Sparse Coding on Local Spatial-Temporal Volumes for Human Action Recognition

Yan Zhu[1], Xu Zhao[1], Yun Fu[2], and Yuncai Liu[1]

[1] Shanghai Jiao Tong University, Shanghai 200240, China
[2] Department of CSE, University at Buffalo (SUNY), NY 14260, USA

Abstract. By extracting local spatial-temporal features from videos, many recently proposed approaches for action recognition achieve promising performance. The Bag-of-Words (BoW) model is commonly used in the approaches to obtain the video level representations. However, BoW model roughly assigns each feature vector to its closest visual word, therefore inevitably causing nontrivial quantization errors and impairing further improvements on classification rates. To obtain a more accurate and discriminative representation, in this paper, we propose an approach for action recognition by encoding local 3D spatial-temporal gradient features within the sparse coding framework. In so doing, each local spatial-temporal feature is transformed to a linear combination of a few "atoms" in a trained dictionary. In addition, we also investigate the construction of the dictionary under the guidance of transfer learning. We collect a large set of diverse video clips of sport games and movies, from which a set of universal atoms composed of the dictionary are learned by an online learning strategy. We test our approach on KTH dataset and UCF sports dataset. Experimental results demonstrate that our approach outperforms the state-of-art techniques on KTH dataset and achieves the comparable performance on UCF sports dataset.

1 Introduction

Recognizing human actions is of great importance in various applications such as human-computer interaction, intelligent surveillance and automatic video annotation. However, the diverse inner-class variations of human poses, occlusions, viewpoints and other exterior environments in realistic scenarios, make it still a challenging problem for accurate action classification.

Recently, approaches based on local spatial-temporal descriptors [1–3] have achieved promising performance. These approaches generally first detect or densely sample a set of spatial-temporal interest points from videos; then describe their spatial-temporal properties or local statistic characteristics within small cuboids centered at these interest points. To obtain global representations from sets of local features, the popular bag-of-words model [1, 4–6] is widely used incorporating with various local spatial-temporal descriptors. In the BoW model, an input video is viewed as an unordered collection of spatial-temporal words, each of which is quantized to their closest visual word existing in a trained

R. Kimmel, R. Klette, and A. Sugimoto (Eds.): ACCV 2010, Part II, LNCS 6493, pp. 660–671, 2011.

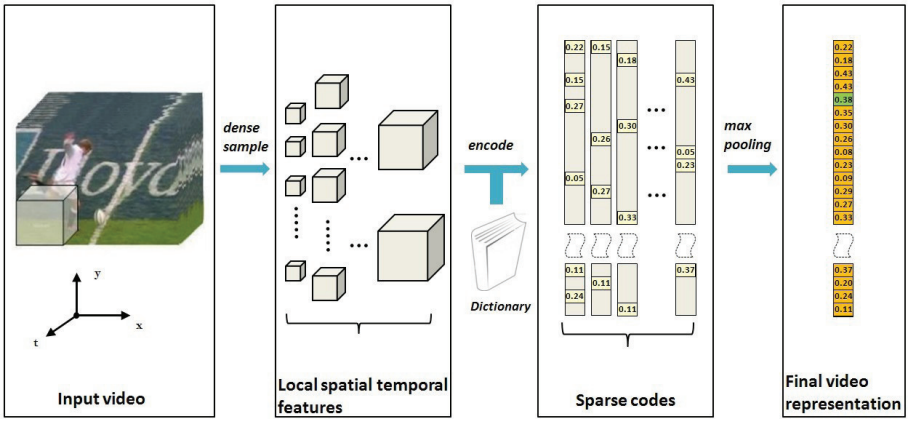

Fig. 1. Framework of our action recognition system. First the input video is transformed to a group of local spatial temporal features through dense sampling. Then the local features are encoded into sparse codes using the pre-trained dictionary. Finally, max pooling operation is applied over the whole sparse code set to obtain the final video representation.

dictionary by measuring a distance metric within the feature space. The video is finally represented as the histogram of the visual words occurrences. However, there is a drawback in its quantization strategy of the BoW model. Simply assigning each feature to its closest visual word can lead to relatively high reconstruction error; therefore the obtained approximation of the original feature is too coarse to be sufficiently discriminative for the following classification task.

