UC-HCC Submission to Thumos 2014

O. V. Ramana Murthy, Roland Goecke

Vision & Sensing, HCC, ESTeM, University of Canberra

Abstract. We basically use a Bag-of-Words framework. We compute the improved dense trajectories to compute Fisher vectors that serve as features. Using the training videos, we compute a mapping function which we conjecture to contain the principal information about each action. Given a temporally untrimmed video, we project it's feature along this mapping. The transformed features are passed to 1-vs all SVM classifiers framework to get the prediction score of each actions in the given video clip.

Keywords: Action recognition, dense trajectories, Fisher vector, PCA

1 Introduction

The overall layout of our proposed framework is shown in Fig. 1. Firstly, interest points improved trajectories of moving objects [10] are detected. Local descriptors are computed around these detected interest points. Gaussian Mixture Modelling is applied on each of these descriptors. Fisher vectors are generated from these Gaussian models. This feature vector is usually of very high order. We apply different dimensionality reduction techniques here. The reduced dimension feature vector is used to learn a classifier (for each action class detection). The details of each stage are discussed in following sections.

Fig. 1. Overall Framework

1.1 Dense Trajectories

Wang *et al.* [9] proposed dense trajectories to model human actions. Interest points were sampled at uniform intervals in space and time, and tracked based on displacement information from a dense optical flow field. Improved trajectories [10] are an improved version of the dense trajectories obtained by estimating the camera motion. Wang and Schmid [10] use a human body detector to separate motion stemming from humans movements from camera motion. The estimate

is also used to cancel out possible camera motion from the optical flow. For trajectories of moving objects, we compute these improved trajectories. In our experiments, we only use the online version [1] of camera motion compensated improved trajectories, without any human body detector.

The local descriptors computed on these trajectories are histograms of oriented gradients (HOG), histograms of optic flow (HOF), motion boundary histograms (MBH) and trajectory shape. While HOG captures the local motion and appearance, HOF captures the temporal changes. MBH are descriptors based on motion boundaries and are computed by separate derivatives for the horizontal and vertical components of the optical flow. The trajectory shape descriptor encodes local motion patterns.

1.2 Feature Encoding

We build Fisher Vectors for each descriptor separately. Other encoding techniques like the hard assignment capture only the information of frequency of the visual words (of the codebook) for given video. Fisher vectors capture the first order (deviations from the visual words) and second order (covariance deviation) statistics. Further, in an recent study on large scale image classification [6], fisher vectors have been found to perform best. So, we use Fisher vector encoding for constructing the features from the local descriptors. Firstly, 10,000 descriptors from each class of the training data are randomly selected. Accumulating these from the 101 classes of the Training Set [3] roughly results in 10^6 descriptors. Next, Principal Component Analysis (PCA) is applied to reduce the descriptor dimensionality by a factor of two. Then, a Gaussian Mixture Model (GMM) is fitted to this data with the number of Gaussians $K = 256$. Each video is, then, represented by a $2 \times D \times K$-dimensional Fisher vector for each descriptor type, where D is the descriptor dimension after performing PCA. Finally, power normalisation and L_2 normalisation are applied to the Fisher vector as set forth in [6].

1.3 Feature Projection

We then apply Kernel Principal Component Analysis (KPCA) to project the feature vector into lower dimensional . Kernel PCA (KPCA) is the reformulation of traditional linear PCA [2, 4, 5] in a high-dimensional space that is constructed using a kernel function [7] i.e., reformulation of linear techniques using the 'kernel trick'. Kernel PCA computes the principal eigenvectors of the kernel matrix, instead of those of the covariance matrix. A kernel matrix is similar to the inner product of the data points in the high-dimensional space that is constructed using the kernel function. The application of PCA in the kernel space provides Kernel PCA the property of constructing non-linear mappings.

For classification we use these reduced feature vectors and linear SVM [1]. We apply the one-versus-all approach in all the cases and select the class with the highest score.

[1] *http://lear.inrialpes.fr/people/wang/improved_trajectories*

1.4 Datasets

THUMOS 2014 dataset [3] is an extension of **UCF101** dataset. The training Set contains over 13,000 temporally trimmed videos from 101 action classes described in **UCF101** [8]dataset. These videos contain one action only. Further there are Validation Set containing 1010 temporally untrimmed videos and a test Set containing 1574 temporally untrimmed videos. In these two Sets- Validation and Test, there is one primary action class shown in each video; however, some videos may include one or more instances from other action classes. Additionally a background Set containing over 2500 relevant videos guaranteed to not include any instance of the 101 actions is also provided. All videos were collected from YouTube. In our experiments we trained our models using Training Set only. The results are reported on Validation and Test Sets.

2 Results and Discussions

The results obtained by applying KPCA on untrimmed videos are shown in Table. 1. It can be observed that KPCA reduced FV performed better than direct FV by 0.05%(absolute). We conjecture that apart from reducing the dimensions, KPCA also computed Principal components dimensions for each of the 101 actions. Recall, that we used training Set only to create the KPCA components. When FV computed on untrimmed videos is projected along these KPCA components, discriminative power is observed better.

Table 1. Performance (mAP) on **Thumos 2014** dataset

Technique	Validation Set	Test Set
Direct Fisher Vector (FV)	0.5821	NA
KPCA reduced FV	**0.6335**	0.5161

3 Conclusions

Improved trajectories have been found to perform the best on most of the existing human action recognition datasets. However, they are large in dimension and unexplored on untrimmed videos. To find a combined solution, Kernel PCA was investigated thoroughly. It has been found out that better performance than original Fisher vector can be achieved.

References

1. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: LIBLINEAR: A library for large linear classification. Journal of Machine Learning Research 9, 1871–1874 (2008)
2. Hotelling, H.: Analysis of a complex of statistical variables into principal components. J. Educ. Psych. 24 (1933)
3. Jiang, Y.G., Liu, J., Roshan Zamir, A., Toderici, G., Laptev, I., Shah, M., Sukthankar, R.: THUMOS challenge: Action recognition with a large number of classes. http://crcv.ucf.edu/THUMOS14/ (2014)
4. van der Maaten, L., Postma, E.O., van den Herik, H.J.: Dimensionality reduction: A comparative review (2009)
5. Pearson, K.: On lines and planes of closest fit to systems of points in space. Philosophical Magazine 2(6), 559–572 (1901)
6. Perronnin, F., Sánchez, J., Mensink, T.: Improving the Fisher kernel for large-scale image classification. In: European Conference on Computer Vision (ECCV) (2010)
7. Schölkopf, B., Smola, A., Müller, K.R.: Nonlinear component analysis as a kernel eigenvalue problem. Neural Comput. 10(5), 1299–1319 (Jul 1998)
8. Soomro, K., Zamir, A.R., Shah, M.: UCF101: A Dataset of 101 Human Action Classes from Videos in the Wild. In: CRCV-TR-12-01 (November 2012)
9. Wang, H., Kläser, A., Schmid, C., Liu, C.L.: Action recognition by dense trajectories. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2011)
10. Wang, H., Schmid, C.: Action Recognition with Improved Trajectories. In: ntenational Conference on Computer Vision (ICCV) (2013)