Latent Semantic Learning by Efficient Sparse Coding with Hypergraph Regularization

Zhiwu Lu and **Yuxin Peng***
Institute of Computer Science and Technology, Peking University, Beijing 100871, China
{luzhiwu,pengyuxin}@icst.pku.edu.cn

Abstract

This paper presents a novel latent semantic learning algorithm for action recognition. Through efficient sparse coding, we can learn latent semantics (i.e. high-level features) from a large vocabulary of abundant mid-level features (i.e. visual keywords). More importantly, we can capture the manifold structure hidden among mid-level features by incorporating hypergraph regularization into sparse coding. The learnt latent semantics can further be readily used for action recognition by defining a histogram intersection kernel. Different from the traditional latent semantic analysis based on topic models, our sparse coding method with hypergraph regularization can exploit the manifold structure hidden among mid-level features for latent semantic learning, which results in compact but discriminative high-level features for action recognition. We have tested our method on the commonly used KTH action dataset and the unconstrained YouTube action dataset. The experimental results show the superior performance of our method.

Introduction

Automatic recognition of human actions in videos has a wide range of applications such as video summarization, human-computer interaction, and activity surveillance. Although many impressive results have been reported on action recognition, it still remains a challenging problem (Turaga et al. 2008) owing to viewpoint changes, occlusions, and background clutters. To handle these challenges, one commonly used strategy is to adopt an intermediate representation based on spatio-temporal interest points (Schuldt, Laptev, and Caputo 2004; Dollar et al. 2005). In particular, recent work has shown promising results when the local spatio-temporal descriptors are used for bag-of-words (BOW) models (Laptev et al. 2008; Kovashka and Grauman 2010), where the local spatio-temporal features are quantized to form a visual vocabulary and each video clip is thus summarized as a histogram of visual keywords. In the following, we refer to the visual keywords as mid-level features to distinguish them from the low-level spatio-temporal features and high-level action categories.

*Corresponding author.

However, this BOW representation may suffer from the redundancy of mid-level features, since typically thousands of visual keywords are formed to obtain better performance on a relatively large action dataset. Here, it should be noted that the large vocabulary size means that the BOW representation would incur large time cost in not only vocabulary formation but also later action recognition. Moreover, the mid-level features are applied to action recognition independently and mainly the first-order statistics is considered. Intuitively, the higher-order semantic correlation between mid-level features is very useful for bridging the semantic gap in action recognition. Although the semantic information can be incorporated into the visual vocabulary using either local descriptor annotation or video annotation, the manual labeling is too expensive and tedious for a large action dataset. Therefore, to reduce the redundancy of mid-level features, in this paper, we focus on automatically extracting high-level features that are compact in size but more discriminative in terms of descriptive power for action recognition.

Previously, unsupervised methods (Niebles, Wang, and Fei-Fei 2008; Wang and Mori 2009) have been developed to learn latent semantics based on topic models. Moreover, information theory has also been applied to latent semantic analysis for action recognition in (Liu and Shah 2008; Liu, Luo, and Shah 2009). The success of latent topic or information theoretic models may be due to that the semantically similar mid-level features generally have a higher probability of co-occurring in a video across the entire dataset. In other words, the mid-level features generated from similar video contents tend to lie in the same geometric or manifold structure. However, this intrinsic information is not considered by the latent topic or information theoretic models. In the literature, very few attempts have been made to explicitly preserve the manifold geometry of the mid-level feature space when learning latent semantics from abundant mid-level features. To our best knowledge, (Liu, Yang, and Shah 2009) can be regarded as the first attempt to extract latent semantics from videos for action recognition using a manifold learning technique based on diffusion maps (Lafon and Lee 2006). Although this method has been shown to achieve better results than the information theoretic models, it requires fine parameter tuning for graph construction which can significantly affect the performance and has been noted as an inherent weakness of graph-based methods.

