EFFICIENT POOLING OF IMAGE BASED CNN FEATURES FOR ACTION RECOGNITION IN VIDEOS

Biplab Banerjee

Department of Computer Sc. & Engg.,
Indian Institute of Technology Roorkee,
Roorkee, India

Vittorio Murino

Pattern Analysis & Computer vision (PAVIS),
Istituto Italiano di Tecnologia,
Genova, Italy

ABSTRACT

In this paper, we propose a new video representation incorporating image based deep features and an efficient pooling strategy for the purpose of action recognition. The Convolutional Neural Network (CNN) based features have very recently emerged as the new state of the art for image classification. Several attempts have been made to extend such CNN models for videos by explicitly focusing on the temporal evolution of the frames. Feature pooling is one such approach which represents video sequences in terms of some statistical properties of the feature dimensions over the frames. However, traditional pooling strategies including max or average pooling explicitly fail to capture the temporal progression of the frame-level contents. In contrast to previous pooling techniques, we propose a two-level video representation which separately focuses on the entire video as well as a number of video sub-volumes. In both levels, we introduce a generic time series pooling on the frame-level deep CNN features efficiently. Further, a self-tuning spectral clustering is considered to highlight video snippets which are highly probable to contain significant sub-action sequences. We validate the proposed feature encoding on the challenging KTH-actions and UCF-50 datasets and find that the proposed encoding outperforms traditional pooling based feature representations by substantial margin.

***Index Terms*—** Action recognition, Convolutional Neural Network, Spectral Clustering

1. INTRODUCTION

Representation learning for sequential data is an active field of research in many applications (action recognition, multi-temporal image analysis etc.) due to the continuous generation of unbounded sequential data from diverse sources. In the last few years, learned deep CNN based representations have exhibited extraordinary recognition performance for images [1]. Nonetheless, it still remains a challenge on how to extend such deep models to sequential data with minimum supervision.

However, direct application of such CNN models to video frames completely fails to capture the dynamics of the video frame sequence. There are some recent deep models (recurrent network/LSTM) [2] which can predict the evolution of the sequential data but such models require a large number of additional parameters and huge training data in order to capture the sequence information. Needless to mention, it is difficult and costly to manually annotate different action primitives of a given category. A comprehensive survey of deep models used for the purpose of action recognition can be obtained in [3].

One of the simplest ways to encode video streams with the CNN models is to apply max/average pooling over the frame-level features. However, such a naive approach fails to capture the time varying information over the frames and any random reorganisation of the frames produces identical feature encoding [4]. In addition, [5] proposes a time-series pooling strategy for first-person videos by highlighting the positive and negative components of the feature gradients. Though our encoding model is inspired by [5], we find that this method does not capture the fine-grained information from the videos. Considering that important action components are likely to occur at certain intervals, it is expected that better recognition performance can be achieved by highlighting such localized snippet-level features in addition to video based features.

In view of the above, we show how a pre-trained image based CNN can be used to extract video descriptors combining appearance and motion information through efficient time-series feature pooling. Since training a CNN with images is much cheaper than with videos, therefore, construction of video features by exploring an image based CNN accounts for major saving in time and effort. Specifically, we highlight the noteworthy contributions of this paper as follows:

- We employ the unsupervised self-tuned spectral clustering [6] technique to temporally cluster video frames into consistent sub-volumes (snippets). The self-tuned clustering does not require prior information regarding the number of available groups and inherently estimates the same by aligning the Eigenvectors of the

graph laplacian to a canonical vector space. Hence, the video snippets can be obtained without any prior knowledge.

- We propose a two-level video representation by coupling both the video-level and snippet-level features efficiently. The cogent snippets are identified as a few selected clusters from the previous spectral clustering stage which depict substantial variations in frame contents. In both levels, our feature encoding is based on average pooling the positive and negative components of the frame-level feature gradients. In contrast to [5] which is based on sum-pooling the positive and negative feature gradient components over the frames, we highlight both the global and local information from the video. In addition, our frame level CNN features consider both the global frame features as well as the features extracted from a few local motion salient region proposals [7]. This substantially reduces the effects of irrelevant frame contents.

