Trajectory-based Fisher Kernel Representation for Action Recognition in Videos

Indriyati Atmosukarto
Advanced Digital Sciences Centre (ADSC), Singapore
indria@adsc.com.sg

Bernard Ghanem
King Abdullah University of Science and Technology
bernard.ghanem@kaust.edu.sa

Narendra Ahuja
University of Illinois at Urbana–Champaign
n-ahuja@illinois.edu

Abstract

Action recognition is an important computer vision problem that has many applications including video indexing and retrieval, event detection, and video summarization. In this paper, we propose to apply the Fisher kernel paradigm to action recognition. The Fisher kernel framework combines the strengths of generative and discriminative models. In this approach, given the trajectories extracted from a video and a generative Gaussian Mixture Model (GMM), we use the Fisher Kernel method to describe how much the GMM parameters are modified to best fit the video trajectories. We experiment in using the Fisher Kernel vector to create the video representation and to train an SVM classifier. We further extend our framework to select the most discriminative trajectories using a novel MIL-KNN framework. We compare the performance of our approach to the current state-of-the-art bag-of-features (BOF) approach on two benchmark datasets. Experimental results show that our proposed approach outperforms the state-of-the-art method [8] and that the selected discriminative trajectories are descriptive of the action class.

1 Introduction

Human action recognition and classification is a very challenging computer vision problem due to subject differences, cluttered background, occlusion, or appearance variation. The same action performed by two different persons may look completely different. One of the popular approaches to action classification in videos is to represent the videos using bag-of-feature (BOF) histograms and to classify the videos using non-linear Support Vector Machines (SVM). This approach starts by extracting descriptors from the videos and clustering them to form visual word clusters that define the vocabulary. A video is then represented as a histogram of visual word occurrences.

In this paper, we go beyond the traditional BOF framework and propose to represent videos using fisher kernel vectors. The Fisher kernel paradigm has been shown excellent performance in image classification [5], detecting protein homologies [4], speech recognition, handwriting recognition [6], etc. However, to the best of our knowledge, it has yet to be applied to the field of action recognition in videos. The Fisher kernel framework combines the advantages of generative and discriminative models. The BOF approach suffers from high dimensionality of the feature vectors thus leading to high computation needed in forming the vocabulary. The Fisher kernel framework solves this problem by representing the vocabulary using Gaussian Mixture Models (GMM) where each Gaussian represents a word in the vocabulary. However, unlike the BOF approach that merely assigns descriptors to the nearest visual word, the Fisher kernel approach retains information about the fitting error of the best visual word by computing the gradient vector of the descriptors. Due to the additional fitting error information, a smaller number of Gaussians, i.e., smaller vocabulary size is usually sufficient to represent the videos. Another advantage of using the Fisher kernel framework is that it transforms data that are of variable size into a fixed length vector where the size of the vector is dependent on the number of parameters in the model. In contrast to most existing action recognition methods that use interest points to extract features from videos, we extract dense motion trajectories [8] from videos and use these trajectories to train the GMM and extract the Fisher kernel features.

Extracting dense trajectories from video creates an ambiguity in deciding which trajectories are important in describing the action present in the video. To overcome this ambiguity, we use a multi-instance learning (MIL) based approach. Instead of using singleton trajectory samples, MIL analyzes groups of trajectories describing a single video (i.e. video bags). A video bag is labeled as positive if at least one of the trajectory instances is positive, and negative if all instances are negative. MIL learns which instances (trajectories)

in the positive bags are positive (action trajectories). Instances with high confidence scores are selected to be discriminative trajectories that best represent the video. To counteract noise in the training samples, Berg et al. [9] proposed a hybrid of the nearest neighbor classification method and support vector machines (SVM-KNN) for the task of multi-class image classification. Inspired by this work, we extend the method to combine nearest neighbor classification methods and multiple instance learning (MIL). We propose to train an MIL based SVM using the collection of nearest neighbors (MIL-KNN), where every video is represented as a bag of trajectories. Using this framework, we select the discriminative trajectories in a video sample.

2 Video Representation

Our framework is general and can be applied to any video descriptor including interest point based descriptors. For our purposes, we adopt trajectory-based representation as we believe trajectories contain more discriminative information on motion and action in videos.

2.1 Trajectory-based representation

Trajectories are efficient in capturing object motions and actions in videos. In our experiments, we use dense trajectories that are obtained by tracking densely sampled points using optical flow fields [8]. Feature points are sampled along a dense grid and are tracked for a fixed number of frames. Trajectories are formed by concatenating tracked feature points in frame sequences.

