ABSTRACT

We propose a novel hierarchical sparse coding algorithm with spatial pooling and multi-feature fusion, to construct the low-level visual primitives, e.g., local image patches or regions, into high-level visual phrases, e.g., image patterns. In the first layer we learn the sparse codes for the visual primitives and then pass them into the second layer by spatial pooling and multi-feature fusion. In the second layer we further learn the sparse codes for the visual phrases. In order to obtain the high-quality representations for visual phrases, our proposed algorithm iteratively optimizes over the two-layer sparse codes, as well as the two-layer codebooks. Since we have explored both the spatial and multi-feature contextual information, more representative sparse codes of the visual phrases can be obtained. The experiments on image pattern discovery, image scene clustering and image classification justify the advantages of the proposed algorithm.

Index Terms— hierarchical sparse coding, spatial pooling, multi-feature fusion

1. INTRODUCTION

The development of various multimedia applications such as image categorization, scene understanding and image retrieval, needs descriptive and discriminative representations of images. The conventional way to represent images is the bag-of-words (BoW) model [1], where the pixels or visual primitives (e.g., image patches or salient regions) are encoded by vector-quantization (VQ) and then the codes are statistically pooled within local regions to obtain the representations of the regions. For example, local descriptors such as SIFT are extracted from images, upon which a codebook is trained to form a visual vocabulary for the dataset. After that, the SIFT descriptors [2] are quantized into visual words by the codebook and visual words are pooled into a statistical histogram representation for the original image.

A more biological plausible representation method is the sparse coding method which is inspired by the V1 cortex in human brain. The sparse coding representations have gained much popularity due to the state-of-the-art performance on many multimedia and computer vision problems [3–7]. For example, instead of using traditional raw image patch features [3] proposes a self-taught sparse coding method to build high-level features from unlabeled dataset and achieves good performance on image classification problems. In [5], a two-stage algorithm is proposed where sparse coding model is applied over the SIFT descriptors, followed by a spatial pyramid max pooling. When a simple linear classifier is adopted, this approach achieves the state-of-the-art performances on several benchmark datasets.

However, there still exists many limitations in the previous sparse coding work in spite of their great success. First, previous work encode local visual primitive descriptors independently, ignoring the spatial neighborhood structure among the descriptors. Due to the semantic gap between low-level features and high-level concepts [8], such sparse coding scheme cannot learn representative visual phrases (e.g., image patterns) from the unlabeled dataset. Second, previous work of-
ten focus on single type of local descriptors such as the hand-crafted SIFT descriptors. Since visual primitives can be described by different types of features, such as color, shape and texture, it is a sensible idea to fuse multiple features for providing more descriptive and discriminative information from various aspects.

This paper contributes to addressing the above issues. We introduce a hierarchical sparse coding framework to construct the low-level visual primitives into high-level visual phrases, based on spatial pooling and multi-feature fusion. We propose a novel algorithm for iteratively learning the two-layer sparse codes, as well as the two-layer codebooks from the unlabeled dataset. Since we utilize both the spatial and multi-feature contextual information, more representative sparse codes of the visual phrases can be obtained. The versatile experiments on image pattern discovery, image scene clustering and image classification justify the effectiveness of our algorithm.

2. METHODOLOGY

Let us declare the notations in this section first. We use upper case letter to denote a matrix, e.g., $X$. We then use upper case letter with two subscripts (row index and column index respectively) to denote an element of the matrix, e.g., $X_{ij}$. We use lower case letter with one subscript to denote each column of the matrix, e.g., $x_i$.

2.1. Hierarchical Sparse Coding

As shown in Fig. 1, our hierarchical framework is a two-layer structure where the first layer is the visual primitive layer including local visual primitive descriptors $X$, the corresponding visual primitive codebook $B$ and visual primitive sparse codes $Y$, the second layer is the visual phrase layer including the spatial pooled visual phrase descriptors $Z$, the corresponding visual phrase codebook $U$ and visual phrase sparse codes $V$.

In the following, we will introduce our hierarchical sparse coding framework from the bottom visual primitive layer to the upper visual phrase layer, followed by the hierarchical codebook learning.

