which in turns, transforms the problem into eLR analysis. These make an urgent demand for eLR video analysis. Prevalent video analysis methods cannot be directly modified to this issue. And it is difficult to construct a rather deep network as applied for High Resolution (HR) videos due to the low spatial resolution. To better deal with eLR videos, the following three factors must be under thorough consideration. (***a***) The extreme low resolution makes the video vulnerable to noises. And (***b***) data augmentation is rather difficult for that eLR video cannot be cropped or resized arbitrarily, which in turn makes (***c***) *training* the model with extra challenge.

This paper proposes a novel pseudo tensor low rank regularized network for eLR video action recognition. The motivation and semantic architecture are illustrated in Fig. 1. Different form the existing works [18–20], in which they exploit the low rank property of convolutional kernels, this paper imposes pseudo low rank

[*] Corresponding author.
E-mail address: tingzhao.yu@nlpr.ia.ac.cn (T. Yu).

https://doi.org/10.1016/j.patcog.2018.07.033
0031-3203/© 2018 Elsevier Ltd. All rights reserved.

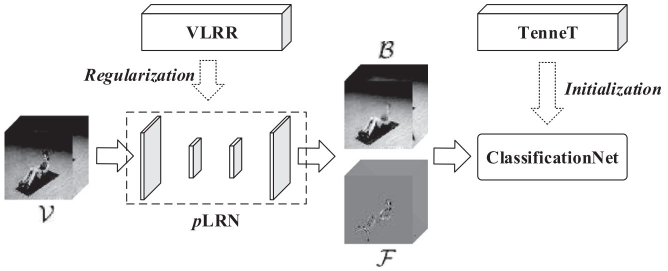


Fig. 1. ***Motivation and architecture***. Given a noised eLR video $\mathcal{V}$, pLRN learns a low rank background $\mathcal{B}$ or a sparse foreground $\mathcal{F}$ with a pseudo low rank regularization. Both $\mathcal{B}$ and $\mathcal{F}$ can be utilized for eLR action recognition. The training data for pLRN can be obtained via VLRR. TenneT is employed to initialize the classification network for faster convergence.

regularization into feature maps. Furthermore, this paper presents a new initialization policy for eLR recognition network. The contributions are summarized as follows:

- This paper proposes a new video low rank representation (VLRR) model for robust motion representation. VLRR decomposes the noised eLR video $\mathcal{V}$ into three parts - a low rank background $\mathcal{B}$, a sparse foreground $\mathcal{F}$ and a noised term $\mathcal{E}$. $\mathcal{B}$ and $\mathcal{F}$ are employed for eLR recognition, and their performances are superior to the original noised eLR video $\mathcal{V}$.
- In order to learn $\mathcal{B}$ or $\mathcal{F}$ end-to-end, a novel convolution network (pLRN) with pseudo tensor nuclear norm (TNN) regularization is proposed. Instead of directly conducting TNN minimization, which is hard to be back-propagated, this paper employs the pre-obtained $\mathcal{B}$ or $\mathcal{F}$ (using VLRR) as an auxiliary guidance. Thus TNN minimization can be transformed into the differentiable l_p-norm minimization. This yields an efficient end-to-end network.
- This paper presents a new network initialization strategy (TenneT) for eLR action recognition. TenneT learns the convolution kernels totally from the training videos without any backpropagation. Thus it is data-driven. Without any data augmentation, TenneT initialization promotes the recognition network converges faster than random initialization.

2. Related work & overview

The motivation behind this paper is to achieve deep robust video representation, and the related work is three-fold.

Video representation. A discriminative video representation [21,22] is essential for video related analysis. Among the past few decades, hand-crafted features [9,10] with certain encoding techniques [23] are the dominant approaches. These methods usually depict the trajectories of spatial-temporal interest points. The recently proposed methods mainly concentrate on convolution networks, and a simple method is to conduct deep networks directly at frame level [24]. Further improvement has been devoted to integrating multiple adjacent frames [25]. Considering the temporal coherence within video clips, researches have also proposed 3D spatial-temporal convolution [11,26]. 3D convolution contains more plentiful temporal correlation, which is vital for video sequence analysis. Current state-of-the-art algorithms adopt a two-stream network [12–14]. Within this framework, a spatial stream operating on frame level is designed to recognize the video agent, and a temporal stream operating on optical flow level is expected to distinguish the video motion. ***Motivation***. Though effective for video analysis, two-stream network has three disadvantages. Firstly, two-stream network needs pre-extracted optical flows, which is time-consuming. Secondly, designing a two-stream network for eLR video may confront with extra challenge, because constructing a rather deep network is impractical for eLR video. Finally, when it comes to eLR video, two-stream network might be susceptible to noise. These issues inspire us to design an efficient and robust video representation for eLR video.

Tensor factorization. Tensor representation and factorization have great potential in compute vision [27–30]. Based upon the assumption of tensor low rank, researches have made great advances. A basic formulation [31] of tensor low rank for robust tenor representation is $\min \|\mathcal{L}\|_{\circledast} + \|\mathcal{S}\|_1, s.t.\ \mathcal{X} = \mathcal{L} + \mathcal{S}$, where $\mathcal{X}$ is the noised data, $\mathcal{L}$ is the recovered tensor low rank data and $\mathcal{S}$ is the sparse noise. However, one drawback of this model is its limitation in handling data with outliers. Thus an improved formulation [32] is $\min \|\mathcal{L}\|_{\circledast} + \|\mathcal{E}\|_{2,1}, s.t.\ \mathcal{X} = \mathcal{L} + \mathcal{E}$, where $\mathcal{E}$ is the outlier. There are also researches devoting to tensor factorization, *e.g.*, Canonical Polyadic (CP) [33]decomposition , Tucker decomposition [34] and tensor Singular Value Decomposition (tSVD) [35]. ***Motivation***. Tensor factorization is inherently suitable for high-dimension data analysis, *e.g.*, videos. In general, for a given video, the background is often low rank and the foreground is usually sparse [30]. Besides, videos themselves contain abundant information. These factors motivate us to consider a tensor low rank involved network for robust representation, and learn some constructive information totally from the data distribution.

Low resolution analysis. Low resolution analysis is a promising research topic with broad applications. For such a task, the most straightforward method is to restore the corresponding HR data [36–38]. Nevertheless, super resolution is itself an ill-posed problem. Utilizing partial least square-canonical correlation analysis [15] is a feasible alternative, yet, it relies heavily on heterogeneous feature fusion, which is time-consuming. Semi-Coupled Two-Stream Fusion ConvNets [17] is a recently proposed method for eLR recognition. During training, this semi-coupled network takes both eLR video and its corresponding HR video as input, while during testing, only the eLR video is required. Using HR videos for training reduces its scalability. ***Motivation.*** On the one hand, eLR video is susceptible to noise, because a single pixel in eLR video may correspond to a large region in its corresponding HR video. On the other hand, eLR video cannot be cropped or resized randomly as HR video due to the limitation of resolution. These considerations raise demands for robust eLR video representation and new network training strategy.

2.1. Overview

In the proposed framework (*see* Fig. 1), eLR video is first preprocessed via a pseudo low rank regularized sub-network. This sub-network performs voxel-level prediction. Then a video classification network is employed to recognize the actions. For better convergence, a totally data-driven strategy is employed to initialize the classification network. In the rest of the paper, Section 3 illustrates the pseudo low rank regularized network for robust video representation, Section 4 demonstrates the newly proposed initialization strategy, Section 5 describes the experimental details, and Section 6 concludes the paper.

3. Robust video representation

For getting a robust representation $\mathcal{O}$, a typical method is low rank regularization [31]. Nevertheless, it is difficult to directly optimize a network subject to

$$\min_{\mathcal{O}} \mathcal{L}^R(\mathcal{O}) = \min \|\mathcal{O}\|_{\circledast}. \tag{1}$$

Here, $\|\cdot\|_{\circledast}$ is the tensor nuclear norm [31]. Instead, this paper transforms the objective function from minimizing the tensor nuclear norm to approximate a low rank label $\mathcal{B}$ via

$$\min_{\mathcal{O}} \mathcal{L}^A(\mathcal{B}, \mathcal{O}) = \min \|\mathcal{B} - \mathcal{O}\|_p^p. \tag{2}$$

Herein $p > 0$ is a constant, thus the loss function can be backpropagated and the sub-network is called *p*seudo Low Rank Network (*p*LRN). The low rank label $\mathcal{B}$ can be obtained through the following proposed Video Low Rank Representation (VLRR) model.

