Max-Margin Multiple-Instance Dictionary Learning

Xinggang Wang[†] WXGHUST@GMAIL.COM
Baoyuan Wang[‡] BAOYUANW@MICROSOFT.COM
Xiang Bai[†] XBAI@HUST.EDU.CN
Wenyu Liu[†] LIUWY@HUST.EDU.CN
Zhuowen Tu[‡,§] ZHUOWEN.TU@GMAIL.COM

[†]Department of Electronics and Information Engineering, Huazhong University of Science and Technology
[‡]Microsoft Research Asia
[§]Lab of Neuro Imaging and Department of Computer Science, University of California, Los Angeles

Abstract

Dictionary learning has became an increasingly important task in machine learning, as it is fundamental to the representation problem. A number of emerging techniques specifically include a codebook learning step, in which a critical knowledge abstraction process is carried out. Existing approaches in dictionary (codebook) learning are either generative (unsupervised e.g. k-means) or discriminative (supervised e.g. extremely randomized forests). In this paper, we propose a multiple instance learning (MIL) strategy (along the line of weakly supervised learning) for dictionary learning. Each code is represented by a classifier, such as a linear SVM, which naturally performs metric fusion for multi-channel features. We design a formulation to simultaneously learn mixtures of codes by maximizing classification margins in MIL. State-of-the-art results are observed in image classification benchmarks based on the learned codebooks, which observe both compactness and effectiveness.

1. Introduction

Finding an effective and efficient representation remains as one of the most fundamental problems in machine learning. A number of important developments in the recent machine learning literature (Blei et al., 2003; LeCun et al., 2004; Hinton et al., 2006; Serre & Poggio, 2010) have an important dictionary learning stage, either explicitly or implicitly. For example the bag of words (BoW) model (Blei et al., 2003), due to its simplicity and flexibility, has been adopted in a wide variety of applications, in document analysis in particular. In computer vision, the spatial pyramid matching algorithm (SPM) (Lazebnik et al., 2006) has demonstrated its success in image classification and categorization.

In this paper, we propose a general dictionary learning method through weakly-supervised learning, in particular multiple instance learning (Dietterich et al., 1997). Our method can be applied in many domains and here we focus on image-based codebook learning for classification. On one hand, visual patterns are given as multi-variate variables and often live in high dimensional spaces; on the other hand, there are intrinsic structural information in these patterns, which might be unfolded into lower dimensional manifolds. Dictionary (codebook) learning provides a way of knowledge abstraction upon which further architectures can be built.

In computer vision applications, given a learned codebook, each image patch in an input image is either assigned with a code or a distribution (on learned codebook); then image representation can be built based on the encoded image patches. In the experiments of this paper, each input sample is denoted by a feature vector, such as SIFT (Lowe, 2004) and LBP (Ojala et al., 1996), extracted from an image patch (say 48 $\times$ 48). Using codebook has several advantages: (1) explicit representations are often enforced; (2) dimensionality reduction is performed through quantization; (3) it facilitates hierarchical representations; (4) spatial configuration can be also imposed. A direct way to learning a codebook is by performing clustering, e.g. the k-means algorithm (Duda et al., 2000). Several

Proceedings of the 30th International Conference on Machine Learning, Atlanta, Georgia, USA, 2013. JMLR: W&CP volume 28.

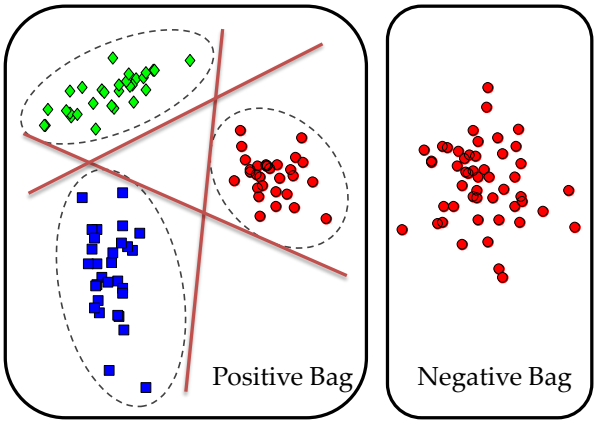

Figure 1. Illustration of max-margin multiple-instance dictionary learning. **Left**: A positive bag containing both positive instances (rectangles and diamonds) and negative instances (circles). In this paper, we assume that positive instances may belong to different clusters. We aim to learn max-margin classifiers to separate the instances in positive bags. **Right**: A negative bag with all negative instances.

approaches have been proposed (Jurie & Triggs, 2005; Lazebnik & Raginsky, 2009) and one often builds further models on top of a learned codebook (Fei-Fei & Perona, 2005). However, a direct clustering approach is often sensitive to: (1) initialization, (2) number of codes, and (3) metric (distance) of the multi-channel features. In a supervised setting where the labels are available, several discriminative codebook learning approaches have also been proposed (Moosmann et al., 2006; Yang et al., 2008; Moosmann et al., 2008).