2.1.1. Visual Primitive Sparse Coding

In our hierarchical framework, an image is represented by a set of local descriptors $X = [x_1, x_2, \cdots, x_N] \in \mathbb{R}^{d \times N}$, where each column vector $x_i$ represents a visual primitive, e.g., a local image patch or region. Given a codebook $B \in \mathbb{R}^{d \times K_1}$ where $K_1$ is the dictionary size of the codebook, the sparse coding representation $Y \in \mathbb{R}^{K_1 \times N}$ of the descriptor set $X$ can be calculated as follows,

$$
\hat{Y} = \arg \min_Y \| X - BY \|_{\ell_2}^2 + \lambda \| Y \|_{\ell_1} \tag{1}
$$

where $\| \cdot \|_{\ell_2}$ is the Frobenius norm of a matrix and $\| \cdot \|_{\ell_1}$ is the $\ell_1$ norm.

2.1.2. Spatial Pooling

In order to represent the high-level visual phrase in the images, we pool the sparse codes of local descriptors in the spatial neighborhood structure (by $K$-NN or $\epsilon$-NN), as used in [9]. The spatial pooling process is illustrated in Fig. 2.

We consider two commonly used spatial pooling methods: weighted pooling and max pooling. Assume that $z_j$ is the $j$th spatial pooled visual phrase and $y_i$ is the sparse coding representation of the local descriptor $x_i$, then weighted pooling is shown as follows:

$$
z_j = \sum_{i \in S(j)} W_{ij} y_i \tag{2}
$$

where $S(j)$ denotes the set of local descriptors contained in the $j$th visual phrase and $W$ is the weight matrix between the visual primitives and the visual phrases. Eq. 2 becomes average-pooling when the weight $W_{ij} = \frac{1}{|S(j)|}$ and sum-pooling when $W_{ij} = 1$.

The max pooling is shown in Eq. 3, where the max operation is the element-wise max operation.

$$
z_j = \max_{i \in S(j)} (|y_i|) \tag{3}
$$

As shown in [10], max pooling produces more discriminative representations when soft coding methods are used, while average-pooling on the other hand, works better when hard quantization method is applied.

2.1.3. Visual Phrase Sparse Coding

After the spatial pooling, we have the visual phrase descriptor set $Z = [z_1, z_2, \cdots, z_M] \in \mathbb{R}^{K_1 \times M}$ where $M$ is the total number of visual phrases. Similar to the sparse coding in visual primitive layer, in visual phrase layer we can calculate

Fig. 2. Illustration of spatial pooling and multi-feature fusion.
the sparse coding representation $V \in \mathbb{R}^{K_2 \times M}$ of the descriptor set $Z$ by Eq. 4:

$$
\hat{V} = \arg \min_{V} J = \|Z - UV\|_2^2 + \lambda\|V\|_{\ell_1} \quad (4)
$$

where $U \in \mathbb{R}^{K_1 \times K_2}$ is the given visual phrase codebook and $K_2$ is the dictionary size.

2.2. Hierarchical Codebook Learning

Effective visual primitive and visual phrase codings require high-quality codebooks $B$ and $U$. We now describe how to learn the codebooks from our two-layer sparse coding structure.

2.2.1. Visual Phrase Codebook Learning

To learn the visual phrase codebook $U$, we fix all other variables in Eq. 4 except for $U$ and solve Eq. 5 by the dual formulation, as discussed in [11].

$$
\hat{U} = \arg \min_{U} \|Z - UV\|_2^2 \quad \text{s.t.} \quad \|u_i\|_2^2 \leq 1, \quad \forall i = 1, \cdots, K_2.
$$

2.2.2. Back Propagation

Since we have a two-layer sparse coding structure, to learn the visual primitive codebook $B$, we have to compute the gradient of $J$ with respect to $B$ using the chain rule:

$$
\frac{\partial J}{\partial B} = \sum_j \frac{\partial J}{\partial z_j} \cdot \sum_i \frac{\partial z_j}{\partial y_i} \cdot \frac{\partial y_i}{\partial B} \quad (6)
$$