- It is observed that the proposed feature encoding exhibits enhanced action recognition performance ($\approx 2-5\%$) in comparison to the traditional pooling based features from the literature for the challenging UCF-50 [8] and KTH-actions [9] datasets.

To the best of our knowledge, there are a few very recent endeavors which aim at solving the problem of temporal pooling of frame based deep features for action recognition [10] [11]. Such techniques mostly subsume the rank pooling operation in a trainable end-to-end network. We instead exploit an already learned model to encode the videos.

2. PROPOSED METHODOLOGY

This section details the proposed action recognition pipeline. Broadly, it consists of two stages: feature extraction from videos and classification. The classification is carried out by multi-class Support Vector Machines (SVM) with linear kernel.

For notational convenience, let us consider that $V = \{V_1, V_2, \ldots, V_M\}$ represents a set of videos each depicting one of L different human activities. Let us also consider $V_i = \{v_{i1}, v_{i2}, \ldots, v_{iT_i}\}$ represents T_i frames of V_i. We describe the spectral clustering based snippet extraction followed by the proposed feature encoding in the following.

2.1. Unsupervised snippet extraction from videos

Given V_i, snippet extraction technique clusters V_i into a set of N_i temporally consistent sub-video volumes $\{C_{ij}\}_{j=1}^{N_i}$ ($N_i << T_i$) where each C_{ij} contains a subset of consecutive frames. Let us consider a weighted undirected temporal graph G_{V_i} which links consecutive pairs of frames in V_i and the edges of G_{V_i} are weighted by Euclidean distance between frame level features. We extend the idea of self-tuning spectral clustering [6] to cluster G_{V_i} in order to obtain C_{ij}s assuming N_i is unknown.

Standard spectral clustering techniques approximate the clustering in the original data space by clustering a few selected components of the Eigenvectors of the corresponding graph lalpacian. However, it requires the number of clusters as input. In contrary, the self tuned version inherently approximates the number of cluster in addition to considering a local scaling function for individual data items (frames). Specifically, the distance between two consecutive nodes (frames) of G_{V_i} is defined in the adjacency matrix A^i in terms of Gaussian kernel as: $A^i_{k,k+1} = exp(\frac{-d^2(v_k, v_{k+1})}{\sigma_k \sigma_{k+1}})$ where $\sigma_k = d(v_k, v_K)$ defines the scaling parameter for frame v_k as its distance to the K^{th} neighbour in the forward temporal scale. $d()$ represents the Euclidean distance between the CNN features of a pair of consecutive frames. It signifies the local statistics in the neighbourhood of v_k in the video sub-volume. Local affinity automatically finds the scales of data manifold and results in high affinities within clusters and low affinities between clusters.

The number of clusters is estimated by recovering an orthogonal rotation matrix that best aligns the subset of dominant Eigenvectors of the weighted laplacian of A^i to a canonical coordinate system. Initially we consider that the maximum number of snippet in V_i is at most $\frac{T_i}{5}$. Let X^i represents the subset of largest Eigenvectors of A^i stored in columns. Ideally, X^i is sparse block diagonal with blocks corresponding to different clusters. For each possible group number $1 \leq n \leq \frac{T_i}{5}$, we recover the rotation which best aligns X^i's columns with the canonical coordinate system and let $Z^i = X^i R$ denote the matrix obtained after rotation R. We focus on recovering R for which every row of Z^i will contain at most one non-zero entry. This can be expressed as a cost function (J) as:

$$J = \sum_{k=1}^{T_i} \sum_{j=1}^{n} \frac{Z_{kj}^2}{\max_j Z_{kj}^2} \tag{1}$$

and is minimized using standard gradient descent based approach. The optimal number of clusters is stipulated as the one providing minimum cost. Usually the optimization is performed incrementally by including a new Eigenvector in X^i in each run. It allows the initialization of the new iteration based on the previous one and thus saves time substantially. The clustering of the frames is obtained from the alignment result Z^i for the top N_i Eigenvectors and v_k is assigned to cluster n if $\max_j Z_{kj}^2 = Z_{kn}^2$.

