Large Margin Dimensionality Reduction for Action Similarity Labeling

Xiaojiang Peng, Yu Qiao, Qiang Peng and Qionghua Wang

Abstract—Action recognition in videos is receiving extensive research interest due to its wide applications. This task needs to assign a specific action class for each video. In this paper, we study the problem of action similarity labeling (ASLAN) that is to verify whether two action videos present the same type of action or not. We show that both Fisher vector (FV) and vector of locally aggregated descriptors (VLAD) with dense trajectory features can achieve state-of-the-art performance on the ASLAN benchmark. Our main contribution is to develop a large margin dimensionality reduction (LMDR) method to compress high-dimensional FV and VLAD. Specially, we leverage the hinge loss objective function and stochastic gradient descent to optimize the discriminative projection matrix of these vectors. Extensive experiments on the ASLAN dataset indicate that our LMDR method not only reduces the dimension significantly but also improves the verification performance.

Index Terms—Action similarity labeling, large margin dimensionality reduction, VLAD, Fisher vector, similarity learning.

I. INTRODUCTION

HUMAN action analysis in videos has become a highly active research area due to its wide applications in video surveillance, human-computer interface, and content based video retrieval [1], [2], [3], [4], [5], [6]. There exist several tasks in this area such as action recognition [2], [4], action detection [7] and Action Similarity Labeling (ASLAN) [8]. This article addresses the ASLAN task: given a pair of videos, we wish to decide whether the videos present the same type of action or not.

Performance in the ASLAN task mainly depends on video representations and the similarity measure used to compare video pairs. For video representation, most of the methods in the ASLAN task followed those in the action recognition [8], [9], [10], [11], such as the popular bag-of-words (BoW) [12] representation with Space–Time Interest Points (STIPs) [1] and dense trajectories [4]. Ori et al. [8] provided the baseline performance using the STIP features with the BoW model, and improved it by introducing metric learning [9]. Ori et al. [10] proposed motion interchange patterns (MIP) with the BoW model for action recognition and action similarity labeling. Yair et al. [11] presented several variants of MIP, namely histMIP and DoGMIP, which replace the original patches employed in MIP by the Histogram of Gradient Orientations (HOG) and Difference of Gaussian (DoG), respectively. Yair et al. [11] also combined these variants of MIP with dense trajectory and the Motion Boundary Histograms (MBH) features, and achieved good results in both action recognition and ASLAN tasks. For similarity measure of video pairs, Ori et al. [8] applied 12 basic (dis-)similarities in the baseline experiments, and introduced the One-Shot-Similarity Metric Learning (OSSML) method [13] for the ASLAN task [9]. Both Ori et al. [9], [10] and Yair et al. [11] employed the Cosine Similarity Metric Learning (CSML) [14] after performing a PCA dimension reduction on the BoW representation.

Recent studies indicate that advanced feature encoding methods other than vector quantization (i.e., hard assignment [12]) can steadily improve action recognition performance [15]. Among these methods, Fisher vector (FV) [16] and vector of locally aggregated descriptors (VLAD) [17] are probably the two most effective approaches. One limitation of using FV and VLAD for ASLAN is their high dimension, which not only implies in a large computational cost but also may harm verification performance.

In this paper, we develop a large margin dimensionality reduction (LMDR) method to deal with this problem, which jointly performs dimensionality reduction and similarity learning. Our LMDR leverages the hinge loss function and stochastic gradient descent to optimize the projection matrix. Experiments show that our LMDR method not only can reduce the dimension significantly but also can improve the performance of both FV and VLAD on the ASLAN benchmark.

II. METHODOLOGY

The framework of our action similarity labeling approach is shown in Fig. 1. For a pair of videos, i) we extract the improved dense trajectories which are described by concatenating the HOG, Histogram of Optical Flow (HOF), and MBH descriptors; ii) we encode the descriptors in each video by VLAD (or FV) using a codebook pre-trained by K-means or Gaussian Mixture Models (GMM) and pool them to yield video-level representations; iii) we apply the proposed LMDR method to project these representations to a discriminative subspace; iv) we transform the two compact video representations to a single video-pair representation by performing
point-wise multiplications, which is better than directly using their (dis)similarities [8] in our test. Finally, a binary SVM is learned to map the video-pair representation to a binary decision of same/not-same as [10].