According to different spatial pooling methods, we have different back propagation results. When the weighted pooling in Eq. 2 is used, Eq. 6 becomes:

$$
\frac{\partial J}{\partial B} = \sum_j \frac{\partial J}{\partial z_j} \cdot \sum_{i \in S(j)} W_{ij} \frac{\partial y_i}{\partial B} \quad (7)
$$

When the max pooling in Eq. 3 is used, Eq. 6 becomes:

$$
\frac{\partial J}{\partial B} = \sum_j \frac{\partial J}{\partial z_j} \cdot \text{sign}(y_i^{\text{max}}) \odot \frac{\partial y_i^{\text{max}}}{\partial B} \quad (8)
$$

where $\odot$ is the element-wise product symbol and $y_i^{\text{max}}$ is obtained as follows (Again, the max operation is the element-wise max operation):

$$
y_i^{\text{max}} = \max_{i \in S(j)} (y_i) \quad (9)
$$

Therefore, we need to calculate $\frac{\partial J}{\partial z_j}$ and $\frac{\partial y_i}{\partial B}$ in Eq. 7 and 8. From Eq. 4 we can easily compute $\frac{\partial J}{\partial z_j}$, as shown in Eq. 10, therefore we will focus on $\frac{\partial y_i}{\partial B}$ in the next section.

$$
\frac{\partial J}{\partial z_j} = 2(z_j - Uv_j) \quad (10)
$$

2.2.3. Visual Primitive Codebook Learning

According to the above discussion, in order to learn the visual primitive codebook $B$, we have to compute $\frac{\partial y_i}{\partial B}$. Since $y_i$ is not directly linked to $B$ according to Eq. 1, we have to compute $\frac{\partial y_i}{\partial B}$ by the implicit differentiation method.

First we calculate the gradient with respect to $y_i$ at its minimum $\hat{y}_i$ on Eq.1, as used in [12, 13]:

$$
2(B^T Bz_i - B^T x_i)|_{y_i = \hat{y}_i} = -\lambda \cdot \text{sign}(y_i)|_{y_i = \hat{y}_i} \quad (11)
$$

Note that Eq. 11 is only correct when $y_i = \hat{y}_i$. For convenience, in the following we will admit the condition that $y_i = \hat{y}_i$ without explicitly showing it in the equations. Then we calculate the gradient with respect to $B$ on both sides of Eq. 11 and get:

$$
\frac{\partial \{2(B^T Bz_i - B^T x_i)\}}{\partial B_{mn}} = \frac{\partial \{-\lambda \cdot \text{sign}(y_i)\}}{\partial B_{mn}} \quad (12)
$$

where $B_{mn}$ is the $m^{\text{th}}$ row and $n^{\text{th}}$ column element of the codebook $B$. Note that the right-hand side of Eq. 12 is not well-defined at zero due to the non-continuous property of $\text{sign}(y_i)$, therefore we choose the non-zero coefficients from $\hat{y}_i$ to form $\hat{y}_i$ and select the corresponding codebook bases $\hat{B}$ by $\hat{y}_i$, and get the following result:

$$
\frac{\partial \{2(B^T B \hat{y}_i - B^T x_i)\}}{\partial B_{mn}} = 0 \quad (13)
$$

By expanding Eq. 13, we can further get:

$$
\frac{\partial \hat{y}_i}{\partial B_{mn}} - \frac{\partial \hat{B}^T \hat{B}}{\partial B_{mn}} \hat{y}_i - \frac{\partial \hat{B}^T x_i}{\partial B_{mn}} = 0 \quad (14)
$$

which leads to the final result of $\frac{\partial y_i}{\partial B}$:

$$
\frac{\partial \hat{y}_i}{\partial B_{mn}} = (\hat{B}^T \hat{B})^{-1} (\frac{\partial \hat{B}^T x_i}{\partial B_{mn}} - \frac{\partial \hat{B}^T \hat{B}}{\partial B_{mn}} \hat{y}_i) \quad (15)
$$

2.3. Hierarchical Sparse Coding based on Spatial Pooling

To summarize our hierarchical framework based on the spatial pooling techniques, we combine the previous discussed hierarchical sparse coding and hierarchical codebook learning and propose the HSC-SP algorithm for learning both visual primitive sparse codes and visual phrase sparse codes, as shown in Alg. 1.