To address the above problems, we propose a novel latent semantic learning algorithm by efficient sparse coding with hypergraph regularization, which can capture the manifold structure hidden among mid-level features similar to (Liu, Yang, and Shah 2009) but without the need to tune any parameter for hypergraph construction. Specifically, we first formulate latent semantic learning as a sparse coding problem which can be solved efficiently similar to locality-constrained linear coding (Wang et al. 2010), and then capture the manifold structure of mid-level features by adding a hypergraph regularization term into the objective function of sparse coding. The hypergraph (Zhou, Huang, and Schölkopf 2007) used for such regularized optimization is constructed through defining the incidence matrix with the occurrences of mid-level features within videos. That is, each video clip that contains multiple mid-level features is regarded as a hyperedge, and its weight can be estimated based on the original cluster centers associated with mid-level features. This means that the hypergraph is constructed in a parameter-free manner. To summarize, we actually formulate latent semantic learning as quadratic optimization, other than time-consuming L_1-norm optimization for the traditional sparse coding. We thus can develop a very efficient algorithm for this quadratic optimization.

In this paper, we apply the learnt latent semantics to action recognition with support vector machine (SVM) by defining a histogram intersection kernel. We have evaluated our method for action recognition on the commonly used KTH action dataset (Schuldt, Laptev, and Caputo 2004) and the unconstrained YouTube action dataset (Liu, Luo, and Shah 2009). The experimental results have demonstrated the superior performance of our method. Finally, we summarize the following advantages of our method:

- Our method has made the first attempt to combine sparse coding with hypergraph regularization for latent semantic learning in the application of action recognition.
- Our method has been shown to significantly outperform other latent semantic learning methods, which turns to be more impressive given that we do not use feature pruning, multiple types of features, or spatio-temporal structural information for action recognition.
- Our method for latent semantic learning is scalable with respect to the data size and then can be applied to action recognition on large video datasets.

The remainder of this paper is organized as follows. Section 2 proposes a novel sparse coding algorithm for learning compact but discriminative latent semantics. In Section 3, we present the details of action recognition with SVM using the learnt latent semantics. In Section 4, our method is evaluated on the KTH and YouTube action datasets. Finally, Section 5 gives our conclusions.

The Proposed Algorithm

In this section, we first formulate latent semantic learning as a sparse coding problem, and then incorporate hypergraph regularization into sparse coding. Finally, we develop an efficient algorithm for the proposed sparse coding.

Latent Semantic Learning by Sparse Coding

Given a vocabulary of mid-level features $\mathcal{V}_m = \{m_i\}_{i=1}^M$, each video clip can be represented as a histogram of mid-level features $\{c_n(m_i) : i = 1, ..., M\}$, where $c_n(m_i)$ is the count of times that m_i occurs in video n ($n = 1, ..., N$). Based on this BOW representation, our goal is to learn a compact set of high-level features $\mathcal{V}_h = \{h_j\}_{j=1}^K$ from $\mathcal{V}_m$, where $K < M$. This latent semantic learning problem can be formulated by sparse coding as follows.

Since each m_i can denoted as a vector $\mathbf{x}_i = \{c_n(m_i) : n = 1, ..., N\} \in \mathcal{R}^N$, we have $X = [\mathbf{x}_1, ..., \mathbf{x}_M] \in \mathcal{R}^{N\times M}$. Given a codebook $B = [\mathbf{b}_1, ..., \mathbf{b}_K] \in \mathcal{R}^{N\times K}$ with K entries (i.e. each entry denotes a high-level feature), a sparse coding scheme can convert each $\mathbf{x}_i$ into a K-dimensional code with most elements being zeros. The corresponding L_1-norm optimization problem is defined as:

$$\min_{B,U} \sum_{i=1}^{M} ||\mathbf{x}_i - B\mathbf{u}_i||_2^2 + \lambda||\mathbf{u}_i||_1, \quad (1)$$

where $U = [\mathbf{u}_1, ..., \mathbf{u}_M] \in \mathcal{R}^{K\times M}$ is the set of sparse codes for X, and $\lambda > 0$ is a regularization parameter. The first term of the above objective function denotes the reconstruction error, while the second term denotes the sparsity constraint. It should be noted that the sparsity constraint allows the learned representation for X to capture salient patterns and thus achieve much less reconstruction error than the traditional clustering methods such as k-means.