3.1. Video low rank representation

Due to the fact that low resolution videos are easily affected by noises, *e.g.*, occlusion and motion blur, we expect to seek robust features for videos. Fortunately, according to early researches [31,32], tensor low rank is a capable regularization for obtaining robust representations. Therefore, we proposed VLRR (Video Low Rank Representation) to get a robust "low rank video representation". For a given video, a simple foreground prior is that the moving object is sparse and a basic assumption of the background is that it is low rank [30]. Combining these two priors with the fact that the eLR video frames are easily affected by noises, a direct formulation of VLRR can be described as

$$\begin{aligned} &\min_{\mathcal{B},\mathcal{F},\mathcal{E}} \|\mathcal{B}\|_{\circledast} + \lambda\|\mathcal{F}\|_1 + \gamma\|\mathcal{E}\|_{2,1} \\ &\;s.t.\; \mathcal{V} = \mathcal{B} + \mathcal{F} + \mathcal{E} \end{aligned} . \tag{3}$$

Here $\mathcal{V}$, $\mathcal{B}$, $\mathcal{F}$ and $\mathcal{E}$ are the eLR video, the low rank background, the sparse foreground and the frame-level noise, respectively. Therefore, the prerequisite of VLRR is the assumption that "video = low rank background + sparse foreground + noise". This comes from the basic cognition that within a short video clip, the neighbor surrounding (background) often varies little and the action agent (foreground) usually occupies a small proportion of the video. This leads to the description that the "background is of low rank" and the "foreground is sparse". Consequently, the goal of VLRR is to extract robust low rank representations for eLR video action recognition, and the obtained "low rank background" is regarded as the "low rank video representation". The sparse "foreground" is an auxiliary restriction because the obtained low rank "background" will be inaccurate without any prior assumption. The tensor nuclear norm $\|\cdot\|_{\circledast}$ restricts the background to be low rank, tensor 1-norm $\|\cdot\|_1$ limits the foreground to be sparse, and tensor 2,1-norm $\|\cdot\|_{2,\,1}$ depicts the frame-level noise. To solve this problem, the augmented Lagrangian function of Eq. (3) is formulated as

$$\begin{aligned} \mathcal{L}(\mathcal{B},\mathcal{F},\mathcal{E},\mathcal{X}) = \|\mathcal{B}\|_{\circledast} + \lambda\|\mathcal{F}\|_1 + \gamma\|\mathcal{E}\|_{2,1} + \langle \mathcal{X}, \mathcal{V} - \mathcal{B} - \mathcal{F} - \mathcal{E}\rangle \\ + \frac{\mu}{2}\|\mathcal{V} - \mathcal{B} - \mathcal{F} - \mathcal{E}\|_F^2, \end{aligned} \tag{4}$$

where $\mathcal{X}$ is the Lagrange multiplier and μ is a positive penalty scalar. This formulation can be solved using alternating methods by keeping one item fixed at each iteration.

Update $\mathcal{B}$. Fix $\mathcal{F}$, $\mathcal{E}$ and $\mathcal{X}$, the $\mathcal{B}$-subproblem can be reformulated as

$$\min_{\mathcal{B}} \|\mathcal{B}_{k+1}\|_{\circledast} + \frac{\mu_k}{2}\|\mathcal{B}_{k+1} - \left(\mathcal{V} - \mathcal{F}_k - \mathcal{E}_k + \frac{\mathcal{X}_k}{\mu_k}\right)\|_F^2. \tag{5}$$

Suppose $\mathcal{M}_k = \mathcal{V} - \mathcal{F}_k - \mathcal{E}_k + \frac{\mathcal{X}_k}{\mu_k}$, according to Hu et al. [30], the globally optimal solution to Eq. (5) is given by the tensor singular value convoluting

$$\mathcal{B}_{k+1} = \mathcal{C}_\tau(\mathcal{M}_k) = \mathcal{U} \times_t \mathcal{C}_\tau(\mathcal{S}) \times_t \mathcal{U}^T, \tag{6}$$

where $\mathcal{U} \times_t \mathcal{S} \times_t \mathcal{U}^T = \mathcal{M}_k$ is the tensor singular value decomposition of $\mathcal{M}_k$, $\times_t$ is the tensor *t*-product, and $\mathcal{C}_\tau$ is the tensor convoluting operator.

Update $\mathcal{F}$. Fix $\mathcal{B}$, $\mathcal{E}$ and $\mathcal{X}$, the $\mathcal{F}$-subproblem [31] can be rewritten as

$$\min_{\mathcal{F}} \lambda\|\mathcal{F}_{k+1}\|_1 + \frac{\mu_k}{2}\|\mathcal{F}_{k+1} - \left(\mathcal{V} - \mathcal{B}_{k+1} - \mathcal{E}_k + \frac{\mathcal{X}_k}{\mu_k}\right)\|_F^2. \tag{7}$$

The closed-form solution for Eq. (7) is

$$\mathcal{F}_{k+1} = \max\left(0, \mathcal{P}_k - \lambda/\mu_k\right) + \min\left(0, \mathcal{P}_k + \lambda/\mu_k\right), \tag{8}$$

where $\mathcal{P}_k = \mathcal{V} - \mathcal{B}_{k+1} - \mathcal{E}_k + \mathcal{X}_k/\mu_k$.

Update $\mathcal{E}$. Fix $\mathcal{B}$, $\mathcal{F}$ and $\mathcal{X}$, the $\mathcal{E}$-subproblem [32] can be solved via

$$\min_{\mathcal{E}} \gamma\|\mathcal{E}_{k+1}\|_{2,1} + \frac{\mu_k}{2}\|\mathcal{E}_{k+1} - \left(\mathcal{V} - \mathcal{B}_{k+1} - \mathcal{F}_{k+1} + \frac{\mathcal{X}_k}{\mu_k}\right)\|_F^2. \tag{9}$$

And the closed-form solution for Eq. (9) is

$$\mathcal{E}_{k+1}(:,i,:) = \begin{cases} \dfrac{\|\mathcal{Q}_k\|_F - \gamma/\mu_k}{\|\mathcal{Q}_k\|_F}\mathcal{Q}_k(:,i,:) & \text{if } \|\mathcal{Q}_k\|_F > \frac{\gamma}{\mu_k}, \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

where $\mathcal{Q}_k = \mathcal{V} - \mathcal{B}_{k+1} - \mathcal{F}_{k+1} + \mathcal{X}_k/\mu_k$, $i = 1, \cdots, h$.

Update $\mathcal{X}$. The Lagrange multiplier is updated through

$$\mathcal{X}_{k+1} = \mathcal{X}_k + \mu_k(\mathcal{V} - \mathcal{B}_{k+1} - \mathcal{F}_{k+1} - \mathcal{E}_{k+1}). \tag{11}$$

Both the low rank background $\mathcal{B}$ and the sparse foreground $\mathcal{F}$ can be employed for eLR action recognition. Section 5 demonstrates its superiority to original eLR video $\mathcal{V}$. Specifically, $\mathcal{B}$ and $\mathcal{F}$ are taken as the desired auxiliary output of *p*LRN, which restricts *p*LRN to learn an approximate low rank auxiliary output.

3.2. Pseudo low rank network

The architecture of *p*LRN is demonstrated in Fig. 2. *p*LRN consists of several 3D convolution, 3D deconvolution and residual concatenation units. Considering the extreme low resolution of the input video, there is only one 3D maxpooling layer. For better preservation of the input details, there are two residual concatenation units. Both of the input video and the middle layer feature maps are transformed to concatenate with the corresponding feature map.