Instead of learning a dictionary in a fully unsupervised way (e.g. k-means) or supervised way (e.g. random forests (Moosmann et al., 2008)), we take a different path to dictionary learning through a multiple instance learning strategy. Given a set of training images with each image assigned with a class label, we treat one particular class of images as positive bags, and the rest images as the negative bags; dense image patches are collected as the instances in each bag. Our algorithm then tries to learn multiple linear SVMs for two purposes: (1) to identify those patches (instances) which are genuine to the class of interest; (2) to learn linear SVM classifiers to classify those identified patches. These linear SVMs naturally cluster the positive instances into different clusters. We repeat the process for all the image classes and collect the learned classifiers, which become our dictionary (codebook). Due to the difference to the codes learned in standard ways, we call each learned linear SVM as *generalized code*, or **G-code**. In this paper, we propose a learning framework to achieve the above goal, which has the following properties: (1) a multiple instance learning strategy is proposed for dictionary learning (an uncommon direction); (2) each code is represented by a linear SVM which naturally performs metric fusion for multi-channel features; (3) we design a formulation to simultaneously learn mixtures of codes by maximizing classification margins in MIL. State-of-the-art results are observed in image classification benchmarks with significantly smaller dictionary (e.g. only 1/6) than the competing methods. Next, we briefly discuss the relations between our work and the existing literature in dictionary learning.

2. Related Work

Based on low-level descriptors (Lowe, 2004; Ojala et al., 1996), bag of words (BoW) model (Fei-Fei & Perona, 2005) using codebooks is widely adapted for image classification and object detection. On one hand, unsupervised learning, such as K-means, is already demonstrated its popularity for codebook learning in many applications. On the other hand, people found that supervised learning methods tend to produce more discriminative codebooks, as described in recent works (Moosmann et al., 2008; Yang et al., 2008; 2010; Jiang et al., 2012; Mairal et al., 2010; Winn et al., 2005). More recently, there are some attempts (Parizi et al., 2012; Singh et al., 2012; Zhu et al., 2012) tried to involve latent structures during both the learning and inference process for image classification, however, their target is not for generic dictionary learning. Different from all the previous work, in this paper we try to explicitly perform the dictionary learning along the line of weakly-supervised learning.

Strongly supervised methods like Attributes (Farhadi et al., 2009; Pechyony & Vapnik, 2010; Parikh & Grauman, 2011), Poselets (Bourdev & Malik, 2009), and Object Bank (Li et al., 2010), have shown to be promising. In our approach, we only use the image-level labels with no additional human annotations to learn the codebook. In Classemes (Torresani et al., 2010), the emphasis was made on learning an image-level representation for image search. From the viewpoint of multiple instance learning, our proposed method is related to multiple component learning (MCL) (Dollár et al., 2008) and multiple clustered instance learning (MCIL) (Xu et al., 2012). Due to the lack of explicit competition among the clusters, however, both MCL and MCIL are hard to generalize to solve the codebook learning problem. From the viewpoint of multiple instance clustering, our proposed method is related to M^{3}MIML (Zhang & Zhou, 2008) and M^{3}IC (Zhang et al., 2009) methods. However,

both M^{3}MIML and M^{3}IC try to maximize the bag-level margin, we instead maximize the instance-level margin with the MIL constrains, which is quite different from (Zhang & Zhou, 2008; Zhang et al., 2009) in problem formulation, research motivation, and task objective.

3. Max-margin Multiple-Instance Dictionary Learning

3.1. Notation and Motivation

We first briefly give the general notations of MIL (Dietterich et al., 1997). In MIL, we are given a set of bags $X = \{X_1, \ldots, X_n\}$, each of which contains a set of instances $X_i = \{\mathbf{x}_{i1}, \ldots, \mathbf{x}_{im}\}$; and each instance is denoted as a d-dimensional vector $\mathbf{x}_{ij} \in \mathbf{R}^{d\times 1}$. In addition, every bag is also associated with a bag label $Y_i \in \{0, 1\}$; and every instance is associated with an instance label $y_{ij} \in \{0, 1\}$ as well. The relation between bag label and instance labels is interpreted in the following way: if $Y_i = 0$, then $y_{ij} = 0$ for all $j \in [1, \ldots, m]$, *i.e.*, no instance in the bag is positive. If $Y_i = 1$, then at least one instance $\mathbf{x}_{ij} \in X_i$ is a positive instance of the underlying concept.