A. Encoding Methods Revisited

**VLAD.** Jégou et al. proposed VLAD in [17]. Similar to standard BoW, a dictionary $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, ..., \mathbf{d}_K] \in \mathbb{R}^{d \times K}$ is first learned by $K$-means from training data. Let $\mathbf{X}_n = [\mathbf{x}_1, ..., \mathbf{x}_N] \in \mathbb{R}^{d \times N}$ denote a set of local descriptors from a video. For each word $\mathbf{d}_i$, a vector $\mathbf{v}_i$ is yielded by aggregating the differences between the assigned features and $\mathbf{d}_i$:

$$\mathbf{v}_i = \sum_{\mathbf{x}_j: N(N(\mathbf{x}_j) = i)} (\mathbf{x}_j - \mathbf{d}_i),$$

where $NN(\mathbf{x}_j) = i$ denotes that the nearest neighborhood of $\mathbf{x}_j$ within $\mathbf{D}$ is $\mathbf{d}_i$. The VLAD representation is the concatenation of all the $d$-dimensional vectors $\mathbf{v}_i$ and is therefore a $Kd$ dimensional vector.

**Fisher vector.** For Fisher vectors, we assume the generation process of local descriptors $\mathbf{X}$ can be modeled by a probability density function $p(\cdot; \theta)$ with parameter $\theta$. The gradient of the log-likelihood w.r.t a parameter can describe how that parameter contributes to the generation process of $\mathbf{X}$ [18]. The probability density function is usually modeled by Gaussian Mixture Model (GMM). An improved version of Fisher vectors proposed by Perronnin et al. [16] is as follows,

$$G_{\mu k}^{\mathbf{X}} = \frac{1}{N \sqrt{2 \pi} \sigma_k} \sum_{n=1}^{N} \gamma_n(k) \left( \frac{\mathbf{x}_n - \mu_k}{\sigma_k} \right)^2,$$

$$G_{sk}^{\mathbf{X}} = \frac{1}{N \sqrt{2 \pi} \sigma_k} \sum_{n=1}^{N} \gamma_n(k) \left( \frac{\mathbf{x}_n - \mu_k}{\sigma_k} \right)^2 - 1,$$

where $\gamma_n(k)$ is the weight of local feature $\mathbf{x}_n$ for the $i$-th Gaussian [16]. The final Fisher vectors representation is the concatenation of all the $G_{\mu k}^{\mathbf{X}}$ and $G_{sk}^{\mathbf{X}}$ which is a $2Kd$ dimensional super vector.

B. Large Margin Dimension Reduction

Both VLAD and FV representations are high-dimensional, which can be storage-consuming and time-consuming for further processing. To obtain compact and discriminative video representations, we propose the larger margin dimensionality reduction method to compress these high-dimensional vectors.

Without loss of generality, we consider the VLAD representation here, and it can be generalized to FV. Suppose the VLAD for video pairs are $\phi_i$ and $\phi_j$, respectively. We aim to learn a discriminative projection matrix $U \in \mathbb{R}^{p \times Kd}, p \ll Kd$, which projects $\phi \in \mathbb{R}^{Kd}$ to low-dimensional one $U \phi \in \mathbb{R}^p$, such that the inner product $< U \phi_i, U \phi_j >$ between the representations of video pairs is larger than a learnt threshold $b$ if the video pairs present the same type of action, and smaller otherwise. We further impose that these conditions are satisfied with a margin of at least one w.r.t $b$ (see the 4th column in Fig. 1), resulting in the constraints:

$$y_{ij}(\phi_i^T U^T U \phi_j - b) > 1, \quad y_{ij} \in \{+1, -1\}$$

where $y_{ij} = 1$ denotes the video pairs contain the same type of action, and $y_{ij} = -1$ otherwise.

To learn the projection matrix $U$, we leverage the hinge-loss function and optimize the following objective function:

$$\arg \min_{U, b} \sum_{(i,j)} \max\{1 - y_{ij}(\phi_i^T U^T U \phi_j - b), 0\}$$

Note that Eq.(5) is different from standard SVM, as it is a quadratic optimization about matrix $U$. The optimal $\{U, b\}$ can be found by stochastic sub-gradient descent method. At each iteration $t$, we randomly sample a batch of video pairs $\mathcal{V}$ and perform the following update for the projection matrix:

$$U_{t+1} = \begin{cases} U_t, & \text{if } y_{ij}(\phi_i^T U_t^T U \phi_j - b) > 1, \forall (i,j) \in \mathcal{V} \\ U_t + \lambda \sum_{(i,j)}(-2y_{ij}\phi_i^T U \phi_j)U, & \text{otherwise} \end{cases}$$

where $(-2y_{ij}\phi_i^T U \phi_j)U$ is the sub-gradient from video pairs $(i,j)$ in the batch $\mathcal{V}$, and $\lambda$ is a given learning rate. $U$ is initialized by the $p$ largest PCA-Whitened dimensions $U_0$ [14], [10], [19], and $b$ by the mean of $< \phi_i U_0, U_0 \phi_j >$ empirically. The final $b$ is discarded, and we keep the projection $U$. Generally, $U$ can be regularized by $\|U\| = 1$ to prevent scaling $U$. We call it as Constrained LMDR (C-LMDR). An evaluation is given in Section III-B.