For a given video $\mathcal{V}$, *p*LRN is desired to learn its low rank representation $\mathcal{B}$. Practically, the desired low rank representation is exactly the low rank background recovered by VLRR. Nevertheless, simply learning a low rank representation at voxel-level is of great challenge. Therefore, to achieve action recognition, *p*LRN is jointly learned with a classification network *C*. Denote the combination of *p*LRN and *C* as *p*LRNC, the low rank representation is taken as an auxiliary output, and the main output is the predicted action label. In summary, given the eLR video as input, *p*LRNC outputs a predicted action label $\boldsymbol{p}$ and meanwhile an auxiliary output $\mathcal{O}$. In experiments, we can also restrict the auxiliary output $\mathcal{O}$ to fit a sparse $\mathcal{F}$. Suppose the desired auxiliary output (i.e., the output of VLRR) is represented as $\mathcal{D}$ and the action label is $\boldsymbol{y}$. Thus the loss function of *p*LRNC is defined as

$$\mathcal{L}^P = \sigma\;\mathcal{L}^A(\mathcal{D},\mathcal{O}) + (1-\sigma)\;\mathcal{L}^C(\boldsymbol{p},\boldsymbol{y}) \tag{12}$$

where $\mathcal{L}^A(\mathcal{D},\mathcal{O})$, $\mathcal{D} \in \{\mathcal{B},\mathcal{F}\}$ is defined as Eq. (2), $\mathcal{L}^C(\boldsymbol{p},\boldsymbol{y})$ is the categorical crossentropy loss, $\boldsymbol{p}$ and $\boldsymbol{y}$ are the corresponding predicted and groundtruth label, respectively.

3.3. Feasibility analysis

It is proved to be effective [30] for tasks, *e.g.*, moving object detection, to learn low rank representations of videos. Benefitting from the low rank assumption, the learned representation is more robust to noise. *p*LRN introduces low rank assumption into convolution neural networks. Instead of learning an exact low rank representation directly, *p*LRN aims to recover an approximate low rank representation pre-obtained by VLRR. Then the objective function changes from TNN minimization to l_p-norm minimization. This makes *p*LRN an end-to-end network.

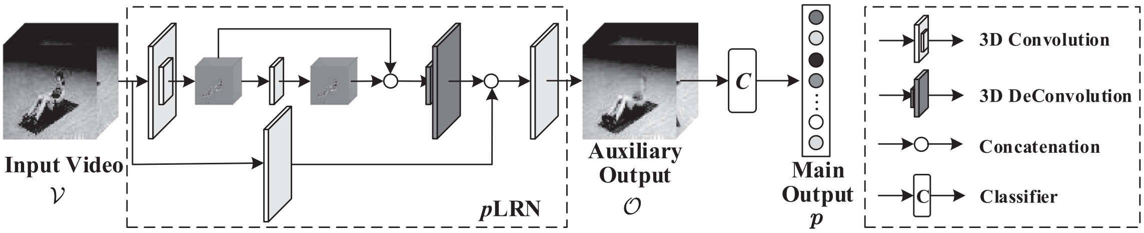


Fig. 2. The semantic architecture of *p* LRN. The output of *p*LRN is to approximate the low rank background $\mathcal{B}$ or the sparse foreground $\mathcal{F}$ pre-obtained through VLRR.

*p*LRN is also effective for large dataset. Specifically, the expense for precomputing $\mathcal{B}$ and $\mathcal{F}$ using VLRR is proportional to the number of videos. And it takes only 0.2 s for a video clip using an Intel i7 CPU. The expected time-consumption for large datasets, *e.g.*, UCF[1], ActivityNet[2] and Kinetics[3] are approximately 0.3, 0.6 and 6 h, respectively. Compared with the long training time of networks, *e.g.*, several days, the precomputing time can be ignored. Also note that one of the key point of *p*LRNC is the auxiliary low rank guidance. *p*LRNC is not the first contribution to explore the effect of auxiliary guidance. For example, Li et al. [39] leverages weak semantic relevance as the auxiliary guidance for event classification, and Pan et al. [40] employs visual-semantic embedding as the auxiliary guidance for video captioning. These contributions demonstrate its feasibility and effectiveness.

Considering VLRR, at each iteration, $\mathcal{B}$, $\mathcal{F}$ and $\mathcal{E}$ have closed form solutions. For gray videos, where the channel number c equals 1, suppose w, h and t are the corresponding video width, height and number of frames, then the computation complexity is $\mathcal{O}(wht\log(t) + 2wh^2t)$.

4. Initialization for classification network

After getting the low rank representation of the video, a classification network is designed to recognize the actions being taken place. The noised video $\mathcal{V}$, the low rank background $\mathcal{B}$ and the sparse foreground $\mathcal{F}$ can be implemented as the input of the classification network. For simplicity, $\mathcal{V}$, $\mathcal{B}$ and $\mathcal{F}$ are not treated distinguished later in this section. For the given video clips $\{\mathcal{V}_i\}_{i=1}^N$, $\mathcal{V}_i \in \mathbb{R}^{w\times h\times t\times c}$, suppose there are S convolutional layers, the numbers of convolution kernels for each layer are s_1, s_2, $\cdots$, s_S, and the size of convolution kernel is $x\times y\times z$.

4.1. Unsupervised convolutional kernel learning

For each $\mathcal{V}_i$, the method starts by selecting a small cube of the same size as the convolution kernel around each voxel, and then this cube is slid within each video clip. The sliding cube values are collected and flatten into a vector $\mathbf{v}_k \in \mathbb{R}^{xyz}$. After padding with zeros and overlapping sliding (including *temporal* padding and *temporal* sliding), there will be wht vectors $\mathbf{v}_k$, $k = 1, \cdots, wht$. Then for each video clip $\mathcal{V}_i$, these vectors can be processed with mean-removing by the mean vector $\bar{\mathbf{v}}$ and stacked into a large matrix

$$\mathbf{V}_i = \left[\hat{\mathbf{v}}_1, \hat{\mathbf{v}}_2, \cdots, \hat{\mathbf{v}}_{wht}\right] \in \mathbb{R}^{xyz\times wht}. \tag{13}$$

Here $\hat{\mathbf{v}}_k = \mathbf{v}_k - \bar{\mathbf{v}}$ is the mean-removed vector. For all N video clips, the stacked matrix is

$$\mathbf{V} = [\mathbf{V}_1, \mathbf{V}_2, \cdots, \mathbf{V}_N] \in \mathbb{R}^{xyz\times Nwht}. \tag{14}$$

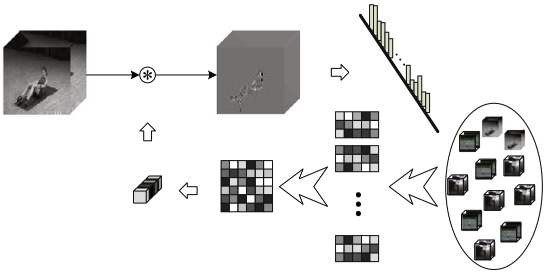


Fig. 3. Illustration of TenneT. TenneT learns convolution kernels without any back-propagation in an unsupervised manner.

And the covariance matrix is obtained via $\mathbf{S_v} = \mathbf{V}\mathbf{V}^T$. According to theories about PCA [41] and PCANet [42], here an optimal projection direction $\mathbf{W}$ can be obtained via

$$\mathbf{W} = \arg\max_{\mathbf{W}\mathbf{W}^T=\mathbf{I}} tr\left(\mathbf{W}^T\mathbf{S_v}\mathbf{W}\right). \tag{15}$$

Using techniques of Singular Value Decomposition (SVD), the first s_1 principle components of $\mathbf{S_v}$, denoted as $\mathbf{q}_1, \mathbf{q}_2, \cdots, \mathbf{q}_{s_1}, \mathbf{q}_i \in \mathbb{R}^{xyz}$, can be reshaped as s_1 kernel cubes of size $x\times y\times z$. Then these kernel cubes are selected as the convolutional kernels for TenneT. Without loss of comprehension, for simplicity, these kernel cubes are denoted as $\mathcal{Q}^s \in \mathbb{R}^{x\times y\times z}$, $s = 1, \cdots, s1$.

Then the first layer output can be defined as

$$\mathcal{M}_i^s = \mathcal{V}_i * \mathcal{Q}^s, i = 1, \cdots, N; s = 1, \cdots, s1, \tag{16}$$

where $*$ denotes 3D convolution. Similarly, the subsequent layers can be defined equally.