2.4. Multi-feature Fusion

In the above sections, we have discussed using only one type of feature to describe visual primitives. Now let us consider fusing different types of features together. In fact, multiple features can be easily added into our hierarchical sparse coding framework.
Assume that we have $T$ types of different visual primitive descriptor sets $D = \{X^{(1)}, X^{(2)}, \ldots, X^{(T)}\}$. Then for each descriptor set $X^{(t)}$, we can get the corresponding codebook $B^{(t)}$, the sparse coding representation $Y^{(t)}$ (by Eq. 1) and the spatial pooled representation $Z^{(t)}$ (by Eq. 2 or 3). After that, we can concatenate all the $Z^{(t)}$ as follows:

\[
\mathbf{z}_i = \bigcup_{t=1}^{T} \left[ z^{(t)}_i \right]
\]

where $\bigcup \cdot$ is the vector concatenation operator. For convenience, we also define the decomposition operator $\bigcup^{-1} \cdot$ as the inverse operator of $\bigcup \cdot$.

After the concatenation, the new descriptor set $Z$ in the visual phrase layer contains both spatial pooled and multi-feature contextual information. Then in order to update each $B^{(t)}$ and $Y^{(t)}$, the back propagation Eq. 6 becomes:

\[
\frac{\partial J}{\partial B^{(t)}} = \sum_j \frac{\partial J}{\partial z_j^{(t)}} \cdot \sum_i \frac{\partial z_i^{(t)}}{\partial y_i^{(t)}} \cdot \frac{\partial y_i^{(t)}}{\partial B^{(t)}}
\]

where each $\frac{\partial J}{\partial z_j^{(t)}}$ is decomposed from $\frac{\partial J}{\partial z_j}$:

\[
\frac{\partial J}{\partial z_j^{(t)}} = \bigcup_i \left[ \frac{\partial J}{\partial z_j} \right]
\]

Although multi-feature fusion can be exploited by early fusion (concatenating visual primitive descriptor features), in our hierarchical framework we use late fusion (concatenating visual primitive sparse codes). This is due to the consideration that, early fusion method forces different feature descriptor sets to share one common codebook in the visual primitive layer, while late fusion on the other hand, allows training different codebooks for different feature descriptor sets and then fusing different visual primitive sparse codes in the visual phrase layer. The benefits of late fusion will be further explored in the experiments.

The above discussed multi-feature fusion process is also illustrated in Fig. 2. In summation, we further propose the HSC-SP-MF algorithm beyond HSC-SP algorithm in Alg. 1, which combines both spatial pooling and multi-feature fusion techniques to provide more discriminative information for the visual phrase layer in our hierarchical framework.

### 3. EXPERIMENTS

#### 3.1. Image Pattern Discovery

In the first experiment, to illustrate the effectiveness of using spatial pooling for discovering visual phrases, we evaluate the proposed HSC-SP algorithm on an LV bag image shown in Fig. 3. From the image we first extract in total 2985 SIFT points [2] as the visual primitives, upon which we then learn the visual primitive sparse codes by Eq. 1 and the visual phrase sparse codes by HSC-SP algorithm.

In the HSC-SP experiment, we construct the visual phrase layer by average-pooling 8 nearest points around each SIFT point. After learning the sparse codes, we perform k-means to cluster all the SIFT points into 4 image patterns, as shown in Fig. 4. Note that we use different colors shown in Fig. 3 to plot SIFT points located at different image patterns.

We can see that, on the one hand, visual primitive sparse codes can hardly distinguish the SIFT points stemming from different visual patterns in the LV bag image. As shown in column (a), SIFT points that represent the same visual patterns may be separated into different clusters (e.g., the 1st row and the 3rd row), while a certain cluster may contain SIFT points that belong to different visual patterns (e.g., the 3rd row).
Fig. 4. (a) Clustering results on sparse codes by Eq. 1; (b) Clustering results on sparse codes by HSC-SP algorithm. We plot each SIFT point by the color defined in Fig. 3.

Fig. 5. Sample images from MSRC-V2 dataset [14].

Fig. 6. Illustration of different types of features used to distinguish different region segmentations.