**Relation to previous methods.** Our LMDR learns a discriminative projection matrix to increase the similarities of video pairs with “same” annotations and meanwhile separate
those video pairs with “not-same” annotations by a large margin. This is closely related to CSML [14] and distance metric learning of large margin nearest neighbor (LMNN) [20], [21], [22]. The CSML encourages the margin between positive and negative samples to be as large as possible, which tries to maximize the following objective:

$$g(U) = \sum_{y_{ij}=1} CS(U\phi_i, U\phi_j) - \alpha \sum_{y_{ij}=-1} CS(U\phi_i, U\phi_j)$$

(7)

where $CS(x, y) = x^T y / \|x\| \|y\|$ denotes cosine similarity. We also use similarity measure here, but unlike CSML that ensures the margin between the sum of positive and negative samples, we try to guarantee the margin between each positive and negative sample to be larger than 2 (see Eq.(4) and Fig.1).

LMNN learns a full-rank Mahalanobis matrix based on a distance measure [20]. This is impractical in our case since the dimension of FV is more than 100,000. We note there exists another method also named as LMDR in [22]. It jointly learned the Mahalanobis matrix and L1-norm SVM for image classification. The L1-norm SVM served as feature selection which can lead to low dimension. This method suffers the same problem as LMNN.

III. EXPERIMENTS

A. Experimental Protocol and Setup

To validate the proposed method, we conduct extensive experiments on the widely-used ASLAN benchmark [8]. The ASLAN dataset contains 3,631 action videos collected from the web, to a total of 432 action classes. The protocol used for the experiments is the 10-fold leave-one-out cross-validation scheme. The dataset is divided into ten splits, each of which includes 300 pairs of same-type actions and 300 pairs of different-type actions. In each experiment, nine splits are used for training, with the remaining split used for testing. Results are reported by calculating a ROC curve and measuring both the area under curve (AUC) and the averaged accuracy ± standard errors for the ten splits.

We extract improved dense trajectories (IDT) using the code from Wang [4]. Each trajectory is described by concatenating HOG, HOF, and MBH descriptors, which is a 396-dimensional vector. We reduce the dimensionality of these descriptors to 200 by performing PCA and Whitening, and fix the codebook size to 256 for both VLAD and FV as usual in the literature [4]. Therefore, the original VLAD and FV are 51,200 and 102,400 dimensional, respectively. The batch size of stochastic sub-gradient descent is fixed to 10 and $\lambda$ to 0.01.

### Table I

<table>
<thead>
<tr>
<th>Compression Ratio</th>
<th>Dim. (Energy)</th>
<th>LMDR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,055</td>
<td>11 (10%)</td>
<td>63.60 ± 0.19 (69.65)</td>
</tr>
<tr>
<td>826</td>
<td>62 (20%)</td>
<td>63.77 ± 0.12 (72.37)</td>
</tr>
<tr>
<td>275</td>
<td>186 (30%)</td>
<td>63.72 ± 0.16 (72.58)</td>
</tr>
<tr>
<td>139</td>
<td>369 (40%)</td>
<td>66.63 ± 0.26 (76.93)</td>
</tr>
<tr>
<td>85</td>
<td>600 (70%)</td>
<td>66.83 ± 0.25 (75.26)</td>
</tr>
<tr>
<td>59</td>
<td>874 (60%)</td>
<td>67.13 ± 0.24 (75.38)</td>
</tr>
<tr>
<td>43</td>
<td>1,194 (70%)</td>
<td>66.42 ± 0.35 (73.46)</td>
</tr>
<tr>
<td>1</td>
<td>51,200 (100%)</td>
<td>Baseline (Full Dim.): 61.38 ± 0.25 (66.39)</td>
</tr>
</tbody>
</table>