The shape of the trajectories, which essentially describe the motion in the videos, is described by the sequence of their displacement vectors. The sequence is further normalized by the sum of the magnitudes of the displacement vectors. In addition to the shape of the trajectory, the trajectories are further described using descriptors computed within the space-time volume around each trajectories [8]. The descriptors include HOG (histogram of oriented gradients) [2], which describes the appearance in the volume, HOF (Histogram of optical flow), which describes the local motion information within the trajectory volume, and MBH (motion boundary histogram) [3], which describes the relative motion between pixels. The final trajectory descriptor is a concatenation of the shape, HOG, HOG, and MBH descriptor vectors.

2.2 Fisher Kernel Framework for Trajectories

Given a training set of videos belonging to multiple classes of actions, we aim to learn a classifier that accurately discriminates between the action classes. Each video is composed of a different number of trajectory descriptors. In this sense, each video can be viewed as a bag of trajectories. In this paper, we propose to apply the Fisher kernel paradigm [4] to represents bags of trajectories in the training set. To do this, we require a *universal*, parametric probability distribution that describes the trajectories in the training set. We choose this distribution to be a (generative) Gaussian Mixture Model (GMM) on the trajectory descriptors. From this GMM, Fisher kernel features can be extracted for each video and a (discriminative) Support Vector Machine (SVM) can be learned in a supervised manner.

We denote video $X = \{\mathbf{x}_t, t = 1 \dots T\}$, where $\mathbf{x}_t$ is a D-dimension trajectory descriptor vector and T the number of trajectories in the video. Let λ be the set GMM parameters such that $\lambda = \{w_i, \mu_i, \Sigma_i, \forall i = 1 \dots N\}$, where N represents the number of Gaussian components and w_i, μ_i, and $\sum_i$ denote the weight, mean vector, and covariance matrix of the i^{th} Gaussian in the mixture respectively.

In the Fisher kernel framework, X is described by the gradient vector $\nabla_\lambda \log p(X|\lambda)$. The basic notion behind the Fisher kernel principle is that the gradient vector of the log-likelihood describes the fitting error of the model parameters in describing the data and provides information on the direction in which the model (GMM) parameters need to be modified to best fit the data. We define $\mathcal{L}(X|\lambda)$ as the log-likelihood of X with respect to the GMM with parameter λ where $\mathcal{L}(X|\lambda) = \log p(X|\lambda) = \sum_{t=1}^{T} \log p(\mathbf{x}_t|\lambda)$. Here, the likelihood that $\mathbf{x}_t$ is generated by the GMM is $p(\mathbf{x}_t|\lambda) = \sum_{t=1}^{N} w_i p_i(\mathbf{x}_t|\lambda)$, where

$$p_i(\mathbf{x}|\lambda) = \frac{\exp\left(-\frac{1}{2}(\mathbf{x}-\mu_i)^T \Sigma_i^{-1}(\mathbf{x}-\mu_i)\right)}{(2\pi)^{D/2}|\sum_i|^{1/2}}. \quad (1)$$

The probability that $\mathbf{x}_t$ is generated by Gaussian component i is denoted as $\gamma(i)$ where

$$\gamma(i) = \frac{w_i p_i(\mathbf{x}_t|\lambda)}{\sum_{j=1}^{N} w_j p_j(\mathbf{x}_t|\lambda)}. \quad (2)$$

In Eq. (3), we calculate the elements of the Fisher kernel vector of X, where $\mathbf{a}^d$ denotes the d^{th} component of vector $\mathbf{a}$. For simplicity, we take the covariance matrices of the GMM to be diagonal.

$$\begin{cases} \frac{\partial \mathcal{L}(X|\lambda)}{\partial w_i} = \sum_{t=1}^{T} \left[\frac{\gamma_t(i)}{w_i} - \frac{\gamma_t(1)}{w_1}\right] \\ \frac{\partial \mathcal{L}(X|\lambda)}{\partial \mu_i^d} = \sum_{t=1}^{T} \gamma_t(i) \left[\frac{\mathbf{x}_t^d - \mu_i^d}{(\sigma_i^d)^2}\right] \\ \frac{\partial \mathcal{L}(X|\lambda)}{\partial \sigma_i^d} = \sum_{t=1}^{T} \gamma_t(i) \left[\frac{(\mathbf{x}_t^d - \mu_i^d)^2}{(\sigma_i^d)^3} - \frac{1}{\sigma_i^d}\right] \end{cases} \quad (3)$$

The final Fisher kernel vector of X is the concatenation of the partial derivatives with respect to the mean and standard deviation parameters. Given that D is the dimension of the trajectory descriptor vectors and N the number of Gaussian components, the size of the gradient vector is thus $2DN$. In our implementation,

the values of D and N are in the order of 100, hence the dimensionality of the gradient vector is in the order of $10,000$. This high-dimensional vector contains discriminative information used to train an SVM classifier.