The convolution kernels are learned via Tensor factorization, thus the neTwork is called TenneT. On the one hand, the learned convolution kernels $\mathcal{Q}^{s_i}$ can be employed to initialize a 3D convolution network with the same structure as TenneT. On the other hand, the final output of TenneT are of size $N\times w\times h\times t\times s_S$, which is hard for further analysis. Different from PCANet, this paper adopts Bag of Features to reduce the dimension of feature maps (here the outputs of convolution layers are also called feature maps). Specifically, suppose the values at the same spatial-temporal position construct a feature vector of length s_S. $\mathcal{K}$-means clustering is employed to learn a code book and its corresponding high level representation. In this case, TenneT can be employed directly for action recognition using SVM. A more intuitive illustration can be found in Fig. 3.

4.2. Feasibility analysis

Learning a network for eLR action recognition is difficult. This is primarily due to the fact that eLR video is hard to be augmented using techniques such as random crop. TenneT learns the convolution kernels directly from the data distribution without any back-propagation, thus it is feasible and efficient. The most related work is PCANet [42], nevertheless, TenneT differs from PCANet in two aspects. (***a***) PCANet is designed for image classification, while

[1] http://www.crcv.ucf.edu/data/UCF101.php.

[2] http://www.activity-net.org/.

[3] https://www.deepmind.com/research/open-source/open-source-datasets/kinetic.

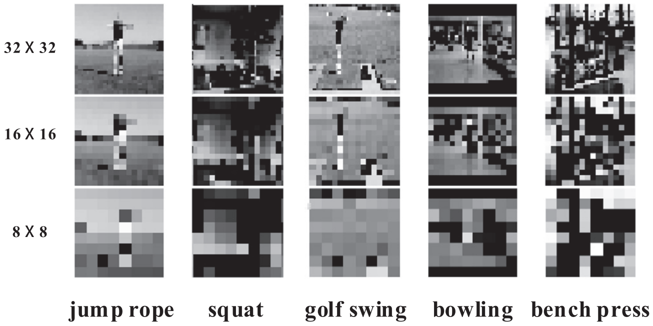


Fig. 4. Examples of eLR Videos in Penn Dataset. This paper mainly deals with resolution of 16×16, which is even hard for humans to recognize the actions.

TenneT is able to handle a sequence of video frames. (***b***) More importantly, TenneT is used for initializing a deep 3D classification network with identical architecture. TenneT makes the network converges faster than random initialization. And it will be discussed in next section.

TenneT is also effective for high resolution video and large datasets. Though the size of $\mathbf{V} \in \mathbb{R}^{xyz \times Nwht}$ is large for large dataset, TenneT only requires computing the SVD of the covariance matrix $\mathbf{S_v} = \mathbf{V}\mathbf{V}^T \in \mathbb{R}^{xyz \times xyz}$ instead of $\mathbf{V}$. Usually, $x = y = z = 3$. As a result, TenneT is efficient for large datasets.

5. Experiments

This section evaluates the performance of the proposed method on two benchmark datasets, i.e., Penn [43], HMDB [44]. The effectiveness of each component, i.e., VLRR, *p*LRN and TenneT, is first analyzed. And then a comparison with other state-of-the-art algorithms is reported. For reproducible research, codes has been released[4].

5.1. Datasets

Penn[5] is a challenging action recognition dataset with large variations in viewpoint, scale, background, illumination, camera motion and temporal duration. It contains of 2326 video sequences categorized into 15 action classes.

HMDB-51[6] is a more challenging action recognition dataset with 6849 videos divided into 51 human action classes. Compared with Penn, the videos are extracted from commercial movies as well as YouTube, thus HMDB-51 is more challenging.

The widths of video frames vary from 270 to 482 pixels, while the heights of video frames vary from 204 to 480 pixels. Most frame resolutions are 480×270 and 480×360. These videos are rescaled into low resolutions, *e.g.*, 32×32, 16×16 and 8×8. Some examples can be found in Fig. 4. Without specific illustration, the spatial resolution is set to be $\mathbf{16} \times \mathbf{16}$ and the temporal length is 16.

5.2. Analysis of VLRR and *p*LRN

5.2.1. Implementation details

VLRR. λ and γ are critical for recovery, and in this paper, λ and γ are both set to be $1/\sqrt{\max\{w, h\} \times t}$ [31,32]. The penalty term $\mu = 1e^{-4}$, $\mu_{\max} = 1e^{10}$, the update parameter $\delta = 1.1$ and the converge tolerance $\tau = 1e^{-8}$.

Table 1
Recognition Accuracy of VLRR and *p*LRN on Penn and HMDB. *p*LRN outperforms VLRR in both datasets.

Methods	VLRR				*p*LRN			
	$\mathcal{V}$	$\mathcal{B}$	$\mathcal{F}$	$\mathcal{B}+\mathcal{F}$	$\mathcal{V}$	$\mathcal{B}$	$\mathcal{F}$	$\mathcal{B}+\mathcal{F}$
Penn	20.5	21.3	21.5	22.6	20.9	23.7	23.9	24.0
HMDB	8.8	8.9	8.8	10.1	8.9	9.2	8.9	11.3

***p*LRN**. The network is implemented on *Keras* [45]. The weight of auxiliary output σ is 0.1 while the weight of the main output is 0.9. Without loss of generality, the classification network is set to be a *two-layer fully connected network* in this subsection. Each layer consists of 4096 hidden units, with dropout ratio 0.9. *Adadelta* with $lr = 1.0$, $\rho = 0.95$ and $\epsilon = 1e^{-8}$ is employed to optimize the network.

5.2.2. Convergence of VLRR

This paper solves VLRR iteratively via alternating method. Fig. 5 demonstrates its convergence on Penn and HMDB under various resolutions. The rank error, as demonstrated in Fig. 5 the vertical axis, converges within 50 iterations.

5.2.3. Performance of VLRR and *p*LRN

This section demonstrates the performance of VLRR and *p*LRN, and the results are reported in Table 1. As demonstrated in Section 5.2.1, for simplicity and without loss of generality, the classification network is set to be a *two-layer fully connected network* (FCNet).

Table 1 illustrates five points. (***a***) Performances of the recovered background $\mathcal{B}$ and foreground $\mathcal{F}$ are better than that of the original video $\mathcal{V}$. This is probably because the eLR video $\mathcal{V}$ is easily affected by noise. The recovered two components $\mathcal{B}$ and $\mathcal{F}$ are more robust since the noise term $\mathcal{E}$ is removed. (***b***) A combination of $\mathcal{B}$ and $\mathcal{F}$ boosts the performance than both of the two terms. From the perspective of two-stream network, $\mathcal{F}$ behaves as optical flow while $\mathcal{B}$ represents multiple static frames. Intuitively, these two components are complimentary to each other. (***c***) *p*LRN outperforms VLRR. Generally, VLRR decomposes the two components totally from the video without any other information. Whereas *p*LRN learns an auxiliary output with the guidance of VLRR, and a main output of the predicted label. Both of the two terms are optimized jointly. With the main objective of improving recognition accuracy, *p*LRN is superior to VLRR. (***d***) The results of *p*LRN without any supervision of VLRR are shown in the sixth row (row $\mathcal{V}$ of *p*LRN). It is inferior to those with VLRR supervision. Nevertheless, they perform better than using the original video. (***e***) $\mathcal{F}$ is superior to $\mathcal{B}$ for Penn, while $\mathcal{F}$ is worse than $\mathcal{B}$ for HMDB. HMDB is a rather challenging dataset even with high resolution. It consists of large object and background shift, thus the background also contains valuable motion information, which can be employed for action recognition. Whereas for Penn, the background is static and the foreground possesses more information.

Fig. 6 presents an intuitive illustration about rank error and recognition accuracy. As training goes, the rank error gets smaller (i.e., the rank drops down) and the testing accuracy gets higher.

5.2.4. Visualization of VLRR and *p*LRN

Fig. 7 visualizes a given video and its corresponding recovered foreground $\mathcal{F}$ and background $\mathcal{B}$. The results obtained by VLRR and *p*LRN are both shown. The background $\mathcal{B}$ depicts the static information as multiple frames, and the foreground $\mathcal{F}$ describes the motion detail as the optical flow. Under the framework of two-stream network, this corresponds to the results reported in Table 1. In general, there is no straightforward distinct among rows (b), (c)

[4] https://www.github.com/Tsingzao/pLRN.