### Table II

<table>
<thead>
<tr>
<th>Compression Ratio</th>
<th>Dim. (Energy)</th>
<th>LMDR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,389</td>
<td>19 (10%)</td>
<td>63.10 ± 0.50 (68.98)</td>
</tr>
<tr>
<td>883</td>
<td>116 (20%)</td>
<td>63.40 ± 0.83 (69.30)</td>
</tr>
<tr>
<td>354</td>
<td>289 (30%)</td>
<td>63.43 ± 0.81 (69.30)</td>
</tr>
<tr>
<td>200</td>
<td>513 (40%)</td>
<td>63.43 ± 0.82 (69.30)</td>
</tr>
<tr>
<td>132</td>
<td>775 (50%)</td>
<td>68.72 ± 0.95 (75.26)</td>
</tr>
<tr>
<td>96</td>
<td>1,067 (60%)</td>
<td>64.33 ± 1.52 (72.98)</td>
</tr>
<tr>
<td>1</td>
<td>102,400 (100%)</td>
<td>Baseline (Full Dim.): 63.13 ± 0.03 (69.28)</td>
</tr>
</tbody>
</table>

Fig. 2. The performance of LMDR for FV and VLAD with changing iterations. $p = 300$ for FV and $p = 600$ for VLAD. C-LMDR denotes the constrained LMDR. The proposed LMDR achieves best performance at iteration 10k for both FV and VLAD.

### Table III

<table>
<thead>
<tr>
<th>Constraint on $U$</th>
<th>LMDR</th>
</tr>
</thead>
<tbody>
<tr>
<td>$CS(U\phi_i, U\phi_j)$</td>
<td>63.13 ± 0.15 (69.28)</td>
</tr>
</tbody>
</table>

### B. Results and Discussion

We first study the effect of changing dimensionality (based on the main energy from initialized PCA-Whitened matrix) for LMDR on VLAD and FV. We fix the iterations to 10k.

**Dimensionality.** For fair comparison, we take the original VLAD and FV as baselines for Table I and Table II, respectively. As shown in both Tables, for both VLAD and FV, our LMDR improves the baseline performance significantly with large compression ratio. Specially, it improves AUC by 7.14% with compression rate 59 for VLAD, and 6.12% with compression ratio 200 for FV. The performance changes slightly when the PCA preserving energy reaches 30%. Our LMDR method for VLAD obtains the best performance with compression rate 59, and for FV with compression ratio about 200. This indicates that VLAD and FV representations are highly redundant on the ASLAN dataset. Compared to PCA, our LMDR not only reduces the dimensionality significantly but also boosts the performance even with thousands of reduction rates.

**Iterations.** We also investigate the effect of iterations for the LMDR method in Fig. 2. We fix the $p$ of our LMDR approach to 600 and 300 for VLAD and FV, respectively. The iteration 0 means that the PCA-Whitened matrix is used which is the initialization for $U$. The optimal iteration for both VLAD and FV is 10k as can be observed from Fig. 2. Too many iterations may lead to overfitting which decreases the performance.

**Constraint on $U$.** To evaluate the impact of constraints on $U$, we conduct an experiment using C-LMDR with VLAD. We apply augmented Lagrangian method to solve the constrained optimization [23]. The results with varying iterations are shown in Fig. 2. We observe that the results of VLAD using both C-LMDR and LMDR are very similar. Compared with LMDR, C-LMDR needs more computational cost.

To gain further insight into our results, we illustrate the most confident predictions made by our LMDR method with...
show that the proposed method achieves superior performance for both FV and VLAD. Experimental results not only reduce the dimension significantly but also improve the performance for action similarity labeling. Our method can be implemented following [10]. We learn a full rank projection matrix for CSML, and therefore the projection matrix from CSML is available at http://mmlab.siat.ac.cn/personal/pxj/.

### Comparisons to the State-of-the-Art Results

<table>
<thead>
<tr>
<th>Method</th>
<th>No LMDR/CSML</th>
<th>PCA+CSML</th>
<th>LMDR</th>
</tr>
</thead>
<tbody>
<tr>
<td>VLAD (%)</td>
<td>61.35±2.25 (66.30)</td>
<td>66.18±2.05 (70.28)</td>
<td>67.13±2.34 (73.53)</td>
</tr>
<tr>
<td>FV (%)</td>
<td>63.15±2.03 (69.28)</td>
<td>66.17±2.83 (72.38)</td>
<td>68.37±3.07 (75.40)</td>
</tr>
</tbody>
</table>

TABLE IV

### Conclusion

This paper proposes a large margin dimensionality reduction method to compress the Fisher vector and VLAD representation for action similarity labeling. Our method can not only reduce the dimension significantly but also improve the performance for both FV and VLAD. Experimental results show that the proposed method achieves superior performance than state-of-the-art methods on ASLAN benchmark. Our code is available at http://mmlab.siat.ac.cn/personal/pxj/.

### References