The above sparse coding problem is convex for B when U is fixed, and is also convex for U when B is fixed. Similar to (Lee et al. 2007), we can minimize the objective function with respect to B and U alternatively. However, solving the optimization problem with respect to U usually requires computationally demanding procedures. For example, through feature-sign search (Lee et al. 2007), the L_1-norm optimization problem with respect to U can be converted to a series of quadratic optimization subproblems, which incur too large computational cost.

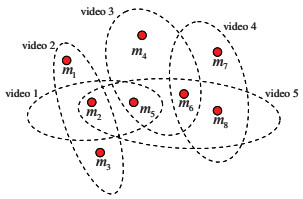
We thus develop a much more efficient sparse coding algorithm using the quadratic locality constraint instead, similar to locality-constrained linear coding (Wang et al. 2010). It should be noted that locality is more essential than sparsity, as locality must lead to sparsity but not necessary vice versa. By replacing the sparsity constraint in equation (1) with the locality constraint, we can define a new optimization problem for sparse coding:

$$\min_{B,U} \sum_{i=1}^{M} ||\mathbf{x}_i - B\mathbf{u}_i||_2^2 + \lambda\mathbf{u}_i^T D_i\mathbf{u}_i, \quad (2)$$

where D_i is a $K \times K$ diagonal matrix with its (j, j)-element $D_i(j, j) = \exp(||\mathbf{x}_i - \mathbf{b}_j||_2/\sigma)$ and σ is used for adjusting the weight decay speed for the locality constraint. Unlike the traditional sparse coding, the solution of equation (2) with respect to U can be derived analytically by

$$\begin{aligned}\mathbf{u}_i^* &= \arg\min_{\mathbf{u}_i} ||\mathbf{x}_i - B\mathbf{u}_i||_2^2 + \lambda\mathbf{u}_i^T D_i\mathbf{u}_i \\ &= (B^T B + \lambda D_i)\backslash B^T\mathbf{x}_i, \end{aligned} \quad (3)$$

which is different from (Wang et al. 2010), since the optimization problem in equation (2) has no constraints.



(a) Hypergraph

	video 1	video 2	video 3	video 4	video 5
m_1	0	0.33	0	0	0
m_2	0.50	0.33	0	0	0.25
m_3	0	0.33	0	0	0
m_4	0	0	0.33	0	0
m_5	0.50	0	0.33	0	0.25
m_6	0	0	0.33	0.33	0.25
m_7	0	0	0	0.33	0
m_8	0	0	0	0.33	0.25

(b) Incidence Matrix

Figure 1: Illustration of the hypergraph constructed in a parameter-free manner. In Figure 1(a), each dashed ellipse denotes a hyperedge (i.e. video), and each red solid node denotes the vertex (i.e. mid-level feature). The incidence matrix H of the hypergraph given by Figure 1(b) is computed using the occurrences of mid-level features within videos.

Sparse Coding with Hypergraph Regularization

To exploit the manifold structure of mid-level features for learning latent semantics from mid-level features, we further incorporate a hypergraph regularization term into the objective function of sparse coding. The hypergraph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, H, \mathbf{w}\}$ used for regularization can be constructed in a parameter-free manner as follows.

Let the vertex set $\mathcal{V} = \mathcal{V}_m = \{m_i\}_{i=1}^M$ and the hyperedge set $\mathcal{E} = \{e_j : e_j = \{m_i : c_j(m_i) > 0, i = 1, ..., M\}\}_{j=1}^N$. The incidence matrix H of $\mathcal{G}$ can be defined by

$$H_{ij} = c_j(m_i) / \sum_{m_{i'} \in e_j} c_j(m_{i'}). \quad (4)$$

Here, we consider a soft incidence matrix (i.e. $H_{ij} \in [0, 1]$), which is different from (Zhou, Huang, and Schölkopf 2007) with $H_{ij} = 1$ or 0. Moreover, we define the hyperedge weights $\mathbf{w} = \{w(e_j)\}_{j=1}^N$ by

$$w(e_j) = \frac{1}{|e_j|} \sum_{m_i \in e_j, m_{i'} \in e_j} R_{ii'}, \quad (5)$$

where $|e_j|$ denotes the number of vertices within e_j, and R is the linear kernel matrix defined with the original cluster centers associated with mid-level features. This ensures that the weight of e_j is set to a larger value when this hyperedge is more compact. Given these hyperedge weights, we can define the degree of a vertex $m_i \in \mathcal{V}$ as $d(m_i) = \sum_{e_j \in \mathcal{E}} w(e_j) H_{ij}$. For a hyperedge $e_j \in \mathcal{E}$, its degree is defined as $\delta(e_j) = \sum_{m_i \in \mathcal{V}} H_{ij}$. An example hypergraph is shown in Figure 1.