To use MIL for dictionary learning, we consider an image as a bag, and a patch (or region) within the image as an instance. Given a set of images from multiple classes with the corresponding class labels, we treat the images of one typical class as positive images, and the rest ones as negative images. Intuitively, for each image, if it is labelled as positive, then at least one patch within it should be treated as a positive patch; while if it is labelled as negative, then all patches within it should be labeled as negative patches. Take the images in 15 Scene dataset (Lazebnik et al., 2006) as an example, if *highway* class is the positive class, then the *mountain* class falls into the negative class; image patches of sky appear in both classes will be treated as negative patches. As shown in Fig. 1, we assume *positive patches are drawn from multiple clusters*, and we view negative patches from a separate negative cluster. The goal of this paper is to learn max-margin classifiers to classify all patches into different clusters, and illustrate the learned classifiers (G-codes) for image categorization/classification. Our dictionary learning problem involves two subproblems: (1) discriminative mixture model learning and (2) automatic instance label assignment (which cluster a patch might belong to). It seems that MIL is a natural way to address the above problem. Hence, in the following, we will first give a naive solution, and then provide detailed formulation and solution to our proposed *max-margin multiple-instance dictionary learning* (MMDL) problem.

> Given positive bags and negative bags, do the following two steps.
>
> **MIL step:** Run mi-SVM on the input positive and negative bags, and obtain positive instances in positive bags.
>
> **Clustering step:** Run k-means on the positive instances obtained by mi-SVM.

Figure 2. A naive solution for multiple-instance dictionary learning.

3.2. A Naive Solution

A naive way to use MIL for dictionary learning is to first run the standard MIL (e.g. mi-SVM) to select positive instances, then run a clustering algorithm to build the dictionary. In Fig. 2, we show a naive solution based on the mi-SVM (Andrews et al., 2002) and k-means algorithm. This method typically treats multiple instance learning and mixture models learning as two independent steps, which is not necessarily the optimal solution. In the following, we will introduce our formulation to perform these two steps simultaneously, which is called max-margin multiple-instance dictionary learning (MMDL).

3.3. Formulation of MMDL

In MMDL, we explicitly maximize the margins between different clusters. To achieve this goal, we build the MMDL based on multi-class SVM, *e.g.*, Crammer and Singer's multi-class SVM in (Crammer & Singer, 2002). Without loss of generality, we simply adapt the linear classifier, which is defined as $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$. Each cluster is then associated with a specific linear classifier. Due to the flexibility introduced by the multi-class SVM formulation, it's very natural to allow all the classifiers to compete with each other during the learning process. In this paper, we introduce a cluster label as latent variable, $z_{ij} \in \{0, 1, \ldots, K\}$, for each instance. If $z_{ij} = k \in \{1, \ldots, K\}$, instance $\mathbf{x}_{ij}$ is in the kth positive cluster. Otherwise, $z_{ij} = 0$, x_{ij} is in the negative cluster. Furthermore, we also define a weighting matrix $\mathbf{W} = [\mathbf{w}_0, \mathbf{w}_1, \ldots, \mathbf{w}_K], \mathbf{w}_k \in \mathbf{R}^{d\times 1}, k \in \{0, 1, \ldots, K\}$ as linear classifiers stacked in each column, where $\mathbf{w}_k$ represents the k-th cluster model. Note that, $\mathbf{w}_0$ denotes the negative cluster model. Hence, instance $\mathbf{x}_{ij}$ can be classified by:

$$z_{ij} = \arg\max_k \mathbf{w}_k^T \mathbf{x}_{ij} \tag{1}$$

Input: Positive bags, negative bags, and the number of positive clusters K.
Initialization: For instances in negative bags, we set $z_{ij} = 0$. For instances in positive bags, we use k-means algorithm to divide all these instances into K clusters, and set cluster label to be index of clustering center. Instance weight is set to 1, $p_{ij} = 1$, for all instances in positive bags.

We iterate the following two steps for N (N is typically set to 5 in our experiments) times:

Optimize W: we sample p^s portion of the instances per positive bag according to instance weight p_{ij} and take all negative instances to form a training set $\mathcal{D}'$; since cluster labels are known, we solve the multi-class SVM optimization problem to obtain $\mathbf{W}$,

$$\min_{\mathbf{W}} \quad \sum_{k=0}^{K} \parallel \mathbf{w}_k \parallel^2 + \lambda \sum_{ij} \max(0, 1 + \mathbf{w}_{r_{ij}}^T \mathbf{x}_{ij} - \mathbf{w}_{z_{ij}}^T \mathbf{x}_{ij})$$

in which $\mathbf{x}_{ij} \in \mathcal{D}'$ and $r_{ij} = \arg\max_{k \in \{0,\ldots,K\}, k \neq z_{ij}} \mathbf{w}_k^T \mathbf{x}_{ij}$.