To address this problem, we propose an action recognition approach within sparse coding framework, which is illustrated in Fig. 1. Firstly we densely extract a set of local spatial-temporal descriptors with varying space and time scales from the input video. The descriptor we adopt is HOG3D [2]. We use sparse coding to encode each local descriptor into its corresponding sparse code according to the pre-trained dictionary. Then maximum pooling procedure is operated over the whole sparse code set of the video. The obtained feature vector, namely the final representation of the video, is the input of the classifier for action recognition. In the classification stage, we use multi-class linear support vector machine.

Another issue addressed in this paper, is how to effectively model the distribution of the local spatial-temporal volumes over the feature space. To this end, constructing a well defined dictionary is a critical procedure in sparse coding framework. Generally a dictionary is built from a collection of training video clips with known labels similar to the test videos. However, a large set of labeled video data are required to get a well generalized dictionary. But it is a difficult task to conduct video annotation on such a large video database; and the generalization capability of the dictionary is degraded using homologous training and test data. In this work, motivated by previous works introducing transfer

learning into image classification [7, 8], we construct our dictionary by utilizing large volumes of unlabeled video clips, from which the dictionary can learn universal prototypes to facilitate the classification task. These video clips are widely collected from movies and sport games. Although these unlabeled video sequences may not be closely relevant with the actions to be classified on the semantic level, they can be used to learn a more generalized latent structure of the feature space. Its efficacy is demonstrated by extensive comparative experiments.

The remaining of the paper is organized as follows. Section 2 reviews the previous related works in action recognition and sparse coding. Section 3 introduces our proposed action recognition approach and the implementation details. Section 4 describes the experimental setup, results, and discussion. Section 5 briefly concludes this paper and addresses the future work.

2 Related Work

Many existing approaches of action recognition extract video representations from a set of detected interest points [9–11]. Diverse local spatial-temporal descriptors have been introduced to describe the properties of these points. Although compact and computational efficient, interest point based approaches mitigate much potentially useful information of the original data, and thus weaken the discriminative power of the representation. Recently two evaluations confirm that dense sampling method could achieve better performance in action recognition tasks [12, 13]. In our work, we adopt dense sampling method, but we use sparse coding instead of BoW model to obtain a novel representation of videos.

Sparse representation has been widely discussed recently and achieved exciting progress in various fields including audio classification [14], image inpainting [15], segmentation [16] and object recognition [8, 17, 18]. These successes demonstrate that sparse representation could flexibly adapt to diverse low level natural signals with desirable properties. Besides, research in visual cortex has justified the biological plausibility of sparse coding [19]. For tasks of human action classification, interest information, inherently, is sparsely distributed in the spatial-temporal domain. Inspired by above insights, we introduce sparse representations into the field of action recognition from video.

To improve the classification performance, transfer learning is incorporated into sparse coding framework to obtain robust representations and learn knowledge from unlabeled data [8, 17, 20]. Raina et al. [8] proposed self-taught learning to construct the sparse coding dictionary using unlabeled irrelevant data. Yang et al. [17] integrated sparse coding with traditional spatial pyramid matching method for object classification. Liu et al. [20] proposed a topographic subspace model for image classification and retrieval employing inductive transfer learning. Motivated by above successes, our work explores the application of sparse coding with transfer learning in video domain. Experimental results will validate that dictionary training with transferable knowledge from unlabeled data can significantly improve the performance of action recognition.