The distinct advantage of the above hypergraph construction method is that it is parameter-free. More importantly, according to (Zhou, Huang, and Schölkopf 2007), the high order correlation between mid-level features can be exploited based on the hypergraph. To define the hypergraph regularization term, we need to first compute the Laplacian matrix just as (Zhou, Huang, and Schölkopf 2007):

$$\mathcal{L} = I - D_v^{-1/2} H W D_e^{-1} H^T D_v^{-1/2}, \quad (6)$$

where Dv, De, and W denote the diagonal matrices of the vertex degrees, the hyperedge degrees, and the hyperedge weights of the hypergraph, respectively.

Based on this Laplacian matrix $\mathcal{L}$ of the hypergraph, we can define the following optimization problem for sparse coding with hypergraph regularization:

$$\min_{B,U} \left(\sum_{i=1}^M ||\mathbf{x}_i - B\mathbf{u}_i||_2^2 + \lambda \mathbf{u}_i^T D_i \mathbf{u}_i\right) + \gamma \mathrm{tr}(U \mathcal{L} U^T), \quad (7)$$

where $\gamma > 0$ is another regularization parameter. The third term of the above objective function denotes hypergraph regularization, which can help to preserve the consistence of sparse codes for similar mid-level features. In other words, the local manifold structure hidden among mid-level features can be exploited for latent semantic learning.

Efficient Sparse Coding Algorithm

The optimization problem for sparse coding with hypergraph regularization can be solved by minimizing the objective function with respect to B and U alternatively, similar to (Lee et al. 2007). It should be noted that we actually formulate latent semantic learning as quadratic optimization, other than time-consuming L_1-norm optimization for the traditional sparse coding. We thus can develop a very efficient algorithm for this quadratic optimization.

More specifically, when U is fixed in equation (7), we can update the codebook B by solving the following quadratic optimization problem using the conjugate gradient decent method (Lee et al. 2007):

$$\min_B \sum_{i=1}^M ||\mathbf{x}_i - B\mathbf{u}_i||_2^2. \quad (8)$$

To handle the scale issue associated with the codebook, we add extra normalization constraints $||\mathbf{b}_j||_2 \leq 1 (j = 1, ..., K)$ into the above optimization.

When B is fixed in equation (7), we then minimize the objective function with respect to each $\mathbf{u}_i$ alternatively and do not consider all the sparse codes $U = [\mathbf{u}_1, ..., \mathbf{u}_M]$ simultaneously. That is, when we focus on $\mathbf{u}_i$, the other sparse codes are forced to be fixed. Hence, the optimization problem in equation (7) is equivalent to:

$$\min_{\mathbf{u}_i} ||\mathbf{x}_i - B\mathbf{u}_i||_2^2 + \lambda \mathbf{u}_i^T D_i \mathbf{u}_i + \gamma L(\mathbf{u}_i), \quad (9)$$

where $L(\mathbf{u}_i) = 2(U\mathcal{L}_{.i})^T \mathbf{u}_i - \mathbf{u}_i^T \mathcal{L}_{ii} \mathbf{u}_i$, with $\mathcal{L}_{.i}$ and $\mathcal{L}_{ii}$ being the i-th column and (i, i)-element of $\mathcal{L}$, respectively. This optimization problem has a analytical solution:

$$\begin{aligned} \mathbf{u}_i^* &= \arg\min_{\mathbf{u}_i} ||\mathbf{x}_i - B\mathbf{u}_i||_2^2 + \lambda \mathbf{u}_i^T D_i \mathbf{u}_i + \gamma L(\mathbf{u}_i) \\ &= (B^T B + \lambda D_i + \gamma \mathcal{L}_{ii} I) \backslash (B^T \mathbf{x}_i + \gamma \Delta_i), \end{aligned} \quad (10)$$

where $\Delta_i = \mathcal{L}_{ii}\mathbf{u}_i - U\mathcal{L}_{.i}$ based on the old version of U.