Update p_{ij} and z_{ij}: for all instances in positive bags:

1. Update p_{ij} according to Eq. (4)
2. Update $z_{ij} = \arg\max_{k \in \{1,\ldots,K\}} \left(\mathbf{w}_k^T x_{ij} - \mathbf{w}_0^T x_{ij} \right)$

Output: The learned classifiers $\mathbf{W}$.

Figure 3. Optimization algorithm for MMDL

With the above definitions, the objective function becomes

$$\begin{aligned} \min_{\mathbf{W}, z_{ij}} \quad & \sum_{k=0}^{K} \parallel \mathbf{w}_k \parallel^2 + \lambda \sum_{ij} \max(0, 1 + \mathbf{w}_{r_{ij}}^T \mathbf{x}_{ij} - \mathbf{w}_{z_{ij}}^T \mathbf{x}_{ij}) \\ \text{s.t.} \quad & \text{if } Y_i = 1, \sum_j z_{ij} > 0, \text{and if } Y_i = 0, z_{ij} = 0, \end{aligned} \tag{2}$$

where $r_{ij} = \arg\max_{k \in \{0,\ldots,K\}, k \neq z_{ij}} \mathbf{w}_k^T x_{ij}$.

In Eq. (2), the first term, $\sum_{k=0}^{K} \parallel \mathbf{w}_k \parallel^2$ is for the margin regularization, while the second term is the multi-class hinge-loss denoted as $\ell(\mathbf{W}; (\mathbf{x}_{ij}, z_{ij}))$.

$$\ell(\mathbf{W}; (\mathbf{x}_{ij}, z_{ij})) = \sum_{ij} \max(0, 1 + \mathbf{w}_{r_{ij}}^T \mathbf{x}_{ij} - \mathbf{w}_{z_{ij}}^T \mathbf{x}_{ij}) \tag{3}$$

Parameter λ controls the relative importance between the two terms. The loss function $\ell(\mathbf{W}; (\mathbf{x}_{ij}, z_{ij}))$ explicitly maximizes soft-margins between all $K+1$ clusters. Constraints in Eq. (2) are equivalent to constraints in MIL. Because $\sum_j z_{ij} > 0 \Leftrightarrow \sum_j y_{ij} > 0$ and $z_{ij} = 0 \Leftrightarrow y_{ij} = 0$.

This MMDL formulation leads to a non-convex optimization problem. However, this problem is *semi-convex* (Felzenszwalb et al., 2010) since optimization problem becomes convex once latent information is specified for the instances in the positive bags. In (Felzenszwalb et al., 2010), a "coordinate descend" method is proposed to address this kind of problem, which guarantees to give a local optimum. Our problem is even harder, since we do not know the number of positive instances in each positive bag.

3.4. Learning Strategies of MMDL

At first, we denote training set as $\mathcal{D} = \{X_1, \ldots, X_n\}$ including all positive and negative bags for training. Then we define *instance weight* as follows:

$$\begin{aligned} p_{ij} &= \text{sigmoid}\left(\max_{k \in \{1,\ldots,K\}} \left(\mathbf{w}_k^T \mathbf{x}_{ij} - \mathbf{w}_0^T \mathbf{x}_{ij} \right) / \sigma \right) \\ &= \left(1 + \exp\left(- \max_{k \in \{1,\ldots,K\}} \left(\mathbf{w}_k^T \mathbf{x}_{ij} - \mathbf{w}_0^T \mathbf{x}_{ij} \right) / \sigma \right) \right)^{-1} \end{aligned} \tag{4}$$

p_{ij} shows "positiveness" of the instance. It is determined by the maximal difference of SVM decision function value between a positive cluster and the negative cluster which is $\max_{k \in \{1,\ldots,K\}} \left(\mathbf{w}_k^T x_{ij} - \mathbf{w}_0^T x_{ij} \right)$. Sigmoid function is used for mapping the difference of SVM decision function value into the range of $(0, 1)$. σ is a parameter for normalization.

In the next step, we solve the problem in (2) using coordinate descend in a stochastic way which is summarized in Fig. 3. We form a new training set $\mathcal{D}'$ out of the original $\mathcal{D}$ by sampling instances from each bag based on p_{ij}. Because latent variables are only effective for instances in positive bags, we take all instances in negative bags into $\mathcal{D}'$. In addition, we only sample p^s portion of the instances per positive bag. Initially, the instance weights are equal for all positives. After the sampling step, the data set $\mathcal{D}'_0$ is used to train

