J. Vis. Commun. Image R. 28 (2015) 28–35

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

Spatially constrained sparse coding scheme for natural scene categorization ☆

Hui Zhang [a,b], Yi Liu [a,*], Bojun Xie [a,b], Jian Yu [a]

[a] Beijing Key Lab of Traffic Data Analysis and Mining, School of Computer and Information Technology, Beijing Jiaotong University, Beijing, China
[b] Key Lab. of Machine Learning and Computational Intelligence, College of Mathematics and Computer Science, Hebei University, Hebei, China

ARTICLE INFO

Article history:
Received 17 November 2013
Accepted 6 January 2015
Available online 14 January 2015

Keywords:
Scene categorization
Receptive fields learning
Boosting
Sparse coding
k-nearest neighbor
Voting
Image classification
Pooling

ABSTRACT

Coding and pooling, the major two sequential procedures in sparse coding based scene categorization systems, have drawn much attention in recent years. Yet improvements have been made for coding or pooling separately, this paper proposes a spatially constrained scheme for sparse coding on both steps. Specifically, we employ the *m*-nearest neighbors of a local feature in the image space to improve the consistency of coding. The benefit is that similar image features will be encoded with similar codewords, which reduced the stochasticity of a conventional coding strategy. We also show that the Viola–Jones algorithm, which is well-known in face detection, can be tailored to learning receptive fields, embedding the spatially constrained information on the pooling step. Extensive experiments on the UIUC sport event, 15 natural scenes and the Caltech 101 database suggests that scene categorization performance of several popular algorithms can be ubiquitously improved by incorporating the proposed two spatially constrained sparse coding scheme.

1. Introduction

Scene categorization system usually contains two sub-modules: feature representation and classifier learning. Feature representation consists of three components: feature extraction, feature coding and feature pooling. Feature extraction generates the description of a local image patch, usually using scale-invariant feature transform [1] (SIFT) or histogram of oriented gradient [2] (HOG) descriptors. Feature coding results a codeword representation of the local patch based on a pre-trained dictionary. Feature pooling is used to produce a word frequency feature vector for the image, based on the summary statistics of encoded feature. After the feature representation module, linear or non-linear classifiers can be learned on the set of feature vectors.

One of the most popular feature coding scheme is the bag-of-features [3,4] (BoF) model. First, SIFT descriptors are densely extracted on the entire set of images. Second, K-means algorithm is performed on the random selected SIFT descriptors to generate the dictionary (codebook) where the cluster centers are called codewords. The dictionary can be seen as a set of basis used to represent the SIFT descriptors by the codeword IDs. In the BoF model, each SIFT descriptor is assigned to and represented by the nearest codeword. Finally, after average pooling, the histogram of codewords occurrence frequencies can be used to represent the image.

BoF model is expected to have less discriminant ability without considering the spatial information of local features on the image plane. To incorporate such information, spatial pyramid matching [5] (SPM) divides the whole image into fixed rectangular subregions and combines all the feature vectors for each region. The success of SPM depends on two things: one is its feature representation with spatial information and the other is the pyramid match kernel [6]. Both BoF and SPM use hard vector quantization (representing each feature using only *one* codeword) to generate the feature representation, which has large quantization error. To reduce this error, sparse codes spatial pyramid matching [7] (ScSPM) is proposed. ScSPM extends SPM in two aspects. First, instead of using the hard vector quantization step, ScSPM uses sparse coding instead (representing each feature using *multiple* codewords). Second, it uses max pooling operation (which takes the maximum of the sparse coding coefficients for each codeword and over the pooling region) to replace average pooling (which takes the average value instead of the max value) used in classic BoF. In this way, ScSPM achieved higher performance on several benchmarks.

The success of ScSPM spurred recent research on sparse coding for image classification. Boureau et al. [8] evaluated several coding

☆ This paper has been recommended for acceptance by Yehoshua Zeevi.

* Corresponding author.
E-mail address: yiliu@bjtu.edu.cn (Y. Liu).

http://dx.doi.org/10.1016/j.jvcir.2015.01.004
1047-3203/

and pooling schemes and proposed macro-features to improve the classification accuracy. Wang et al. [9] used the m-nearest codewords to represent local patches (LLC), while Liu et al. [10] proposed the soft assignment coding scheme (LSC) and Huang et al. [11] enhanced codewords with salient measure to represent local patches (SC). Besides image classification, sparse coding has also been extensively used in other fields, for example, Yu et al. recently proposed a sparse patch alignment approach for image clustering [12] and used sparse coding to perform click prediction [13]. Finally, in recent years, deep learning methods have also been used to infer deep sparse codes [14], extract image features and perform image classification [15,16], which are related to this work in a broad sense.

Yet improvements of sparse coding algorithms have been made for coding or pooling separately, little work explicitly considered modeling spatial constraints/relationships in image classification tasks. Our proposed method specifically addressed this limitation. In particular, we propose two new methods to model spatial relationships of codewords in the general sparse coding framework, one in the coding step and the other in the pooling step. A preliminary conference version of this work [17] has addressed the adaptive spatial pooling issue. However, here we show that by also incorporating spatial information in the coding step results in an even better scene classification performance.

The key contributions of this work are two-folds: First, we propose to use the m-nearest neighbors of a local feature to vote for the sparse coding result of that feature in the image space, which encourages similar image features to be encoded by similar codewords. Second, we propose a boosting-based approach to adaptively learning receptive fields, which imposed informative spatial constrains to the pooling step of sparse coding algorithms for better image classification.