The snippets (clusters in G_{V_i}) are further ranked based on the average edge weights of the frames within the snippets. A large average edge weight indicates the presence of multiple consecutive action representatives within the snippet. We

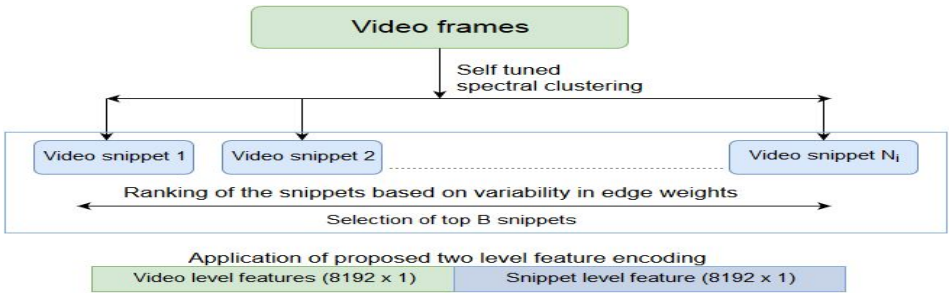

Fig. 1. The proposed feature encoding scheme

select B top ranked snippets expecting that they refer to important sub-action volumes.

2.2. Feature extraction from the videos

The proposed CNN features combine the appearance and motion information for the entire video into a fixed length feature vector (Figure 1). We initially calculate per frame deep features by incorporating global frame-level information in addition to localized region-level information.

A set of region proposals are extracted per frame v_k using Objectness [7]. We select a subset of r proposals which are less overlapping, have high Objectness scores and are of moderate size. Further, we ensure that the selected proposals refer to salient regions in the frame by thresholding their average pixel-level motion saliency scores. The multi-scale graph based saliency detection technique [12] is used for this purpose. We calculate CNN features (l_1 normalized 4096 dimensional output of $fc7$ of ImageNet-VGG-f) [13] for the entire frame as well as the selected region proposals separately. VGG-f has similar architecture to the AlexNet model but is computationally fast since it involves sparse connections between pairs of consecutive layers. The features for the frame under consideration are calculated by averaging the features of the frame as well as that of the selected region proposals. Such a feature representation better captures different aspects of the underlying action by focusing on motion salient parts. In particular, let $[f_1^k, f_2^k, \ldots, f_{4096}^k]$ be the frame-level CNN feature (for the k^{th} frame) whereas $\{[f_{R_1^\rho}^k, f_{R_2^\rho}^k, \ldots, f_{R_{4096}^\rho}^k]\}_{\rho=1}^r$ is a matrix of $r \times 4096$ dimensions representing the CNN features for a set of r region proposals extracted in the aforementioned manner. We define the cumulative frame level deep feature as:

$$\widehat{f}_{1:4096}^k = [average(f_1^k, \{f_{R_1^\rho}^k\}_{\rho=1}^r), average(f_2^k, \{f_{R_2^\rho}^k\}_{\rho=1}^r), \ldots, average(f_{4096}^k, \{f_{R_{4096}^\rho}^k\}_{\rho=1}^r)] \quad (2)$$

Hence, for $V_i = \{v_{i1}, v_{i2}, \ldots, v_{iT_i}\}$, T_i different 4096-dimensional feature vectors are calculated. Further, the feature gradient for a pair of consecutive frames (v_{k+1} and v_k) is computed in the following fashion:

$$\widehat{f}^\delta(k, k+1) = [\widehat{f}_1^{k+1} - \widehat{f}_1^k, \widehat{f}_2^{k+1} - \widehat{f}_2^k, \ldots, \widehat{f}_{4096}^{k+1} - \widehat{f}_{4096}^k] \quad (3)$$