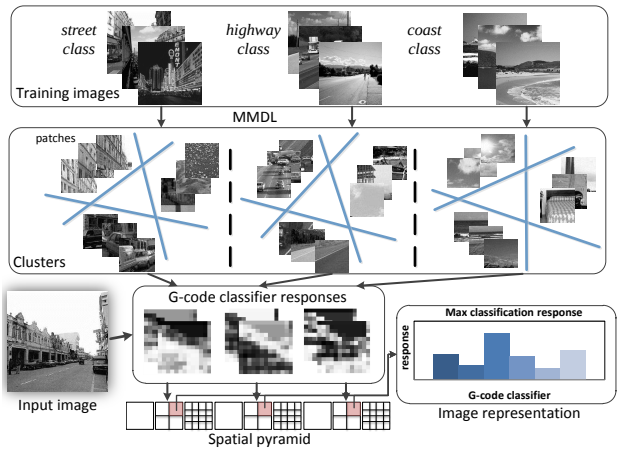

Figure 4. Illustration of MMDL for image classification. Given a set images (in the first row) from multiple classes, we divide image patches into different clusters and obtain G-codes through MMDL. Some patches in the learned clusters (both positive and negative) are shown in the second row. For an input image, image representation is build based on response maps of G-code classifiers and spatial pyramid (shown in the third row).

a standard multi-class SVM classifier f_0. This completes the **Optimize W** step. Once we get f_0, we can apply it to the original positive bags to perform **Update** p_{ij} **and** z_{ij} step. Then, we sample another p^s portion of instances from each positive bag based on the classification results, forming a new dataset $\mathcal{D}'_1$ and then obtain f_1. This process is repeated until the desired number of iterations N is reached. Sampling instances according to their "positiveness" makes sure that a portion of instances in positive bag have positive instance labels. This satisfies the constraint in Eq. (2). In addition, this sampling procedure can also increase the efficiency of our optimization algorithm.

4. MMDL for Image Representation

A learned dictionary consists of a set of linear classifiers (G-code classifiers) for different patch clusters from different image classes. Similar to the way in object bank (Li et al., 2010), our image representation is constructed from the responses of G-code classifiers. Our MMDL framework is illustrated in Fig. 4. Suppose there are M categories in the dataset, for each image category, we use the training images in this category as positive examples, and the rest training images as negative examples. Through MMDL, $K+1$ G-code classifiers are learned. Given an input image, patch-level image features are densely extracted. Suppose $\mathbf{x}$ is a local feature vector, response of $\mathbf{x}$ given by the kth G-code is $\mathbf{w}_k^T\mathbf{x}, k \in \{0, 1, \ldots, K\}$. Thus, we can obtain a response map for each G-code classifier. For each response map, a three-level spatial pyramid representation (Lazebnik et al., 2006) is used, resulting in $(1^2 + 2^2 + 4^2)$ grids; the maximal response for each G-code classifier in each grid is computed, resulting in $M \times (K+1)$ length feature vector for each grid. A concatenation of features in all grids leads to a compact image descriptor of the input image.

Note that the complexity of feature encoding using G-codes is very low. It involves no more than a dot product operation. The benefit of using G-codes of low complexity is evident, since feature encoding is a time-consuming process in many classification systems (Chatfield et al., 2011). For the high-level image classification tasks, our image representation achieves the state-of-the-art performance on several benchmark datasets.

5. Experiments

Dataset We evaluate the proposed MMDL method for image classification on three widely used datasets, including scene image (15 Scene (Lazebnik et al., 2006), MIT 67 Indoor (Quattoni & A.Torralba, 2009) datasets), activity images (UIUC Sports dataset (Li & Fei-Fei, 2007)). Experimental setting for the three datasets are listed below:

- 15 Scene: It contains 4,485 images divided into 15 categories, each category has 200 to 400 images, and the average image size is 300×250. Following the same experimental setup as in (Lazebnik et al., 2006), we take 100 images per class for training and use the remaining images for testing.
- MIT 67 Indoor: This dataset contains images from 67 different categories of indoor scenes. There is a fixed training and test set containing approximately 80 and 20 images from each category respectively.
- UIUC Sports: This is a dataset of 8 event classes. Seventy randomly drawn images from each class are used for training and 60 for testing following (Li & Fei-Fei, 2007).

For the 15 Scene and UIUC Sports datasets, we randomly run experiments for 5 times, and record average and standard deviation of image classification accuracies over all image classes.

Experiment Setup For each image, image patches are densely sampled by every 16 pixels on image, under