3 Approach

3.1 Local Descriptor and Sampling Strategy

Each video can be viewed as a 3D spatial-temporal volume. In order to capture sufficient discriminative information, we choose to densely sample a set of local features across space-time domain throughout the input video volume. Each descriptor is computed within a small 3D cuboid centered at a space-time point. Multiple sizes of 3D cuboid are adopted to increase scale and speed invariance. We follow the dense sampling parameter settings as described in [13]. To capture local motion and appearance characteristic, we use HOG3D descriptor proposed in [2]. The descriptor computation can be summarized as following steps:

1. Smooth the input video volume to obtain the integral video representation.
2. Divide the sampled 3D cuboid centered at $\mathbf{p} = (x_p, y_p, t_p)^\top$ into $n_x \times n_y \times n_t$ cells and further divide each cell into $S_b \times S_b \times S_b$ subblocks.
3. For each subblock $\mathbf{b}_i$, compute the mean gradient $\bar{\mathbf{g}}_{\mathbf{b}_i} = (\bar{g}_{\mathbf{b}_i \partial x}, \bar{g}_{\mathbf{b}_i \partial y}, \bar{g}_{\mathbf{b}_i \partial t})^\top$ using the integral video.
4. Quantize each mean gradient $\bar{\mathbf{g}}_{\mathbf{b}_i}$ as $\mathbf{q}_{\mathbf{b}_i}$ using a regular icosahedron.
5. For each cell $\mathbf{c}_j$, compute the histogram $\mathbf{h}_{\mathbf{c}_j}$ of the quantized mean gradients over $S_b \times S_b \times S_b$ subblocks.
6. Concatenate the $n_x \times n_y \times n_t$ histograms to one feature vector. After ℓ_2 normalization, the obtained vector $\mathbf{x}_\mathbf{p}$ is the HOG3D descriptor for $\mathbf{p}$.

In Section 4, parameter settings for sampling and descriptor computation will be discussed in detail.

3.2 Sparse Coding Scheme for Action Recognition

In our action recognition framework, sparse coding is used to obtain a more discriminative intermediate representation for human actions. Suppose we have obtained a set of local spatial-temporal features $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_N] \in \mathbb{R}^{d \times N}$ to represent a video, where each feature is a d-dimensional column vector; and we have a well-trained dictionary $\mathbf{D} = [\mathbf{d}_1, ..., \mathbf{d}_S] \in \mathbb{R}^{d \times S}$. Sparse coding method [8, 17, 18] manages to sparsely encode each feature vector in $\mathbf{X}$ into a linear combination of a few atoms of dictionary $\mathbf{D}$ by optimizing

$$\widehat{\mathbf{Z}} = \underset{\mathbf{Z} \in \mathbb{R}^{S \times N}}{\arg\min} \ \frac{1}{2} \|\mathbf{X} - \mathbf{D}\mathbf{Z}\|_{\ell_2}^2 + \lambda \|\mathbf{Z}\|_{\ell_1}, \tag{1}$$

where λ is a regularization parameter which determines the sparsity of the representation of each local spatial-temporal feature. Dictionary $\mathbf{D}$ is pre-trained to be an overcomplete basis set which is composed of S atoms, in which each atom is a d-dimensional column vector. Note that S typically is greater than $2d$. To avoid numerical instability, each column of $\mathbf{D}$ subjects to the constraint of $\|\mathbf{d}_k\|_{\ell_2} \leq 1$. Once the dictionary $\mathbf{D}$ is fixed, the optimization over $\mathbf{Z}$ alone is convex, thus can be viewed as an ℓ_1 regularized linear least square problem. The twofold optimization goal ensures the least reconstruction error and the

sparsity of the coefficients set $\widehat{\mathbf{Z}}$ simultaneously. We use the LARS-lasso implementation provided by [15] to find the optimal solution. After optimization, we get a set of sparse codes $\widehat{\mathbf{Z}} = [\widehat{\mathbf{z}}_1, ..., \widehat{\mathbf{z}}_N]$, where each column vector $\widehat{\mathbf{z}}_i$ has only a few nonzero elements. It can also be interpreted as that each descriptor only responds to a small subset of dictionary $\mathbf{D}$.