[5] http://www.dreamdragon.github.io/PennAction/.

[6] http://www.serre-lab.clps.brown.edu/resource/hmdb-a-large-human-motion-database/.

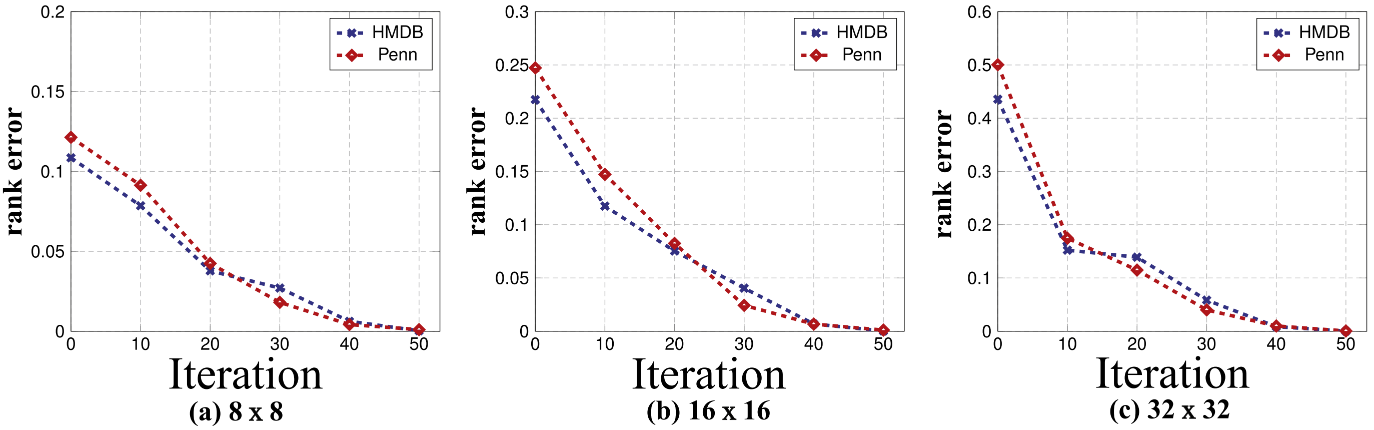


Fig. 5. Rank error vs. iteration curves. Two datasets, i.e., Penn and HMDB, conditioned on three resolutions, i.e., 32 × 32, 16 × 16 and 8 × 8, are reported.

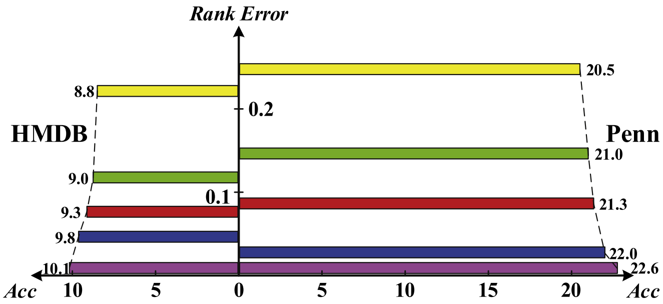


Fig. 6. Rank error vs. testing accuracy on both Penn and HMDB datasets.

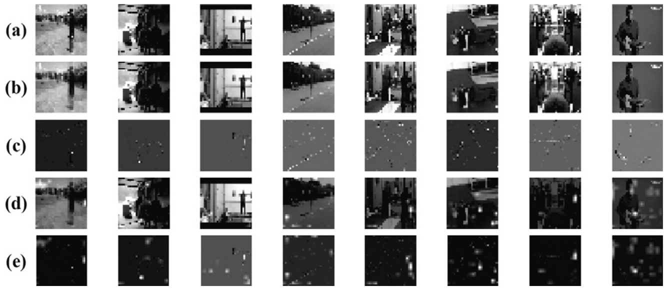


Fig. 7. Visualization of VLRR and *p*LRN. The first row (a) represents the original video, the second row (b) and the third row (c) are the low rank background $\mathcal{B}$ and the sparse foreground $\mathcal{F}$ obtained by VLRR, while the fourth row (d) and the fifth row (e) are the results of *p*LRN.

Table 2
Accuracy of C3D and TenneT. TenneT initialization for C3D outperforms random initialization.

Methods	Penn		HMDB	
	$\mathcal{V}$	$\mathcal{B}+\mathcal{F}$	$\mathcal{V}$	$\mathcal{B}+\mathcal{F}$
C3D	26.0	35.8	12.9	14.1
C3D*	28.3	36.3	14.3	15.9
TenneT	29.4	38.1	15.0	17.9

*Note the results are re-implemented using TenneT for initialization.

and rows (d), (e), which indicates that *p*LRN indeed learns a low rank and a sparse component.

5.3. Analysis of TenneT

5.3.1. Implementation details

For better TenneT initialization, this section employs a two-layer 3D convolution network for eLR recognition. The number of convolution layers and convolution kernels are set to be $S = 2$ and $s_i = 10,\ i = 1, 2$, and the kernel size is set to be 3 × 3 × 3. For TenneT, it employs SVM with $C = 10$ for classification. The size of code book is set to be 256. *SGD* is employed for C3D network optimization. The parameters are set without cross validation in this section.

5.3.2. Performance of TenneT

This section evaluates both the effectiveness of TenneT for eLR action recognition and the feasibility of TenneT for network initialization. The results are shown in Table 2. $\mathcal{B}$ and $\mathcal{F}$ are obtained using VLRR.

Compared with Table 1, C3D outperforms FCNet at a great deal because C3D depicts more spatial-temporal correlation. Furthermore, the two-stream framework integrating background $\mathcal{B}$ and foreground $\mathcal{F}$ boosts the performance about 9% on Penn. Using TenneT as the initialization strategy is superior to random initialization in testing accuracy for both Penn and HMDB. This is due to the property that TenneT learns convolution kernels totally from the data distribution. In particular, TenneT with SVM classifier achieves better performance than the former two (C3D and C3D*) methods. The reason is that C3D requires more parameters in fully connected layers, which is more likely to be overfitting due to the lack of training data.

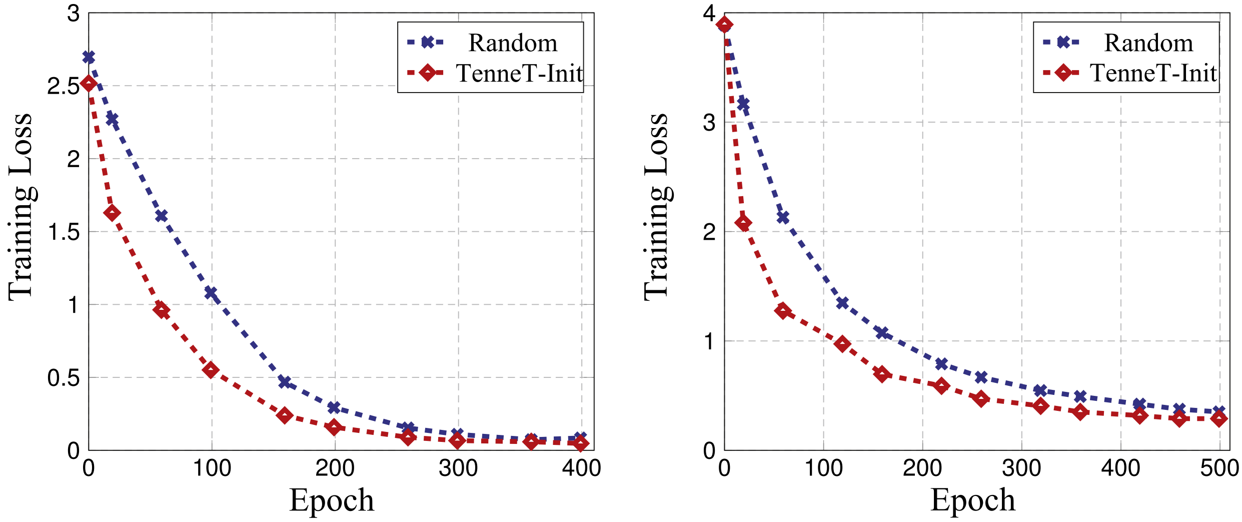
5.3.3. Convergence analysis of TenneT

This section demonstrates the advantage that using TenneT for network initialization promotes the convergence. A comparison of *with* and *without* TenneT initialization for both Penn and HMDB is shown in Fig. 8.