In the following, our latent semantic learning algorithm by efficient sparse coding (ESC) with hypergraph regularization will be denoted as LapESC, given that the Laplacian matrix $\mathcal{L}$ plays a key role in hypergraph regularization. When only our efficient sparse coding (without hypergraph regularization) is used for latent semantic learning, the corresponding algorithm is denoted as ESC. Since the worst-case time complexity of our LapESC algorithm is $O((N + M)K^3)$ ($K \ll N$ and $K \ll M$), it can be run very efficiently even on a large video dataset.

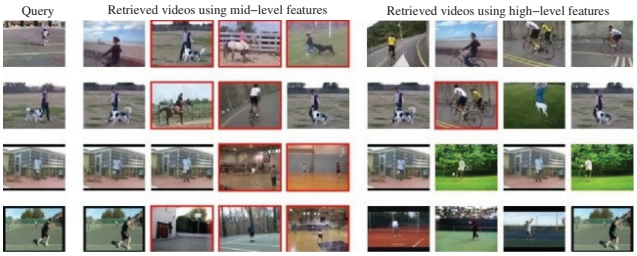


Figure 2: Retrieval examples using mid-level and high-level features on the YouTube action dataset (Liu, Luo, and Shah 2009). For each query, four videos with the highest values of the histogram intersection kernel are retrieved. The incorrectly retrieved videos (which do not come from the same action category as the query) are marked with red boxes.

Action Recognition with SVM

This section presents the details of action recognition with SVM using the latent semantics learnt by the proposed algorithm. We first derive a new semantics-aware representation (i.e. histogram of high-level features) for each video clip from the original BOW representation, and then define a histogram intersection kernel based on this new representation for action cognition with SVM.

Let $\mathcal{V}_h = \{h_j\}_{j=1}^K$ be the vocabulary of high-level features learnt from the vocabulary of mid-level features $\mathcal{V}_m = \{m_i\}_{i=1}^M$ by our LapESC. The BOW representation with $\mathcal{V}_h$ for each video can be derived from the original BOW representation with $\mathcal{V}_m$ as follows. Given the count of times $c_n(m_i)$ that mid-level feature m_i occurs in video n ($n = 1, ..., N$), the count of times $c_n(h_j)$ that high-level feature h_j occurs in this video can be estimated by:

$$c_n(h_j) = \sum_{i=1}^{M} c_n(m_i) c(m_i, h_j), \tag{11}$$

where $c(m_i, h_j) = |\mathbf{u}_i(j)|$ with $\mathbf{u}_i$ being the sparse code learnt by our LapESC for mid-level feature m_i. That is, each video is now represented as a histogram of high-level features. Similar to the traditional BOW representation, this new semantics-aware representation can be used to define a histogram intersection kernel A:

$$A(x_n, x_{\tilde{n}}) = \sum_{j=1}^{K} \min(c_n(h_j), c_{\tilde{n}}(h_j)), \tag{12}$$

where n (or $\tilde{n}$) $= 1, ..., N$. This kernel is further used for action recognition with SVM.

To provide preliminary evaluation of our learnt latent semantics, we apply the above semantics-aware kernel to action retrieval, and some retrieval examples on the YouTube action dataset (Liu, Luo, and Shah 2009) are shown in Figure 2. Here, we only learn 300 high-level features from 2,000 mid-level features by our LapESC. We can observe that the high-level features can achieve significantly better action retrieval results than the mid-level features. This observation means that the learnt high-level features can provide a semantically more succinct representation but a more discriminative descriptor of human actions than the mid-level features. Moreover, in the experiments, we can also observe that similar dominating high-level features are used to represent the videos from the same action category, although their exact meanings are yet unknown. This is also the reason why we call them "latent semantics" in this paper just as the traditional topic models. In the following, we will apply our semantics-aware representation to action recognition on the commonly used KTH action dataset (Schuldt, Laptev, and Caputo 2004) and the unconstrained YouTube action dataset (Liu, Luo, and Shah 2009).