To capture the global statistics of the whole video, we use a maximum pooling function [17, 18] defined as

$$\boldsymbol{\beta} = \xi_{\max}(\widehat{\mathbf{Z}}), \tag{2}$$

to pool over the sparse code set $\widehat{\mathbf{Z}}$, where $\xi_{\max}$ returns a vector $\boldsymbol{\beta} \in \mathbb{R}^S$ with the k-th element defined as

$$\beta_k = \max\{|\widehat{Z}_{k1}|, |\widehat{Z}_{k2}|, ..., |\widehat{Z}_{kN}|\} \ . \tag{3}$$

Through maximum pooling, the obtained vector $\boldsymbol{\beta}$ is viewed as the final video level feature. Maximum pooling operation has been successfully used in several image classification frameworks to increase spatial translation invariance [17, 18, 21]. Such mechanism has been proven to be consistent with the properties of the cells in visual cortex [21]. Motivated by above insights, we adopt the similar procedure to increase both spatial and temporal translation invariance. Within the sparse code set, only the strongest response for each particular atom is preserved without considering its spatial and temporal location. Experimental results demonstrate that the maximum pooling procedure can lead to compact and discriminative final representation of the videos. Note that our previous choice of dense sampling strategy provides sufficient low-level local features, which guarantee the maximum pooling procedure statistically reliable.

3.3 Dictionary Construction Based on Transfer Learning

Under sparse coding framework, constructing a dictionary for a specific classification task is essentially to learn a set of overcomplete bases to represent the basic pattern of the specific data distribution within feature space. Given a large collection of local descriptors $\mathbf{Y} = [\mathbf{y}_1, ..., \mathbf{y}_M]$, the dictionary learning process can be interpreted as jointly optimizing with respect to the dictionary $\mathbf{D}$ and coefficients set $\mathbf{Z} = [\mathbf{z}_1, ..., \mathbf{z}_M]$,

$$\underset{\mathbf{Z}\in\mathbb{R}^{S\times M}, \mathbf{D}\in\mathcal{C}}{\arg\min} \ \frac{1}{M}\sum_{i=1}^{M}\frac{1}{2}\|\mathbf{y}_i - \mathbf{D}\mathbf{z}_i\|_{\ell_2}^2 + \lambda\|\mathbf{z}_i\|_{\ell_1}, \tag{4}$$

where $\mathcal{C}$ is defined as a convex set $\mathcal{C} \triangleq \{\mathbf{D} \in \mathbb{R}^{d\times S} \text{ s.t. } \|\mathbf{d}_k\|_{\ell_2} \leq 1, \forall k \in \{1, ..., S\}\}$. In dictionary training stage, the optimization is not convex when both dictionary $\mathbf{D}$ and coefficients set $\mathbf{Z}$ are varying. How to solve the above optimization, especially in situations of large training set, is still an open problem. Recently, Mairal et al. [15] presented an online dictionary learning algorithm, which is much faster than previous methods and proven to be more suitable for large training sets with action recognition purpose. Due to the desirable properties mentioned above, we use this technique to train our dictionary.

To discover the latent structure of the feature space for actions, we use a large set of unlabeled video clips collected from movies and sport games as the "learning material", which is different from [22]. In recent years, transfer learning from unlabeled data to facilitate the supervised classification task has sparked many successful applications in machine learning [8, 20]. Although these unlabeled video clips are not necessarily belong to the same classes with the test data, they are similar in that they all contain human motions sharing the universal patterns. This transferable knowledge is helpful for our supervised action classification.

In our experiment, we will construct two dictionaries in different ways. The first one is trained with patches from target classification videos and the second one is trained with unlabeled video clips. Experimental results will show that the second dictionary yields higher recognition rate. This can be explained that the unlabeled data contain more diverse patterns of human actions, which are helpful for the dictionary to thoroughly discover the nature of human action. In contrast, dictionary constructed merely from training data can hardly grasp the universal basis because of the insufficient information provided by training data, which made the dictionary unable to encode the test data sparsely and accurately. It is interesting to observe that the dictionary constructed from unlabeled data can potentially be utilized in other relevant action classification tasks. The transferable knowledge makes the dictionary more universal and reusable.