Next, we give a summary of our Fisher Kernel classification method, denoted as FK-SVM. Given a training set, we extract dense trajectories from the training videos. We describe trajectories in each video using the descriptors mentioned above. We further reduce the dimensionality of the feature vectors using PCA. These features are then used to train a universal GMM. Then, we compute the gradient vector of every training video with respect to the GMM parameters. These Fisher kernel vectors are then used to train a non-linear SVM classifier. Given a new test sample, we compute the Fisher kernel vector of the test sample with respect to the trained GMM model. This vector is then classified by the learned classifier.

2.3 Discriminative Trajectories

We extend our FK-SVM framework to perform trajectory selection and find the most discriminative trajectories in a video sample. To do this, we train a Multiple Instance Learning (MIL) based Support Vector Machine using only the nearest neighbors of a video sample. We denote this approach as FK-MIL-KNN. Given a new test video, we compute its Fisher kernel vector, which is used to find its K-nearest neighbors in the training videos. Only these K training samples are used to classify the test sample. If the K nearest neighbors of the test sample have all the same action label, the test sample is given the same label. Otherwise, we train a binary MIL for every action present in the K training samples. Every video is a bag and the trajectories extracted from the video are instances of the bag. The bags are labeled positive or negative depending on the action model being trained. The given test sample is labeled according to the action model with the highest bag accuracy. The MIL approach allows us to find the discriminative instances (trajectories) in a bag (video). We select those trajectory instances that are labeled as positive and have high instance confidence score. We observe that these discriminative trajectories hold semantic meaning for each action class. In fact, these learned trajectories can be used to build more sophisticated discriminative action models; however, we leave this for future work.

3 Experimental Results

In this section, we validate the performance of our proposed framework by applying it to two action recognition benchmark datasets: Weizmann [1] and KTH [7]. The first dataset contain actions performed in front of a uniform background using static camera, while the second dataset is more challenging due to some change in scale and background. The **Weizmann** dataset [1] contains 90 video clips showing nine different people performing ten actions including running, walking, skipping, jumping-jack, jumping-forward-on-two-legs, jumping-in-place-on-two-legs, galloping- sideways, waving-two-hands, waving-one-hand and bending. The **KTH** dataset [7] contains six types of human actions: boxing, handclapping, handwaving, jogging, running, and walking. Each human action is performed several times by 25 subjects in four different scenarios: outdoors, outdoors with scale variation, outdoors with different clothes, and indoors.

To extract the trajectories from the video samples, we used the original dense trajectories source code [8]. We experimented with different trajectory lengths and sampling set size. Short trajectory length is preferrable as short trajectories are better at representing repetitive movements such as boxing or handwaving. We report on experiments performed using trajectory length to $L = 15$ and dense sampling step size $W = 10$. We used the default parameters to compute the descriptors (HOG, HOF, and MBH), the parameters were set as $N = 32$, $n_\sigma = 2$, $n_\tau = 3$. We randomly sampled 100 trajectories per video clip.

We systematically evaluated our trajectory representation with a set of experiments. We first tested our FK-SVM method on the Weizmann dataset. The Weizmann dataset was filmed in a controlled environment. Using a leave-one-out setup, our FK-SVM method achieved 100% classification accuracy on this dataset.