Experimental Results

In this section, the proposed method for latent semantic learning is evaluated on two standard action datasets. We first describe the experimental setup and then compare our method with other closely related methods.

Experimental Setup

We select two different action datasets for performance evaluation. The first dataset is KTH (Schuldt, Laptev, and Caputo 2004) which contains 598 video clips from 6 action categories. The six actions are performed by 25 actors under four different scenarios. The second dataset is YouTube (Liu, Luo, and Shah 2009) which has lots of camera movement, cluttered backgrounds, and different viewing directions. This dataset contains 1,168 video clips from 11 action categories, organized into 25 relatively independent groups. To the best of our knowledge, this is the most extensive realistic action dataset in the literature.

To extract low-level features from the two action datasets, we adopt the spatio-temporal interest point detector proposed in (Dollar et al. 2005). In the following experiments, we extract 400 descriptors from each video clip for the KTH dataset, while for the YouTube dataset more descriptors (i.e. 1,600) are extracted from each vide clip since this dataset is more complex and challenging. Finally, on the two action datasets, we quantize the extracted spatio-temporal descriptors into M mid-level features by k-means clustering. Here, we only adopt very simple experimental setting for low-level feature extraction, given that our main goal is to develop a novel sparse coding method for learning compact but discriminative latent semantics in this paper.

Since the diffusion map (DM) method for latent sematic learning proposed in (Liu, Yang, and Shah 2009) has been reported to outperform other manifold learning techniques (Balasubramanian and Schwartz 2002; Belkin and Niyogi 2003) and also the information theoretic approaches (Liu and Shah 2008), we focus on comparing our method (i.e. LapESC) only with DM in this paper and do not make direct comparison with these methods. In fact, our method has been shown in later experiments to perform much better than DM, and thus we succeed in verifying the superiority of our method indirectly with respect to (Balasubramanian and Schwartz 2002; Belkin and Niyogi 2003; Liu and Shah 2008). Moreover, we also compare our method with probabilistic latent semantic analysis (PLSA) and BOW. Here,

Table 1: The comparison of the five methods with varied number of features on the KTH dataset

#features	50	100	200	300	400
LapESC	**92.2**	**93.2**	**93.3**	**94.8**	**93.6**
ESC	91.3	92.3	92.3	93.5	93.0
DM	88.4	89.1	91.5	93.3	93.0
PLSA	87.8	87.0	86.5	87.0	84.8
BOW	85.3	88.5	90.0	90.8	92.5

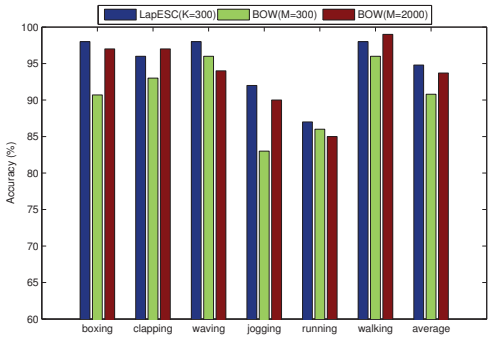


Figure 3: The comparison between LapESC and BOW with fixed number of features on the KTH dataset.

all the methods for comparison except BOW are designed to learn latent semantics from a large vocabulary of abundant mid-level features. We select $M = 2,000$ for the four latent semantic learning methods. For our LapESC algorithm, we set $\lambda = 0.1$ and $\gamma = 0.1$. To train SVM on the two datasets, we use 24 actors (or groups) for the training set and the rest for the test set, just as previous work (Liu, Luo, and Shah 2009; Liu, Yang, and Shah 2009).

Results on the KTH Dataset

The five methods are first compared on the KTH dataset when the number of features is varied from 50 to 400. The results are shown in Table 1. We can find that our method (i.e. LapESC) can achieve consistently better performance in action recognition than the other latent semantic learning methods (i.e. ESC, DM, and PLSA). This observation indeed verifies that our method can learn more compact but discriminative latent semantics through sparse coding with hypergraph regularization. Moreover, we can also find that the high-level features learnt by our method perform consistently better than the mid-level features. Although the commonly used PLSA can also learn high-level features from the abundant mid-level features, it completely fails (even not better than BOW) when more high-level features are extracted (e.g. $\geq$ 100). To our best knowledge, our method has been shown to achieve most outstanding performance among all the latent semantic learning methods.