3.4 Multi-Class Linear SVM

In classification stage, we use multi-class support vector machine with linear kernels as described in [17]. Given a training video set $\{(\boldsymbol{\beta}_i, y_i)\}_{i=1}^n$ for an L-class classification task, where $\boldsymbol{\beta}_i$ denotes the feature vector of the i-th video and $y_i \in \mathcal{Y} = \{1, ..., L\}$ denotes the class label of $\boldsymbol{\beta}_i$, we adopt one-against-all method to train L linear SVMs, each of which seeks to learn L linear functions $\{\mathbf{W}_c^\top \boldsymbol{\beta} | c \in \mathcal{Y}\}$. The trained SVMs predict the class label y_j for a test video feature $\boldsymbol{\beta}_j$ by solving

$$y_j = \arg\max_{c \in \mathcal{Y}} \mathbf{W}_c^\top \boldsymbol{\beta}_j. \tag{5}$$

Note that the traditional histogram based action recognition models usually need some specific designed nonlinear kernels, which lead to time-consuming computations for classifier training. However, the linear kernel SVM operated on sparse coding statistics can achieve satisfying accuracy with much faster speed.

4 Experiments

We evaluate our approach on two benchmark human action datasets: the KTH action dataset [5] and the UCF Sports dataset [23]. Some sample frames from the two databases are shown in Fig. 2. In addition, we evaluate the different effects of two dictionary construction manners.

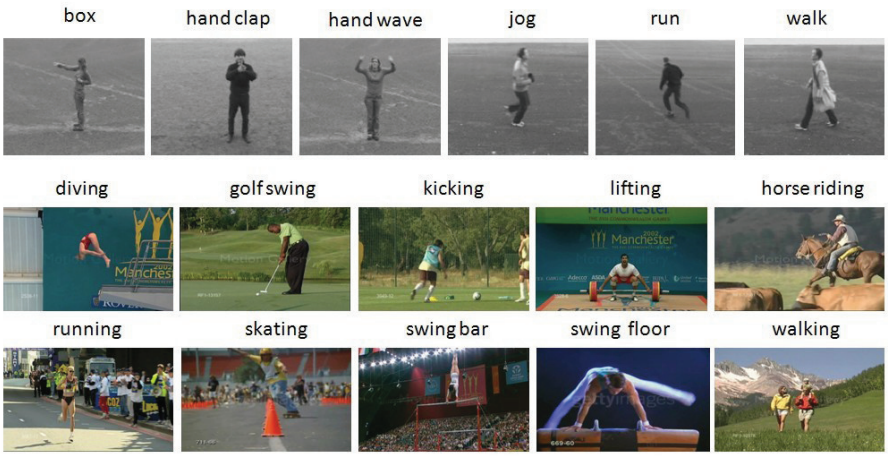

Fig. 2. Sample frames from KTH dataset (top row) and UCF sports dataset (middle and bottom rows)

4.1 Parameter Settings

Sampling and descriptor parameters. In the sampling stage, we extract 3D patches of varying sizes from the test video. The minimum size of 3D patches is 18 pixels× 18 pixels × 10 frames. We employ the sampling setting as suggested in [13], specifically, using eight spatial scales$(18, 18\sqrt{2}, 36, 36\sqrt{2}, 72, 72\sqrt{2}, 144)$, and two temporal scales $(10, 10\sqrt{2})$. Patches with all possible combinations of temporal scales and spatial scales are densely exacted with 50% overlap. Note that we discard those patches whose spatial scales exceed the video resolution, e.g. 144 pixels × 144 pixels for KTH frames of 160 × 120 resolution. We calculate HOG3D features using the executable provided by the authors of [2] with default parameters, namely, number of supporting subblocks $S^3 = 27$ and number of histogram cells $n_x = n_y = 4, n_t = 3$, thus each patch corresponds to a 960-dimensional feature vector.