Random initialization converges at about 300 and 410 epochs while TenneT initialization converges at about 250 and 370 epochs for Penn and HMDB, respectively. Combined with the results in Table 2, TenneT initialization outperforms random initialization both in speed and accuracy.

5.3.4. Visualization of TenneT

Fig. 9 visualizes the learned convolution kernels for TenneT. In Fig. 9, there are two convolution layers. From Fig. 9, the unsupervised learned kernels are similar to the basis of Discrete Wavelet Transform, i.e., DWT. DWT is originally proposed for signal processing, e.g., foreground detection [46], and graph decomposition [47]. It defines a set of bases that can represent an arbitrary function [48]. In fact, researches have demonstrated that Wavelets are naturally appropriate for analysis of biological data or bio-inspired strategy [49]. TenneT agrees with this mechanism.



(a) Penn

(b) HMDB

Fig. 8. Training loss vs. epoch curves on both Penn and HMDB. TenneT initialization converges faster than random initialization.

Table 3
Recognition accuracy of TenneT compared with other methods on Open Domain Action Recognition (ODAR) dataset. The resolution is set to be 32 × 32. Results show TenneT is more suitable to open domain analysis.

Method		ODAR					Average
		Weizmann	URALD	UIUC	MSR	UCFARG	
Pixel level feature	Average pooling + SVM	40.0	79.7	59.8	22.7	72.3	54.9
	Max pooling + SVM	49.0	83.7	49.5	24.6	78.1	57.0
Deep CNN feature	FCNet	60.0	81.4	64.3	35.6	78.1	63.9
	CIFARNet	63.6	80.9	63.1	36.4	80.0	64.8
	AlexNet	65.4	82.5	56.0	36.7	79.0	63.9
	C3D[11]	**67.3**	82.9	79.0	**37.8**	80.1	69.4
	C3D*	65.4	85.2	75.5	36.7	**81.9**	68.9
Unsupervised feature	PCANet[42] + SVM	63.6	83.1	67.2	29.0	79.0	64.4
	TenneT + SVM	**67.3**	**94.9**	**87.0**	36.8	80.0	**73.2**

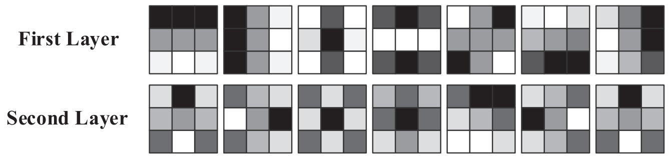


Fig. 9. Visualization of the convolutional kernels of TenneT.

5.3.5. Additional experiment for TenneT

Additionally, this section conducts an extra experiment for demonstrating the effectiveness of TenneT. Table 3 presents the results on ODAR dataset. ODAR[7] is an open domain action recognition dataset composed of several small datasets, i.e., Weizmann, URALD, UIUC, MSR and UCFARG. In this section, the spatial resolution is set to be 32 × 32.

From Table 3, methods that employ 3D convolution are superior to other 2D convolution methods. This is because a sequence of video frames is quite different from a series of temporal independent images. Taking temporal correlation into consideration boosts the performance. Note that TenneT is mainly inspired by PCANet, Table 3 also makes a comparison with PCANet. Nevertheless, PCANet can not take temporal information into consideration, thus it deals with multiple frames via simple temporal pooling. On the contrary, TenneT takes full consideration of temporal relevance by 3D convolution, thus TenneT is more suitable for eLR action recognition than PCANet.

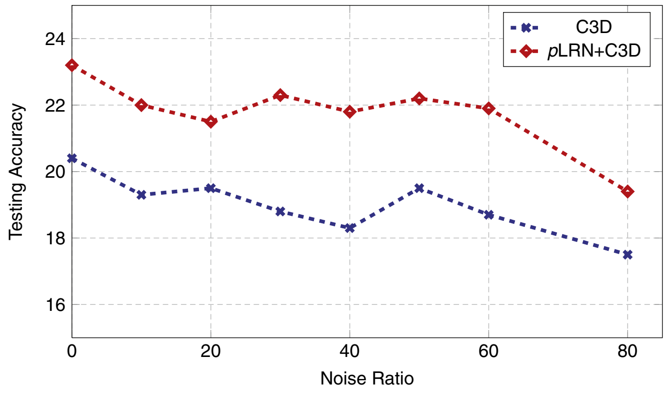
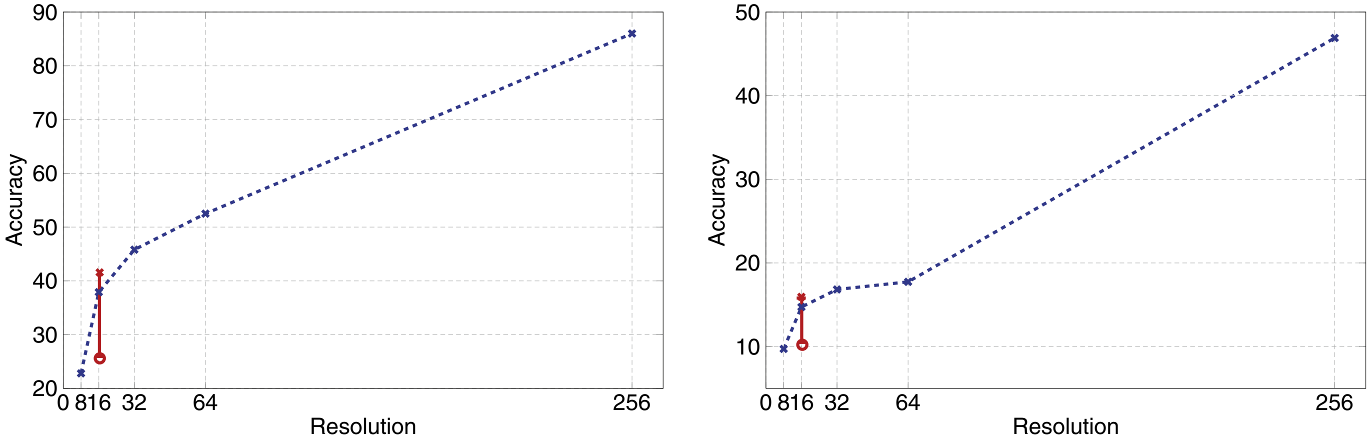


Fig. 10. Accuracy vs. NR curve on Penn. Benefiting from the pseudo low rank guidance, *p*LRN+C3D is more robust than C3D.

5.4. Analysis of noise and resolution

This section demonstrates the robustness of *p*LRN, and illustrates the capability of TenneT in handling different resolutions. Note that in this section, the networks are trained within 100 epochs without cross-validation.

[7] http://www.sesame.comp.nus.edu.sg/workshop/odar2017/.



(a) Penn

(b) HMDB

Fig. 11. Accuracy vs. resolution curve of TenneT on both Penn and HMDB. Recognition accuracy decreases rapidly when resolution is smaller than 16 × 16.

5.4.1. *Analysis of noise*

Fig. 10 describes the performance of C3D and *p*LRN + C3D with different noise ratio *NR*. The noise ratio describes the percentage of noised frames defined by $NR = \frac{nF}{aF}$, where *nF* is the number of noised frames and *aF* is the number of all frames. Generally, without the auxiliary regularization of pseudo low rank, C3D is susceptible to noise.

In fact, the VLRR model is designed to remove additional noises as illustrated in Eq. (3). And *p*LRN aims to generate an approximate output of VLRR. Thus it is expected to be robust to noise. Specifically, *p*LRN has an additional skip connection from the input layer to the output layer (*see* Fig. 2). At the worst case, *p*LRN+C3D is competitive against single C3D by learning an identity mapping. As the increasing of noise ratio, both C3D and *p*LRN+C3D are getting worse performance on testing accuracy. Nevertheless, with the guidance of pseudo low rank, the performance of *p*LRN+C3D decreases in a much slower tendency, especially when the noise ratio is less than 60%.