To further show the effectiveness of our method, we need to directly compare it to BOW with $M = 2,000$. Here, we only consider our LapESC with $K = 300$. Moreover, to make extensive comparison, we take BOW with $M = 300$ as a baseline method. The comparison between our LapESC and these two BOW methods is shown in Figure 3. We can find that our LapESC performs better than BOW ($M = 2,000$) on most of the six action categories, even when the number of features is decreased from 2,000 to 300. The ability of our LapESC to achieve promising results using only a small number of features is important, because it means that our method is scalable for large action datasets. Moreover, our LapESC is shown to perform better than BOW ($M = 300$) on all the action categories when they select the same number of features.

Table 2: Comparison of our LapESC with previous methods for action recognition on the KTH dataset (MF: multiple features; SI: structural information)

Method	MF	SI	Accuracy
(Schuldt et al. 2004)	no	no	71.7
(Dollar et al. 2005)	no	no	81.2
(Laptev et al. 2008)	yes	yes	91.8
(Niebles et al. 2008)	no	no	83.3
(Liu and Shah 2008)	no	yes	94.2
(Liu, Luo, and Shah 2009)	yes	no	93.8
(Liu, Yang, and Shah 2009)	no	no	92.3
(Cao et al. 2010)	yes	no	94.1
(Kovashka et al. 2010)	no	yes	94.5
Our method	no	no	**94.8**

Table 3: The comparison of the five methods with varied number of features on the YouTube dataset

#features	50	100	200	300	400
LapESC	**54.8**	**57.1**	**61.3**	**63.9**	**63.8**
ESC	53.1	55.4	60.1	63.0	63.3
DM	51.3	55.0	58.9	60.4	62.4
PLSA	49.0	51.1	48.5	48.6	48.0
BOW	46.6	53.2	55.3	59.0	59.7

Since we focus on developing a novel sparse coding method to learn compact but discriminative latent semantics for action recognition, we only consider very simple experimental setting in this paper. For example, in the experiments, only a single type of low-level spatio-temporal descriptors are extracted from videos the same as (Dollar et al. 2005). Moreover, the learnt high-level features are directly applied to action recognition without considering their spatio-temporal layout information. That is, we do not make use of multiple types of features (Laptev et al. 2008; Liu, Luo, and Shah 2009; Cao et al. 2010), or spatio-temporal structural information (Laptev et al. 2008; Liu and Shah 2008; Kovashka and Grauman 2010) for action recognition. However, even with such simple experimental setting, our method for latent semantic learning can still achieve improvements with respect to the state of the arts, as shown in Table 2. This also provides further convincing validation of the effectiveness of our latent semantic learning by sparse coding with hypergraph regularization.

Results on the YouTube Dataset

The YouTube dataset is more complex and challenging than KTH, since it has lots of camera movement, cluttered backgrounds, and different viewing directions. We repeat the same experiments on this dataset, and the results are shown

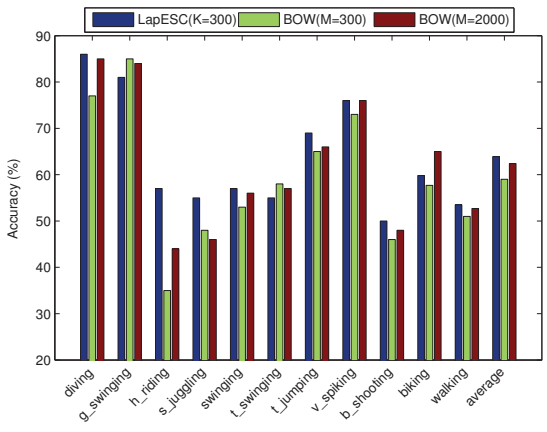


Figure 4: The comparison between LapESC and BOW with fixed number of features on the YouTube dataset.

in Table 3 and Figure 4. Here, in Table 3 the five methods for action recognition are compared when the number of features is varied from 50 to 400, while in Figure 4 we focus on the comparison between LapESC and BOW with fixed number of features. We can make the same observations on this dataset as we have done with KTH, which provides further validation of our LapESC. More importantly, the extensive evaluations on such challenging dataset actually serve to pave the way for bridging the semantic gap of video content analysis in realistic applications.