3.2. Image Scene Clustering

In the second experiment, to demonstrate the effectiveness of multi-feature fusion in our algorithm, we perform image scene clustering on the MSRC-V2 dataset [14]. We select a collection of 150 images from 5 scene categories: sheep, cow, aeroplane, boat, bicycle. Each image contains several region segmentations of the following 9 ones: grass, cow, sheep, sky, aeroplane, water, bicycle, road, boat. Sample images are shown in Fig. 5.

The ground-truth labeling of each region segmentation is provided by [15]. As shown in Fig. 6, multiple features have to be fused to distinguish different region segmentations, e.g., while color feature can distinguish sheep and cow, it cannot distinguish aeroplane, boat, or bike. Therefore following the work in [16] we describe each region segmentation with three features: Color Histogram (CH), Texton Histogram (TH) [16], and pyramid of HOG (pHOG) [17].

In the experiment, we consider region segmentations as visual primitives and the full-sized images as visual phrases in our hierarchical sparse coding framework. We use region segmentations in the same image as spatial neighbors for max pooling. After learning the visual phrases sparse codes, we perform $k$-means ($k=5$) algorithm and evaluate the clustering performance by Hungarian matching algorithm.

Table. 1 shows the final clustering results. We can see that, on the one hand, HSC-SP algorithm achieves only slightly better performance on the concatenated feature (TH+CH+pHOG, error 30%) than on the best individual feature (TH, error 32%); on the other hand, HSC-SP-MF algorithm using multi-feature fusion in the visual phrase layer, however, can significantly outperform the best individual feature (error decreases from 32.0% to 21.3%). The experiment shows that, using multi-feature late fusion can be more effective than using single individual feature or using multi-feature early fusion to learn the visual phrase sparse codes.

3.3. Image Classification

In the third experiment, we apply our algorithms for image classification on the 15-scene dataset [18]. We extract dense SIFT [5] and dense edge-SIFT [9] as visual primitive descrip-
Table 1. Clustering results on the MSRC-V2 dataset.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Method</th>
<th>Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CH</td>
<td>HSC-SP</td>
<td>41.3</td>
</tr>
<tr>
<td>TH</td>
<td>HSC-SP</td>
<td>32.0</td>
</tr>
<tr>
<td>pHOG</td>
<td>HSC-SP</td>
<td>40.0</td>
</tr>
<tr>
<td>TH+CH+pHOG</td>
<td>HSC-SP</td>
<td>30.0</td>
</tr>
<tr>
<td>TH+CH+pHOG</td>
<td>HSC-SP-MF</td>
<td>21.3</td>
</tr>
</tbody>
</table>

4. CONCLUSION

We introduce a hierarchical sparse coding framework based on spatial pooling and multi-feature fusion, to construct the low-level visual primitives into high-level visual phrases. We propose a novel algorithm for learning the two-layer sparse codes, as well as the two-layer codebooks from unlabeled dataset. Since we utilize both spatial and multi-feature contextual information, more representative sparse codes of visual phrases can be obtained. Experiments on image pattern discovery, image scene clustering and image classification justify the effectiveness of our proposed algorithm.

Table 2. Accuracy results on the 15-scene dataset.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Classification Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kernel SPM [18]</td>
<td>81.40 ± 0.50</td>
</tr>
<tr>
<td>Kernel Codebook [19]</td>
<td>76.67 ± 0.39</td>
</tr>
<tr>
<td>Sparse Coding SPM [5]</td>
<td>80.28 ± 0.93</td>
</tr>
<tr>
<td>Mid-level Feature [10]</td>
<td>85.60 ± 0.20</td>
</tr>
<tr>
<td>$\ell_p$-norm pooling SPM [20]</td>
<td>83.20</td>
</tr>
<tr>
<td>Soft assignment SPM [21]</td>
<td>82.70 ± 0.39</td>
</tr>
<tr>
<td>HSC-SP (Ours)</td>
<td>83.17 ± 0.81</td>
</tr>
<tr>
<td>HSC-SP-MF (Ours)</td>
<td>85.48 ± 0.60</td>
</tr>
</tbody>
</table>

References