Dictionary training parameters. As for dictionary construction, we set the dictionary size as 4000 empirically. In dictionary learning stage, we extract 400000 HOG3D features from 500 video clips collected from movies and sport games. In selecting dictionary training video clips, we do not impose any semantic constraints on the contents of the video clips except one criterion: all video clips must contain at least one moving subject. ℓ_1 regularization parameter λ is set to $\frac{1.2}{\sqrt{m}}$, as suggested in [15], where $m = 960$, denoting the dimension of the original signal.

4.2 Performance on KTH Dataset

The KTH dataset is a benchmark dataset to evaluate various human action recognition algorithms. It consists of six types of human actions including walk-

ing, jogging, running, boxing, hand waving and hand clapping. Each action is performed several times by 25 subjects under four different environment settings: outdoors, outdoors with scale variation, outdoors with different clothes and indoors. Currently KTH dataset contains 599 video clips in total. We follow the common experimental setup as [5], randomly dividing all the sequences into the training set (16 subjects) and the test set (9 subjects). We train a multi-class support vector machine using one-against-all method. The experiment is repeated 100 times and the average accuracy over all classes is reported.

Table 1. Comparisons to previous published results on KTH dataset

Methods	Average Precision	Experimental Setting
Niebles[4]	81.50%	leave-one-out
Jhuang[24]	91.70%	split
Fathi [25]	90.50%	split
Laptev[1]	91.80%	split
Bregonzio [26]	93.17%	leave-one-out
Kovashka [6]	94.53%	split
Our Method	**94.92%**	split

Table 1 shows that our proposed method achieves an average accuracy of 94.92%, which outperforms previous published results. Note that some of the above results [4, 26] are implemented under leave-one-out settings. Confusion matrix shown in Fig. 3 demonstrates that all the classes are predicted with satisfying precision except the pair of jogging and running. This is understandable since these two sorts of actions look ambiguous even for human beings.

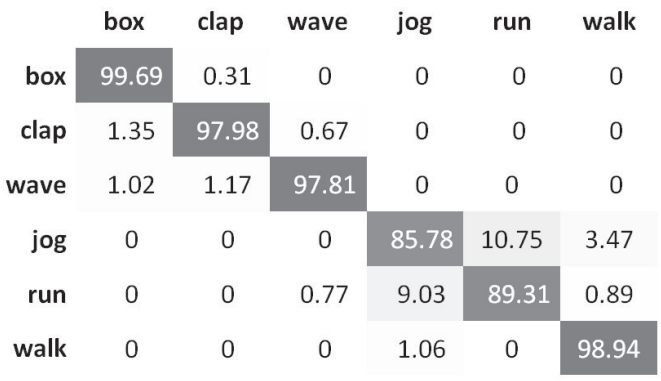

Fig. 3. Confusion matrix for the KTH dataset, the rows are the real labels while the columns are predicted ones. All the reported results are the averages of 100 rounds.

4.3 Performance on the UCF Sports Dataset

UCF sports dataset consists of 150 video clips belonging to 10 action classes including diving, golf swinging, kicking, lifting, horse riding, running, skating, swinging (around high bars), swinging (on the floor or on the pommel). All the video clips are collected from realistic sports broadcasts. Following [13, 23], we enlarge the dataset by horizontally flipping all the original clips. Similar to the setting in the KTH dataset, we train a multi-class SVM using one-against-all setting. To fairly compare with previous results, we also employ leave-one-out manner, specifically, test each original video clips successively while training all of the remaining clips. The flipped version of the test video clip is excluded from the training set. We report the average accuracy over all classes. Experimental results demonstrate that our method achieves comparable accuracy with state-of-art techniques, as shown in Table 2. Fig. 4 shows the confusion matrix for the UCF dataset. Note that UCF Sports dataset used by the authors listed in Table 2 differs slightly since some of the videos are removed from the original version due to copyright issues.