Conclusions

We have investigated the challenging problem of latent semantic learning in action recognition. To bridge the semantic gap associated with action recognition, we have proposed a novel sparse coding algorithm for learning latent semantics from a large vocabulary of mid-level features. Particularly, to capture the manifold structure hidden among mid-level features, we have incorporated hypergraph regularization into sparse coding. Although many efforts have been made to explore sparse coding for different applications in the literature, we have made the first attempt to combine sparse coding with hypergraph regularization for latent semantic learning in action recognition. The experimental results have shown that the proposed method can achieve most outstanding performance among all the latent semantic learning methods. In the future work, we will apply the proposed method to learning latent semantics from multiple local descriptors and also consider the spatio-temporal structural information of the learnt latent semantics.

Acknowledgements

The work described in this paper was fully supported by the National Natural Science Foundation of China under Grant Nos. 60873154 and 61073084, the Beijing Natural Science Foundation of China under Grant No. 4082015, and the National Development and Reform Commission High-tech Program of China under Grant No. [2010]3044.

References

Balasubramanian, M., and Schwartz, E. 2002. The isomap algorithm and topological stability. *Science* 295(5552):7–7.

Belkin, M., and Niyogi, P. 2003. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation* 15(6):1373–1396.

Cao, L.; Tian, Y.; Liu, Z.; Yao, B.; Zhang, Z.; and Huang, T. 2010. Action detection using multiple spatial-temporal interest point features. In *Proc. ICME*, 340–345.

Dollar, P.; Rabaud, V.; Cottrell, G.; and Belongie, S. 2005. Behavior recognition via sparse spatio-temporal features. In *Proc. 2nd Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, 65–72.

Kovashka, A., and Grauman, K. 2010. Learning a hierarchy of discriminative space-time neighborhood features for human action recognition. In *Proc. CVPR*, 2046–2053.

Lafon, S., and Lee, A. 2006. Diffusion maps and coarse-graining: A unified framework for dimensionality reduction, graph partitioning, and data set parameterization. *IEEE Trans. Pattern Analysis and Machine Intelligence* 28(9):1393–1403.

Laptev, I.; Marszalek, M.; Schmid, C.; and Rozenfeld, B. 2008. Learning realistic human actions from movies. In *Proc. CVPR*, 1–8.

Lee, H.; Battle, A.; Raina, R.; and Ng, A. 2007. Efficient sparse coding algorithms. In *Advances in Neural Information Processing Systems 19*, 801–808.

Liu, J., and Shah, M. 2008. Learning human actions via information maximization. In *Proc. CVPR*, 1–8.

Liu, J.; Luo, J.; and Shah, M. 2009. Recognizing realistic actions from videos in the wild. In *Proc. CVPR*, 1996–2003.

Liu, J.; Yang, Y.; and Shah, M. 2009. Learning semantic visual vocabularies using diffusion distance. In *Proc. CVPR*, 461–468.

Niebles, J.; Wang, H.; and Fei-Fei, L. 2008. Unsupervised learning of human action categories using spatial-temporal words. *IJCV* 79(3):299–318.

Schuldt, C.; Laptev, I.; and Caputo, B. 2004. Recognizing human actions: A local SVM approach. In *Proc. ICPR*, 32–36.

Turaga, P.; Chellappa, R.; Subrahmanian, V.; and Udrea, O. 2008. Machine recognition of human activities: A survey. *IEEE Trans. Circuits and Systems for Video Technology* 18(11):1473–1488.

Wang, Y., and Mori, G. 2009. Human action recognition by semilatent topic models. *IEEE Trans. Pattern Analysis and Machine Intelligence* 31(10):1762–1774.

Wang, J.; Yang, J.; Yu, K.; Lv, F.; Huang, T.; and Gong, Y. 2010. Locality-constrained linear coding for image classification. In *Proc. CVPR*, 3360–3367.

Zhou, D.; Huang, J.; and Schölkopf, B. 2007. Learning with hypergraphs: Clustering, classification, and embedding. In *Advances in Neural Information Processing Systems 19*, 1601–1608.