5.4.2. *Analysis of resolution*

Typically, low resolution is the main challenge that affects action recognition. Fig. 11 illustrates the performance of TenneT with different resolutions. Principally, the proposed method concentrate on eLR video action recognition, while it is still effective for high resolution videos.

From Fig. 11, the results of resolution 64 × 64 and 32 × 32 are acceptable compared with resolution 256 × 256 when taking other conditions, *e.g.*, memory storage, into consideration. When the resolution is smaller than 16 × 16, *e.g.*, 8 × 8, the recognition accuracy drops rapidly. As have shown in Fig. 4, it is even hard for human to recognize the actions with resolution smaller than 16 × 16. To demonstrate that the proposed method is effective irrespective of aspect ratio, this subsection also report the recognition accuracy on videos with unequal width and height. Note that the action agent, i.e., human body, often lies in a tall and thin rectangle area, this subsection mainly consider videos of size 16 × 8 and 32 × 16. Specifically, the videos are first resized into 16 × 16 and 32 × 32, and then cropped to 16 × 8 and 32 × 16. The results are also shown in Fig. 10. The red circles denote the testing accuracy under 16 × 8 and the red crosses represent the testing accuracy under 32 × 16. Interestingly, the testing accuracy is dominant by $\min\{width, \ height\}$. For example, the result of 16 × 8 is close to 8 × 8, and the result of 32 × 16 is close to 16 × 16. One possible explanation is that the discriminative information video contained is restricted by $\min\{width, \ height\}$. To keep the basic aspect ratio, as illustrated in the previous paragraph, the videos are cropped from 16 × 16 and 32 × 32 to 16 × 8 and 32 × 16. This rough strategy might omit the discriminative background information.

Table 4
Testing accuracy on Penn and HMDB.

Methods	OF	HR	TI	Penn	HMDB
C3D[11]	–	–	✓	37.1	14.3
ConvNet+SVM[16]	–	–	✓	–	18.9
Two-Stream[12]	✓	–	✓	41.0*	19.6
ConvNet++ISR+SVM[16]	–	✓	✓	–	20.8
SCN[17]	✓	✓	✓	44.9*	21.4
PCANet[42]+SVM	–	–	–	28.3	12.7
VLRR+FC	–	–	–	22.6	10.1
*p*LRN+FC	–	–	–	24.0	11.3
VLRR+C3D	–	–	✓	35.8	14.1
*p*LRN+C3D	–	–	✓	43.7	20.1
VLRR+TenneT	–	–	✓	38.1	17.9
*p*LRN+TenneT	–	–	✓	**47.1**	**21.7**

*The re-implemented results. Here OF represents optical flow, HR denotes high resolution data and TI indicates temporal information.

5.5. *Comparison with state-of-the-art methods*

This section compares the proposed VLRR, *p*LRN, TenneT with other methods, *e.g.*, C3D [11], Two-Stream [12], and Semi-Coupled Network [17] *etc.*, on both Penn and HMDB. The results are shown in Table 4. This section uses five-fold cross-validation for parameter selection.

C3D and Two-Stream are two most widely used methods for action recognition. Semi-Coupled Network (SCN) and Inverse Super Resolution (ISR) network are two recently proposed methods for eLR recognition. However, they need high resolution videos for training. From Table 4, an integration of *p*LRN and TenneT outperforms others methods.

6. Conclusion

This paper has proposed a new method for eLR action recognition. Basically, the proposed method contains three components, a video low rank representation (VLRR) model, a pseudo low rank network (*p*LRN) and a new data-driven network initialization strategy (TenneT). Extensive experiments demonstrate that the proposed method is effective to eLR action recognition. Compared with other methods, the proposed method is more robust benefitting from VLRR and *p*LRN, and it converges much faster due to the introduction of TenneT initialization.

Nevertheless, though the proposed methods are still effective for high resolution videos, the time consumption of VLRR can not be overlooked. Our future work will focus on seeking a more efficient video low rank representation model. And a more direct method of imposing low rank regularization to deep networks is under construction. Besides, the proposed TenneT learns convolution kernels based on tensor factorization, and this is time-consuming. The future work will focus on a more efficient method without tensor factorization.

Acknowledgments

This work was supported by the National Natural Science Foundation of China [grant numbers 61773377, 61573352, 91646207, 61620106003].

References

[1] M. Ma, N. Marturi, Y. Li, A. Leonardis, R. Stolkin, Region-sequence based six-stream CNN features for general and fine-grained human action recognition in videos, Pattern Recognit. 76 (2018) 506–521.

[2] G. Sigurdsson, S. Divvala, A. Farhadi, A. Gupta, Asynchronous temporal fields for action recognition, in: International Conference on Computer Vision and Pattern Recognition, 2017, pp. 585–594.

[3] X. Wang, M. Wang, W. Li, Scene-specific pedestrian detection for static video surveillance, IEEE Trans. Pattern Anal. Mach. Intell. 36 (2) (2014) 361–374.

[4] S. Liu, C. Wang, R. Qian, H. Yu, R. Bao, Surveillance video parsing with single frame supervision, in: International Conference on Computer Vision and Pattern Recognition, 2017, pp. 1529–1538.

[5] C. Li, Z. Huang, Y. Yang, J. Cao, X. Sun, H. Shen, Hierarchical latent concept discovery for video event detection, IEEE Trans. Image Process. 26 (5) (2017) 2149–2162.

[6] N. Hussein, E. Gavves, A. Smeulders, Unified embedding and metric learning for zero-exemplar event detection, in: International Conference on Computer Vision and Pattern Recognition, 2017, pp. 1529–1538.

[7] L. Cao, X. Zhang, W. Ren, K. Huang, Large scale crowd analysis based on convolutional neural network, Pattern Recognit. 48 (10) (2015) 3016–3024.

[8] D. Sam, S. Surya, V. Babu, Switching convolutional neural network for crowd counting, in: International Conference on Computer Vision and Pattern Recognition, 2017, p. 6.

[9] I. Laptev, T. Lindeberg, Space-time interest points, in: International Conference on Computer Vision, 2003, pp. 432–439.

[10] H. Wang, C. Schmid, Action recognition with improved trajectories, in: International Conference on Computer Vision, 2013, pp. 3551–3558.

[11] D. Tran, L. Bourdev, R. Fergus, L. Torresani, M. Paluri, Learning spatiotemporal features with 3d convolutional networks, in: International Conference on Computer Vision, 2015, pp. 4489–4497.

[12] K. Simonyan, A. Zisserman, Two-stream convolutional networks for action recognition in videos, in: Neural Information Processing System, 2014, pp. 568–576.

[13] C. Feichtenhofer, A. Pinz, A. Zisserman, Convolutional two-stream network fusion for video action recognition, in: International Conference on Computer Vision and Pattern Recognition, 2016, pp. 1933–1941.

[14] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, L. Van Gool, Temporal segment networks: Towards good practices for deep action recognition, in: European Conference on Computer Vision, 2016, pp. 20–36.

[15] H. Chen, X. Zhao, S. Sun, M. Tan, Pls-cca heterogeneous features fusion-based low-resolution human detection method for outdoor video surveillance, Int. J. Autom. Comput. 14 (2) (2017) 136–146.

[16] M. Ryoo, B. Rothrock, C. Fleming, H. Yang, Privacy-preserving human activity recognition from extreme low resolution., in: AAAI Conference on Artificial Intelligence, 2017, pp. 4255–4262.

[17] J. Chen, J. Wu, J. Konrad, P. Ishwar, Semi-coupled two-stream fusion convnets for action recognition at extremely low resolutions, in: Winter Conference on Applications of Computer Vision, 2017, pp. 139–147.

[18] M. Jaderberg, A. Vedaldi, A. Zisserman, Speeding up convolutional neural networks with low rank expansions, arXiv preprint arXiv:1405.3866 (2014).

[19] C. Tai, T. Xiao, Y. Zhang, X. Wang, Convolutional neural networks with low-rank regularization, arXiv preprint arXiv:1511.06067 (2015).