4.4 Dictionary Construction Analysis

We also explore different methods for dictionary construction on two datasets and evaluate the corresponding effects on performance. We train two dictionaries from

Table 2. Comparisons to previous published results on UCF sports dataset

Methods	Average Precision	Experimental Setting
Rodriguez[23]	69.20%	leave-one-out
Yeffet[27]	79.30%	leave-one-out
Wang [13]	85.60%	leave-one-out
Kovashka[6]	87.27%	leave-one-out
Our Method	**84.33%**	leave-one-out

Fig. 4. Confusion matrix of UCF dataset

Table 3. Comparison of different dictionary training sources on KTH dataset

Dictionary Source	Average Sparsity	Standard Deviation	Classification Accuracy
Training Set	0.65%	0.20%	91.94%
Diverse Source	**0.54%**	**0.09%**	**94.92%**

Table 4. Comparison of different dictionary training sources on UCF Sports dataset

Dictionary Source	Average Sparsity	Standard Deviation	Classification Accuracy
Training Set	0.61%	0.09%	82.09%
Diverse Source	**0.50%**	**0.06%**	**84.33%**

different sources for each dataset. The first one is trained on patches collected from the training set of the KTH dataset or UCF Sports dataset while the other one is trained with patches extracted from diverse sources including movies and sports videos. All the parameter settings for training are the same. Results show that the dictionary trained with diverse sources yields higher accuracy than the one trained merely using training sets.

To further investigate the effect of dictionary sources, we calculate the sparsity of the sparse codes transformed from the original features. For each sparse code vector $\mathbf{z} \in \mathbb{R}^S$, we define the sparsity of $\mathbf{z}$ as

$$\text{sparsity}(\mathbf{z}) = \frac{\|\mathbf{z}\|_{\ell_0}}{S}, \tag{6}$$

where zero-norm $\|\mathbf{z}\|_{\ell_0}$ denotes the number of nonzero elements in $\mathbf{z}$. We calculate the average sparsity and the corresponding standard deviation of the whole sparse code set, as shown in Table 3 and Table 4. We use these two statistics to measure how suitable a dictionary is for the given dataset. We find that the diverse source dictionary gets lower average sparsity and its corresponding standard deviation is also lower than using training set. Low sparsity ensures the more compact representations and low standard deviation over the whole dataset manifests that the diverse source dictionary is more universal for different local motion patterns. In contrast, the dictionary obtained from training set appears less sparse and more inclines to fluctuate. Although it can encode certain local features into extremely sparse form in some cases (probably due to overfitting for certain patterns), the overall sparsity of the whole sparse code set is still not comparable. This can be explained that the distribution disparity between training videos and test videos makes the dictionary learned merely from training set neither to precisely model the target subspace, nor to further impair the discriminative power of the final video representation. In contrast, the diverse source dictionary captures common patterns from diversely distributed data. It generates a set of bases with higher robustness, and thus better models the subspace which is more generative to correlate with the target data distribution. Table 3 and Table 4 can help demonstrate this point.

5 Conclusion

In this paper, we have proposed an action recognition approach using sparse coding and the local spatial-temporal descriptors. To obtain high-level video representations from local features, we also suggest to use transfer learning and maximum pooling procedure rather than the traditional histogram representation of BoW model. Experimental results demonstrate that sparse coding can provide a more accurate representation with desirable sparsity, which strengthens the discriminative power and eventually help improve the recognition accuracy. In the future work, we would like to investigate the inner structure of the dictionary and the mutual relationship of different atoms, which will be helpful to explore the semantic representations. Besides, the supervised dictionary learning algorithm will also be our future research interest.

Acknowledgements. This work is supported by the 973 Key Basic Research Program of China under Grant 2011CB302203 (2006CB303103), NSFC Key Program under Grant 60833009, National 863 Program under Grant 2009AA01Z330 and the SUNY Buffalo Faculty Startup Funding.