$\widehat{f}^\delta$ highlights the changes in abstract latent concepts over the frames and inherently models the evolution of the frames throughout the extent of the video (snippets). We further segregate the positive and negative components of the l^{th} feature dimension ($1 \leq l \leq 4096$) of $\widehat{f}^\delta$ over all the T_i frames into two 4096×1 vectors as follows:

$$\widehat{f}_{l+}^\delta = \{\widehat{f}_l^{k+1} - \widehat{f}_l^k\}_{k=1}^{T_i-1} \quad (4)$$

if $\widehat{f}_l^{k+1} - \widehat{f}_l^k \geq 0$,

$$\widehat{f}_{l-}^\delta = \{\widehat{f}_l^{k+1} - \widehat{f}_l^k\}_{k=1}^{T_i-1} \quad (5)$$

if $\widehat{f}_l^{k+1} - \widehat{f}_l^k < 0$.

Ideally, we consider the l^{th} dimension of $\widehat{f}^\delta$ as a one dimensional signal and both the $\widehat{f}_{l+}^\delta$ and $\widehat{f}_{l-}^\delta$ characterize the positive and negative slopes of the signal. By unfolding $\widehat{f}^\delta$ into $[\widehat{f}_{l+}^\delta]_{l=1}^{4096}$ and $[\widehat{f}_{l-}^\delta]_{l=1}^{4096}$, it is possible to quantify the advancement of specific abstract concepts over the frames for a given action category.

As aforementioned, the proposed two-level video representation explicitly focuses on the entire video as well as a set of localized snippets. For the entire video, we average pool the dimensions of both the $[\widehat{f}_{l+}^\delta]_{l=1}^{4096}$ and $[\widehat{f}_{l-}^\delta]_{l=1}^{4096}$ and concatenate them in order to obtain a 8192×1 dimensional feature vector.

Further for the set of B snippets identified in the previous stage, we first average pool $[\widehat{f}_{l+}^\delta]_{l=1}^{4096}$ and $[\widehat{f}_{l-}^\delta]_{l=1}^{4096}$ specifically for the frames within each snippet separately in order to obtain $B \times 8192$ dimensional feature vectors. These features are further max-pooled to obtain another 8192×1 dimensional representation of the video. In total, each video is efficiently represented by a $16,384 \times 1$ dimensional feature vector combining the video-level (coarse) and snippet-level (fine) information.

It is plausible to use any of the levels of the proposed two-level hierarchy for the sake of action classification. However, in order to compactly represent the detailed characteristics of the video, it is suggested to use the combined representation.

Table 1. Performance analysis of the proposed feature encoding (in $\%$)

Dataset	Average pool	Max pool	Time series pool [5]	Proposed encoding
UCF-50	74.0	76.0	73.0	78.0
KTH	79.0	87.0	88.0	90.0

3. EXPERIMENTAL RESULTS

3.1. Experimental setup

We evaluate the performance of the proposed framework on two datasets: KTH-actions [9] and UCF-50 [8].

The UCF-50 dataset contains videos representing 50 human actions in unconstrained environments. The dataset contains a total of 6681 videos with about $100 - 150$ videos per category. This dataset is a superset of the popular UCF-11 dataset. Further, several action videos from similar subject (long videos) are present in this dataset similar to UCF-11.

The KTH dataset includes six action classes: walking, jogging, running, boxing, hand waving and hand clapping. Each action class includes 25 subjects which perform the actions several times in four different scenarios (indoors, outdoors, outdoors with scale variation, outdoors with different clothes). The database contains 600 video sequences and all of them are captured in homogeneous and static backgrounds.

For both the datasets, we randomly select 70% of the videos per category to represent the training set and the remaining 30% is used to evaluate the classification performance of the proposed framework. It is ensured that videos from similar subjects are not split over the training and test sets. This setup helps in assessing the robustness of the proposed encoding for diverse scenarios. We consider multiple train-test splits and report the average classification accuracy. Besides, we employ the one-against-all multi-class SVM setup and the SVM trade-off parameter is fixed by 5-fold cross-validation.

We compare the classification performance of the proposed feature encoding with three of the recent and standard feature pooling strategies: average pooling, max pooling and the time-series pooling of [5].