Next, we compared the performance of our FK-SVM method to the BOF method [8]. Wang et al. [8] extracted dense trajectories from every video sample and used a standard bag-of-features approach to construct the video descriptors. They randomly selected a subset of the training features where for each feature type (trajectory shape, HOG, HOF, MBH), they constructed a codebook. The number of visual words per descriptor was set to 4000. The histograms of visual word occurences are used as video descriptors. The descriptors were then used to train a non-linear SVM classifier. In fact, the BOF approach is a special case of the Fisher Kernel approach where only the gradient with respect to the weight parameters are considered. In our FK-SVM method, we take the derivatives with respect to the means and standard deviations, thus expanding the BOF descriptor to a higher dimensional vector. For our experiments, we used 100 Gaussian components when training our GMM. The classification accuracy for the two datasets are presented in Table 3. The KTH dataset is more challenging as the videos were taken at varied scales and backgrounds. The results show that our method is comparable or in some cases better than the BOF method. However, our approach reduces training

	Weizmann	KTH
KLT-BOF-SVM [8]	-	93.4
DT-BOF-SVM [8]	-	94.2
KLT-FK-SVM	95.3	91.20
DT-FK-SVM	100	95.37

Table 1. Classification accuracy of the BOF (DT-BOF-SVM and KLT-BOF-SVM) and proposed Fisher Kernel (DT-FK-SVM and KLT-FK-SVM) approaches on KLT and dense (DT) trajectories

computational time. Even though the videos are characterized with high-dimensional vectors, small number of Gaussian components are needed during training time. The results also show that using dense trajectories to construct the descriptor achieves better classification accuracy than using KLT trajectories. This result agrees with the results in [8] confirming that trajectories obtained by the KLT tracker are often too sparse.

Next, we extended the FK-framework to perform trajectory selection. We classified the test video samples using the FK-MIL-KNN framework. Each sampled trajectory is treated as an instance and all instances from a video sample form a bag. As mentioned in the previous section, all instances in a non-action bag are negative samples, while only some instances in the action bags are positive. From the positive samples, we can identify which are the interesting and discriminative trajectories. We select the top $K = 5$ trajectories from the FK-KNN-MIL classification results. The trajectories are sorted based on the classifier's confidence score for a particular instance in a bag. Figure 1 and 2 show examples of the top 5 trajectories for videos in the Weizmann and KTH datasets respectively. The results show that the discriminative trajectories accurately describe the motion in the action class.

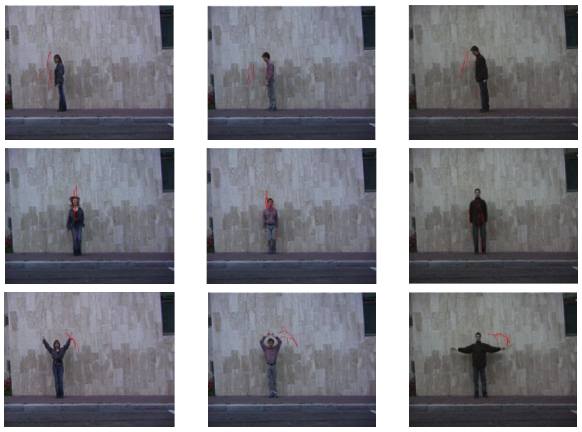

Figure 1. Discriminative trajectories selected using the FK-KNN-MIL framework (overlaid red trajectories) from sample frames in the Weizmann dataset. Top to bottom row: bend, pjump, and handwaving.

Figure 2. Discriminative trajectories selected using the FK-KNN-MIL framework (overlaid red trajectories) from sample frames in the KTH dataset. Top to bottom row: boxing, handclapping, and handwaving.

4 Conclusion

In this paper, we propose to use Fisher Kernel to represent trajectory motion in video. We further extend our framework by selecting the most discriminative trajectories to describe the actions in the video. Experimental result on action recognition datasets show that our method is comparable to existing BOF methods. The selected discriminative trajectories can be used to assist as instance prototypes in multiple instance learning frameworks and to reduce noise in training data when classifying action videos.

Acknowledgments: This study is supported by the research grant for the Human Sixth Sense Program at ADSC from Singapore's Agency for Science, Technology and Research (A*STAR).

References

[1] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. In *ICCV*, pages 1395–1402, 2005.

[2] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[3] N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. In *ECCV*, 2006.

[4] T. Jaakola and D. Haussler. Exploiting generative models in discriminative classifiers. In *NIPS*, 1999.

[5] F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *CVPR*, 2007.

[6] F. Perronnin and J. A. Rodriguez-Serrano. Fisher kernels for handwritten word-spotting. In *International Conference on Document Analysis and Recognition*, 2009.

[7] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: A local svm approach. In *ICPR*, 2004.

[8] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu. Action recognition by dense trajectories. In *CVPR*, pages 3169–3176, 2011.

[9] H. Zhang, A. C. Berg, M. Maire, and J. Malik. Svm-knn: Discriminative nearest neighbor classification for visual category recognition. In *CVPR*, 2006.