three scales, 48×48, 72×72, and 96×96. For each image patch, we resize it to 48×48 and compute five kinds of features for describing it. The features are HoG, LBP, GIST (Oliva & Torralba, 2001), encoded SIFT and LAB color histogram. For the HoG and LBP features, we use the implementation in VLFeat (Vedaldi & Fulkerson, 2008); their dimensions are 279 and 522, respectively. For the GIST feature, we use the implementation provided by the authors of (Oliva & Torralba, 2001); its dimension is 256. When computing the encoded SIFT feature, we densely compute SIFT feature at the size of 16 by every 6 pixels; then the SIFTs are quantized by a 100 bins via k-means by assigning each SIFT feature to its nearest entry in the cluster; a histogram is built on the quantized SIFTs; dimension of the encoded SIFT feature is 100. For the LAB color histogram feature, we compute a 16 dimension histogram for each of the three channels. These five diverse features are normalized separately, concatenated into a 1205 dimensional vector, and normalized by its ℓ_2 norm as local patch representation. In MMDL, the weight parameter λ is set to 1; the number of iterations N in the optimization algorithm in Fig. 3 is set to 5; the sampling portion p^s is set to 0.7; and the normalization parameter σ is set to 0.5. In the step of "optimize $\mathbf{W}$", we use LibLinear (Fan et al., 2008) to solve this multi-class SVM problem. Training images of each dataset are used for learning our dictionary. The overall image representation is based on the description in Sec. 4. LibLinear is also used for image classification after image representation is computed.

5.1. Nature Scene Image Classification: A Running Example

In experiments on the 15 Scene dataset, we compare MMDL to k-means codebook learning, extremely randomized clustering forests (ERC-Forests) (Moosmann et al., 2008), the naive solution in Sec. 3.2, and some of the existing methods.

In Fig. 5, X-axis shows the number of codewords of k-means or G-codes; Y-axis shows average classification accuracy (in percentage) of different test. HoG, LBP, GIST and encoded SIFT are tested separately with MMDL (using 165 G-codes, 11 G-codes per-class); average classification accuracy of LBP (81.23%) is much higher than HoG (75.7%), encoded SIFT (74.74%) and GIST (74.27%). Fusing these four descriptors, we can obtain an improved accuracy of 86.35%. Color descriptor is not used in this dataset, because all images in this dataset are grey.

Using multiple features, we also test traditional k-means codebook learning, ERC-Forests codebook learning and our baseline method, mi-SVM + k-means, in Sec. 3.2. Codebooks learned by k-means, ERC-Forests, and mi-SVM + k-means are used for locality-constrained linear coding (LLC) in (Wang et al., 2010) which is a popular state-of-the-art feature encoding method. Then we follow the pipeline in (Wang et al., 2010) for image classification. In Fig. 5, we observe that ERC-Forests works even worse than k-means in this situation. Our baseline method (mi-SVM + k-means in Fig. 2) works better than raw k-means method, since it can explore discriminative image feature for each scene class. However, it is still worse than MMDL. Mi-SVM + k-means obtains an average classification accuracy of 85.06% using 1500 codewords, while the average classification accuracy of MMDL is 86.35% when only using 165 G-codes.

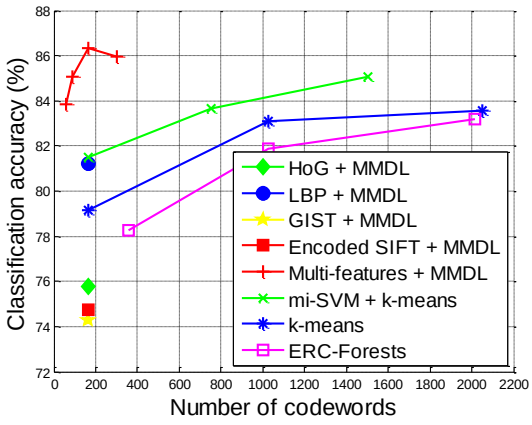

Figure 5. Average classification accuracies of different methods comparison on 15 Scene dataset over different number of codewords.

We compare MMDL with some previous methods in Table. 1. Notice that in (Lazebnik et al., 2006) and (Bo et al., 2010) non-linear SVM is used for image classification; (Li et al., 2010), (Yang et al., 2009) and our method adopt linear SVM. We observe that the performance of our method is very close to the best performance obtained by kernel descriptors, with very small number of codewords using linear SVM.

Learning G-codes using MMDL is computationally efficient. In this dataset, learning 11 G-codes for one category takes about 8 minutes on a 3.40GHz computer with an i7 multi-core processor. In the testing stage, it takes about 0.8 second for patch-level feature extraction, and takes less than 0.015 second for computing the image representation. The conclusions drawn from experiments on this dataset are general:

Table 1. Classification accuracy and number of codewords used of different methods on 15 Scene dataset.

Methods	Accuracy (%)	Number of codewords
Object Bank (Li et al., 2010)	80.90	2400
Lazebnik *et al.*(Lazebnik et al., 2006)	81.10±0.30	200
Yang *et al.*(Yang et al., 2009)	80.40±0.45	1024
Kernel Descriptors (Bo et al., 2010)	**86.70±0.40**	1000
Ours	86.35±0.45	**165**

Table 2. Classification accuracy on MIT 67 Indoor dataset.