3.2. Discussion on results

Table 1 reports the comparative analysis of the proposed feature encoding with respect to the aforementioned pooling strategies. It can be observed that performance measures of the average and max pooling are moderate for both the datasets ($74,79\%$ and $76,87\%$ respectively) and max-pooling outperforms average pooling for sports videos. The performance of the time-series pooling technique of [5] is comparatively less for the UCF-50 data (73%) since many of the videos contain substantial background motion as well as spurious motion from other sources. On the contrary, the videos in the KTH dataset contain static backgrounds with no side objects present. Time-series pooling [5] is able to capture the action characteristics in this respect and provides enhanced recognition performance (88%).

Our feature encoding explicitly focuses on the entire video contents as well as the fine-grained information extracted from a set of snippets. Combination of both the features is capable of suppressing the effects of inconsequential side activities to a considerable extent and the classification performance with the proposed encoding is reported to be 78% for the UCF-50 data. It is found that the proposed feature encoding substantially outperforms the features of [5] for action classes like horse riding, kayaking, trampoline jumping etc. which have highly cluttered backgrounds. Similarly, the spectral clustering based temporal video segmentation eliminates the background effect for actions like running, jogging and walking in the KTH dataset and highlights the snippets with sophisticated action components. An enhancement of 1.3% can be observed in the recognition accuracies for those classes when both the global and snippet-level features are considered instead of only global video-level features. Further, a recognition performance of 90% is obtained for the KTH dataset with the proposed encoding which clearly outperforms the rests.

4. CONCLUSIONS

We introduce an efficient feature pooling based representation of videos for action classification. In contrast to the traditional average/max pooling based paradigms which works globally on the entire video, we propose a two-level video representation taking into account the global video representation as well as features extracted from a set of local snippets. In both the cases, we introduce efficient pooling techniques applied to the difference vectors of consecutive frames. A self-tuned spectral clustering is employed to temporally segment the video frames in order to obtain the snippets. As a result, the proposed encoding is capable of capturing both the global and localized characteristics of the action under consideration and sharply reduces the effects due to irrelevant contents. Experimental results obtained on the challenging UCF-50 and KTH datasets establish the robustness of the proposed encoding. Currently, we are engaged in ranking the video frames on the basis of their contributions towards recognizing the underlying action category.

5. REFERENCES

[1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[2] Sepp Hochreiter and Jürgen Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[3] Samitha Herath, Mehrtash Harandi, and Fatih Porikli, "Going deeper into action recognition: A survey," *arXiv preprint arXiv:1605.04988*, 2016.

[4] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2014, pp. 1725–1732.

[5] Michael S Ryoo, Brandon Rothrock, and Larry Matthies, "Pooled motion features for first-person videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 896–904.

[6] Lihi Zelnik-Manor and Pietro Perona, "Self-tuning spectral clustering," in *Advances in neural information processing systems*, 2004, pp. 1601–1608.

[7] Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari, "Measuring the objectness of image windows," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, no. 11, pp. 2189–2202, 2012.

[8] Kishore K Reddy and Mubarak Shah, "Recognizing 50 human action categories of web videos," *Machine Vision and Applications*, vol. 24, no. 5, pp. 971–981, 2013.

[9] Ivan Laptev, "On space-time interest points," *International Journal of Computer Vision*, vol. 64, no. 2-3, pp. 107–123, 2005.

[10] Hakan Bilen, Basura Fernando, Efstratios Gavves, Andrea Vedaldi, and Stephen Gould, "Dynamic image networks for action recognition," in *IEEE International Conference on Computer Vision and Pattern Recognition CVPR*, 2016.

[11] Basura Fernando, Efstratios Gavves, Jose M Oramas, Amir Ghodrati, and Tinne Tuytelaars, "Modeling video evolution for action recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5378–5387.

[12] Jonathan Harel, Christof Koch, and Pietro Perona, "Graph-based visual saliency," in *Advances in neural information processing systems*, 2006, pp. 545–552.

[13] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," in *British Machine Vision Conference*, 2014.