[20] Y. Ioannou, D. Robertson, J. Shotton, R. Cipolla, A. Criminisi, Training cnns with low-rank filters for efficient image classification, arXiv preprint arXiv: 1511.06744 (2015).

[21] N. Chesneau, K. Alahari, C. Schmid, Learning from web videos for event classification, IEEE Trans. Circuits Syst. Video Technol. (2017).

[22] S. Lee, W. Baddar, Y. Ro, Collaborative expression representation using peak expression and intra class variation face images for practical subject-independent emotion recognition in videos, Pattern Recognit. 54 (2016) 52–67.

[23] H. Jégou, M. Douze, C. Schmid, P. Pérez, Aggregating local descriptors into a compact image representation, in: International Conference on Computer Vision and Pattern Recognition, 2010, pp. 3304–3311.

[24] F. Ning, D. Delhomme, Y. LeCun, F. Piano, L. Bottou, P. Barbano, Toward automatic phenotyping of developing embryos from videos, IEEE Trans. Image Process. 14 (9) (2005) 1360–1371.

[25] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, L. Feifei, Large-scale video classification with convolutional neural networks, in: International Conference on Computer Vision and Pattern Recognition, 2014, pp. 1725–1732.

[26] S. Ji, W. Xu, M. Yang, K. Yu, 3D convolutional neural networks for human action recognition, IEEE Trans. Pattern Anal. Mach. Intell. 35 (1) (2013) 221–231.

[27] Y. Shi, T. Furlanello, A. Anandkumar, Compact tensor pooling for visual question answering, arXiv preprint arXiv:1706.06706(2017).

[28] J. Kossaifi, A. Khanna, Z. Lipton, T. Furlanello, A. Anandkumar, Tensor contraction layers for parsimonious deep nets, arXiv preprint arXiv:1706.00439 (2017).

[29] F. Huang, A. Anandkumar, Unsupervised learning of word-sequence representations from scratch via convolutional tensor decomposition, arXiv preprint arXiv:1606.03153 (2016).

[30] W. Hu, Y. Yang, W. Zhang, Y. Xie, Moving object detection using tensor-based low-rank and saliently fused-sparse decomposition, IEEE Trans. Image Process. 26 (2) (2017) 724–737.

[31] C. Lu, J. Feng, Y. Chen, W. Liu, Z. Lin, S. Yan, Tensor robust principal component analysis: Exact recovery of corrupted low-rank tensors via convex optimization, in: International Conference on Computer Vision and Pattern Recognition, 2016, pp. 5249–5257.

[32] P. Zhou, J. Feng, Outlier-robust tensor pca, in: International Conference on Computer Vision and Pattern Recognition, 2017, pp. 2263–2271.

[33] D. Carroll, J. Chang, Analysis of individual differences in multidimensional scaling via an n-way generalization of eckart-young decomposition, Psychometrika 35 (3) (1970) 283–319.

[34] L. Tucker, Some mathematical notes on three-mode factor analysis, Psychometrika 31 (3) (1966) 279–311.

[35] M. Kilmer, C. Martin, Factorization strategies for third-order tensors, Linear Algebra Appl. 435 (3) (2011) 641–658.

[36] M. Ben, A. Zomet, S. Nayar, Video super-resolution using controlled subpixel detector shifts, IEEE Trans. Pattern Anal. Mach. Intell. 27 (6) (2005) 977–987.

[37] C. Liu, D. Sun, A bayesian approach to adaptive video super resolution, in: International Conference on Computer Vision and Pattern Recognition, 2011, pp. 209–216.

[38] Y. Huang, W. Wang, L. Wang, Video super-resolution via bidirectional recurrent convolutional networks, IEEE Trans. Pattern Anal. Mach. Intell. 13 (9) (2017) 1–14.

[39] L. Chao, C. Jiewei, H. Zi, Z. Lei, H.-T. Shen, Leveraging weak semantic relevance for complex video event classification, in: IEEE International Conference on Computer Vision, 2017, pp. 3667–3676.

[40] P. Yingwei, M. Tao, Y. Ting, L. Houqiang, R. Yong, Jointly modeling embedding and translation to bridge video and language, in: IEEE International Conference on Computer Vision and Pattern Recognition, 2016, pp. 4594–4602.

[41] K. Peason, On lines and planes of closest fit to systems of point in space, Philos. Mag. 2 (11) (1901) 559–572.

[42] T. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, Y. Ma, Pcanet: a simple deep learning baseline for image classification? IEEE Trans. Image Process. 24 (12) (2015) 5017–5032.

[43] W. Zhang, M. Zhu, K. Derpanis, From actemes to action: A strongly-supervised representation for detailed action understanding, in: International Conference on Computer Visison, 2013, pp. 2248–2255.

[44] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, T. Serre, Hmdb: a large video database for human motion recognition, in: Internation Conference on Computer Visison, 2011, pp. 2556–2563.

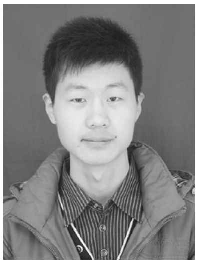
[45] F. Chollet, Keras, 2015, (https://www.github.com/fchollet/keras).

[46] L. Shuai, F. Dinei, L. Wanqing, Z. Yaqin, C. Cook, A fusion framework for camouflaged moving foreground detection in the wavelet domain, IEEE Trans. Pattern Anal. Mach. Intell. (2018). doi: 10.1109/TPAMI.2017.2780248.

[47] J. Zeng, G. Cheung, A. Ortega, Bipartite approximation for graph wavelet signal decomposition, IEEE Trans. Signal Process. 65 (20) (2017) 5466–5480.

[48] C. Boris, Z. Damjan, Directional 3d wavelet transform based on gaussian mixtures for the analysis of 3d ultrasound ovarian volumes, IEEE Trans. Image Process. (2018). doi: 10.1109/TIP.2018.2828329.

[49] E. Bullmore, J. Fadili, V. Maxim, L. Şendur, B. Whitcher, J. Suckling, M. Brammer, M. Breakspear, Wavelets and functional magnetic resonance imaging of the human brain, Neuroimage 23 (2004) S234–S249.



Tingzhao Yu received the B.S. degree in automation from Ocean University of China, Qingdao, China, in 2013, the M.S. degree in pattern recognition and intelligent systems from Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2016, and he is currently working toward the Ph.D. degree in National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China. His current research interest is large scale video understanding.

Lingfeng Wang received the B.S. degree in computer science from Wuhan University, Wuhan, China, in 2007. He received the Ph.D. degree from the Institute of Automation, Chinese Academy of Sciences in 2013. He is currently an associate professor with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. His research interests include computer vision and image processing.

Chaoxu Guo received the B.S. degree in automation from South China University of Technology, Guangzhou, China, in 2017. He is currently working toward the Ph.D. degree in National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China. His current research interests include computer vision and video analysis.

Huxiang Gu received the B.S. degree in computer science from Beihang University, Beijing, China, in 2010. He received the Ph.D. degree from the Institute of Automation, Chinese Academy of Sciences in 2016. He is currently an assistant professor with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. His research interests include computer vision and data mining.

Shiming Xiang received the B.S. degree in mathematics from Chongqing Normal University, Chongqing, China, in 1993, the M.S. degree from Chongqing University, Chongqing, China, in 1996, and the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2004. From 1996 to 2001, he was a Lecturer with the Huazhong University of Science and Technology, Wuhan, China. He was a Postdoctoral Researcher with the Department of Automation, Tsinghua University, Beijing, China, until 2006. He is currently a Professor with the Institute of Automation, Chinese Academy of Sciences. His current research interests include pattern recognition and machine learning.

Chunhong Pan received the B.S. degree in automatic control from Tsinghua University, Beijing, China, in 1987, the M.S. degree from Shanghai Institute of Optics and Fine Mechanics, Chinese Academy of Sciences, Shanghai, China, in 1990, and the Ph.D. degree in pattern recognition and intelligent systems from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2000. He is currently a Professor with the Institute of Automation, Chinese Academy of Sciences. His current research interests include computer vision, image processing, computer graphics, and remote sensing.