Methods	Accuracy(%)
ROI+Gist (Quattoni & A.Torralba, 2009)	26.5
MM-scene (Zhu et al., 2010)	28.0
Centrist (Wu & Rehg, 2011)	36.9
Object Bank (Li et al., 2010)	37.6
DPM (Pandey & Lazebnik, 2011)	30.4
RBoW (Parizi et al., 2012)	37.93
Disc. Patches (Singh et al., 2012)	38.1
SPMSM (Kwitt et al., 2012)	44.0
LPR (Sadeghi & Tappen, 2012)	44.84
Ours	**50.15**

1. MMDL can naturally learn a metric to take the advantage of multiple features.
2. The max-margin formulation leads to very compact code for image representation and very competitive performance. It is clearly better than the naive solution.

5.2. Indoor Scene Image Classification

In the experiment on MIT 67 Indoor dataset, for each of the 67 classes, we learn 11 G-codes, 10 for positive cluster and 1 for negative cluster. Therefore, we have 737 G-codes in total. Fig. 6 shows some cluster models learned in *buffet* and *computer-room* category. Take the *computer-room* category as an example: cluster 2 corresponds to computers; and cluster 8 corresponds to desks. Clusters are learned given no more than image class labels. But it seems that they are very semantic meaningful.

Table. 2 summarizes the performances of our method and some previously published methods. Our performance is much better than traditional scene recognition methods, such as (Quattoni & A.Torralba, 2009; Zhu et al., 2010; Wu & Rehg, 2011). Here we focus on comparisons with three mid-level image representations, DPM (Pandey & Lazebnik, 2011), RBoW (Parizi et al., 2012), and Discriminative Patches (Singh et al., 2012). DPM, RBoW and our methods have used labels of training images for learning. Discriminative Patches method learns mid-level representation in an unsupervised way. In (Singh et al., 2012), they combine Discriminative Patches with DPM, Gist-color, and SP and obtained a classification accuracy of 49.4%. Our much better performance indicates the efficiency and effectiveness of MMDL.

Table 3. Classification accuracy on UIUC Sports dataset.

Methods	Accuracy (%)
Li *et al.*(Li & Fei-Fei, 2007)	73.4
Wu *et al.*(Wu & Rehg, 2009)	84.3
Object Bank (Li et al., 2010)	76.3
SPMSM (Kwitt et al., 2012)	83.0
LPR (Sadeghi & Tappen, 2012)	86.25
Ours	**88.47±2.32**

5.3. UIUC Sports Image Classification

In this experiment, we report the performance result of event recognition in the UIUC Sports dataset. For each event category, we only learn 11 different G-codes. This results in 88 codewords in total for image representation. However, our performance is consistently better than object bank (requires detailed human annotations) and two very recent approaches, LPR (Sadeghi & Tappen, 2012) and SPMSM (Kwitt et al., 2012) as shown in Table. 3. In addition, a codebook learning method (Wu & Rehg, 2009) using histogram intersection kernel has also been compared.

6. Conclusion

In this paper, we have proposed a dictionary learning strategy along the line of multiple instance learning. We demonstrate the effectiveness of our method, which is able to learn compact codewords and rich semantic information.

Acknowledgment: The work was supported by NSF CAREER award IIS-0844566, NSF award IIS-1216528, and NIH R01 MH094343. Part of this work was done while the first author was an intern at Microsoft Research Asia. It is also in part supported by the National Natural Science Foundation of China (NSFC) Grants 60903096, 61173120 and 61222308. Xinggang Wang was supported by Microsoft Research