Other researchers have also considered improving the spatial consistency in the coding step. For example, Gao et al. [18] used Laplacian matrix to penalize coding inconsistency while Shabou [19] formulated codewords selection as a labeling problem. However, both [18,19] used complex optimization algorithms to search for the optimal codewords. They employ complicated mathematical formulation and are also computationally expensive. In contrast, our method only used the m-nearest neighbors of a local feature in the image space to vote for a spatially consistent coding result, which is computationally faster, easier to formulate, and has a higher flexibility.

Similarly, a large amount of existing work have also tried to add more information (e.g., by using large codebook [20], using more pooling regions [21], incorporating object location information [22] and using hierarchical pooling [23]) in the pooling step to achieve higher performance. However, high dimensional feature representations pose significant challenges in machine learning and computation, which are not always viable. Different from their approaches, our boosting-based approach to receptive field learning only require a small codebook size and naturally leads to a low dimensional feature representation, which is more efficient in both the computation and classifier design. In particular, our method suggests that the Viola–Jones algorithm [24], which is well-known in face detection, can actually be tailored to learning receptive fields for the pooling step of sparse coding algorithms.

This paper is organized as follows: In Section 2, we briefly review the sparse coding framework for scene categorization. In Section 3, we introduce two novel methods to incorporate spatial constrained information into sparse coding algorithms: the spatial constrained coding method based on codewords voting (Section 3.1) and the spatial constrained pooling method based on the boosting algorithm (Section 3.2). Then, we quantify the performance of the proposed methods against existing algorithms on several benchmarks in Section 4. Finally, we conclude this work in Section 5.

2. The sparse coding framework for scene categorization

We first review the sparse coding framework for scene categorization. Denote $X = \{x_1, x_2, \ldots, x_N\} \in R^{d\times N}$ be the SIFT features extracted from image I, d the dimensionality of SIFT descriptors (by default $d = 128$). $B = \{b_1, b_2, \ldots, b_M\} \in R^{d\times M}$ the codebook with M codewords, usually obtained by performing the K-means algorithm on a random subset of SIFT features. In general, the codebook is overcompleted, i.e. the size of codebook is much larger than the dimension of feature (i.e. $M \gg d$).

The coding step is to represent a feature x_p by some selected codewords in B. The encoded vector for local feature x_p is denoted by $c_p \in R^{M\times 1}$. Different coding method employs different strategies to choose the codewords. After each local feature is coded by the selected codewords, we obtain code matrix $C = \{c_1, c_2, \ldots, c_N\} \in R^{M\times N}$ for image I. Here, we briefly introduce the coding formulation for three popular sparse coding methods: locality constrained linear coding [9] (LLC), soft assignment coding [10] (LSC) and salient coding [11] (SC).

LLC is an extension of ScSPM [7], which assumes that features lie on a lower dimensional manifold and can be approximately represented by the m-nearest codewords in B. The approximated LLC is formulated by Eq. (1),

$$\begin{aligned} &\min_{\tilde{C}} \sum_{p=1}^{N} \|x_p - B_p \tilde{c_p}\|^2 \\ &s.t.\ \mathbf{1}^T \tilde{c_p} = 1, \quad \forall p \end{aligned} \tag{1}$$

where x_p is a local feature, $B_p = \{b_1^p, b_2^p, \ldots, b_m^p\}$ are the m-nearest codewords of x_p, $\tilde{c_p}$ is the encoded vector for local feature x_p. $\mathbf{1}$ is a column vector with all entries one so that the sum of elements in $\tilde{c_p}$ equal one, this constraint is used to solving the least-squares problem.

In LSC, features are encoded by the m-nearest codewords of x_p, like LLC. However, LSC uses distance ratios only to determine the coding coefficients instead of resolving the reconstruction problem in LLC. The coding step of LSC is given in Eq. (2) below,

$$\begin{aligned} c_{pj} &= \frac{exp(-\beta \hat{d}(x_p, b_j))}{\sum_{l=1}^{m} exp(-\beta \hat{d}(x_p, b_l))} \\ \hat{d}(x_p, b_l) &= \begin{cases} d(x_p, b_l), & \text{if } b_l \in \{b_1^p, b_2^p, \ldots, b_m^p\} \\ +\infty, & \text{otherwise} \end{cases} \end{aligned} \tag{2}$$

where c_{pj} is the encoded value for local feature x_p by codeword b_j^p, $d(x_p, b_l)$ is the Euclidean distance between feature x_p and codeword b_l.

In contrast, SC encodes the feature using a saliency function instead of solving a linear system. It assigns more weight on the nearest codeword that is much closer to the feature than other codewords. The coding step of SC is given in Eq. (3),

$$c_{pj} = \begin{cases} \emptyset(\|x_p - b_j^p\|^2 / \frac{1}{m-1} \sum_{s\neq j}^{m} \|x_p - b_s^p\|^2), & j = \underset{s=1,2,\ldots,m}{\operatorname{argmin}}(\|x_p - b_s^p\|^2) \\ +\infty, & \text{otherwise} \end{cases} \tag{3}$$

where c_{pj} is the encoded value for local feature x_p by the nearest codeword b_j^p, $\emptyset$ is a monotonically decreasing function, in [11], $\emptyset(x) = 1 - x$. $\{b_1^p, b_2^p \ldots, b_m^p\}$ is the m-nearest codewords of x_p.

After feature encoding, we obtain sparse coding coefficient matrix $C = \{c_1, c_2, \ldots, c_N\} \in R^{M\times N}$ for image I, where each column represents the sparse coefficients for encoding each feature using the codebook. However, this matrix is too large to be practically used by any conventional classifier. To avoid this problem, a