References

1. Laptev, I., Marszalek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: IEEE CVPR. (2008)
2. Kläser, A., Marszałek, M., Schmid, C.: A spatio-temporal descriptor based on 3D-gradients. In: British Machine Vision Conference. (2008)
3. Scovanner, P., Ali, S., Shah, M.: A 3-dimensional SIFT descriptor and its application to action recognition. In: ACM Multimedia. (2007) 357–360
4. Niebles, J.C., Wang, H., Fei-Fei, L.: Unsupervised learning of human action categories using spatial-temporal words. IJCV **79** (2008) 299–318
5. Schuldt, C., Laptev, I., Caputo, B.: Recognizing human actions: A local SVM approach. In: ICPR. (2004) 32–36
6. Kovashka, A., Grauman, K.: Learning a hierarchy of discriminative space-time neighborhood features for human action recognition. In: IEEE CVPR. (2010)
7. Ando, R.K., Zhang, T.: A framework for learning predictive structures from multiple tasks and unlabeled data. Journal of Machine Learning Research **6** (2005) 1817–1853
8. Raina, R., Battle, A., Lee, H., Packer, B., Ng, A.: Self-taught learning: transfer learning from unlabeled data. In: International Conference on Machine learning, ACM (2007) 759–766
9. Dollar, P., Rabaud, V., Cottrell, G., Belongie, S.: Behavior recognition via sparse spatio-temporal features. In: IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance. (2005) 65–72
10. Laptev, I.: On space-time interest points. IJCV **64** (2005) 107–123
11. Wong, S.F., Cipolla, R.: Extracting spatiotemporal interest points using global information. In: IEEE ICCV. (2007)
12. Dikmen, M., Lin, D., Del Pozo, A., Cao, L., Fu, Y., Huang, T.S.: A study on sampling strategies in space-time domain for recognition applications. Advances in Multimedia Modeling (2010) 465–476

13. Wang, H., Ullah, M., Kläser, A., Laptev, I., Schmid, C.: Evaluation of local spatio-temporal features for action recognition. In: British Machine Vision Conference. (2009)
14. Grosse, R., Raina, R., Kwong, H., Ng, A.Y.: Shift-invariant sparse coding for audio classification. UAI (2007)
15. Mairal, J., Bach, F., Ponce, J., Sapiro, G.: Online dictionary learning for sparse coding. In: International Conference on Machine Learning, ACM (2009) 689–696
16. Mairal, J., Bach, F., Ponce, J., Sapiro, G., Zisserman, A.: Discriminative learned dictionaries for local image analysis. In: IEEE CVPR. (2008)
17. Yang, J., Yu, K., Gong, Y., Huang, T.: Linear spatial pyramid matching using sparse coding for image classification. In: IEEE CVPR. (2009)
18. Yang, J., Yu, K., Huang, T.S.: Supervised translation-invariant sparse coding. In: IEEE CVPR. (2010)
19. Olshausen, B.A., Field, D.J.: Sparse coding with an overcomplete basis set: A strategy employed by V1? Vision research **37** (1997) 3311–3325
20. Liu, Y., Cheng, J., Xu, C., Lu, H.: Building topographic subspace model with transfer learning for sparse representation. Neurocomputing **73** (2010) 1662–1668
21. Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M., Poggio, T.: Robust object recognition with cortex-like mechanisms. IEEE T-PAMI **29** (2007) 411–426
22. Taylor, G., Bregler, C.: Learning local spatio-temporal features for activity recognition. Snowbird Learning Workshop (2010)
23. Rodriguez, M.D., Ahmed, J., Shah, M.: Action MACH: a spatio-temporal maximum average correlation height filter for action recognition. In: IEEE CVPR. (2008)
24. Jhuang, H., Serre, T., Wolf, L., Poggio, T.: A biologically inspired system for action recognition. In: IEEE ICCV. (2007)
25. Fathi, A., Mori, G.: Action recognition by learning mid-level motion features. In: IEEE CVPR. (2008)
26. Bregonzio, M., Gong, S., Xiang, T.: Recognising action as clouds of space-time interest points. In: IEEE CVPR. (2009)
27. Yeffet, L., Wolf, L.: Local trinary patterns for human action recognition. In: IEEE ICCV. (2009)