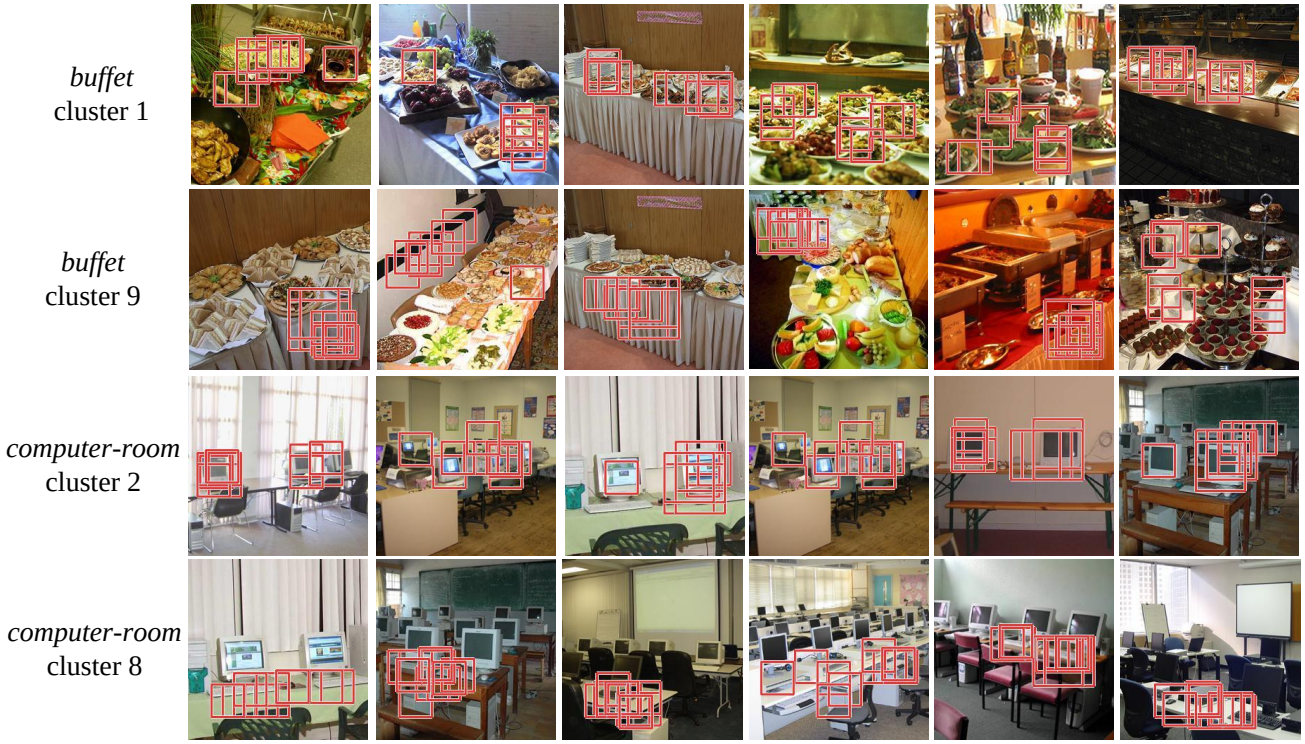

Figure 6. Some meaningful clusters learned by MMDL for different categories. Each row illustrates a cluster model: red rectangles shows positions of G-code classifier fired where SVM function value is bigger than zero.

Asia Fellowship 2012. We thank Jun Zhu, Liwei Wang, and Quannan Li for helpful discussions.

References

Andrews, S., Tsochantaridis, I., and Hofmann, T. Support vector machines for multiple-instance learning. *NIPS*, 15:561–568, 2002.

Blei, David M., Ng, Andrew Y., and Jordan, Michael I. Latent dirichlet allocation. *J. of Machine Learning Res.*, 3:993–1022, 2003.

Bo, L., Ren, X., and Fox, D. Kernel descriptors for visual recognition. *NIPS*, 7, 2010.

Bourdev, Lubomir and Malik, Jitendra. Poselets: Body part detectors trained using 3d human pose annotations. In *ICCV*, 2009.

Chatfield, K., Lempitsky, V., Vedaldi, A., and Zisserman, A. The devil is in the details: an evaluation of recent feature encoding methods. In *BMVC*, 2011.

Crammer, K. and Singer, Y. On the algorithmic implementation of multiclass kernel-based vector machines. *The Journal of Machine Learning Research*, 2:265–292, 2002.

Dietterich, T.G., Lathrop, R.H., and Lozano-Perez, T. Solving the multiple-instance problem with axis parallel rectangles. *Artificial Intelligence*, 89:31–71, 1997.

Dollár, P., Babenko, B., Belongie, S., Perona, P., and Tu, Z. Multiple component learning for object detection. *ECCV*, pp. 211–224, 2008.

Duda, R., Hart, P., and Stork, D. *Pattern Classification and Scene Analysis*. John Wiley and Sons, 2000.

Fan, Rong-En, Chang, Kai-Wei, Hsieh, Cho-Jui, Wang, Xiang-Rui, and Lin, Chih-Jen. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008.

Farhadi, A., Endres, I., Hoiem, D., and Forsyth, D. Describing objects by their attributes. In *CVPR*, pp. 1778–1785, 2009.

Fei-Fei, L. and Perona, P. A bayesian hierarchical model for learning natural scene categories. In *CVPR*, 2005.

Felzenszwalb, P.F., Girshick, R.B., McAllester, D., and Ramanan, D. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010.

Hinton, G. E., Osindero, S., and Teh, Y. W. A fast learning algorithm for deep belief nets. *Neural Computation*, 18 (7):1527–1554, 2006.

Jiang, Z., Zhang, G., and Davis, L.S. Submodular dictionary learning for sparse coding. In *CVPR*, pp. 3418–3425. IEEE, 2012.

Jurie, F. and Triggs, B. Creating efficient codebooks for visual recognition. In *ICCV*, pp. 604–610, 2005.

Kwitt, R., Vasconcelos, N., and Rasiwasia, N. Scene recognition on the semantic manifold. In *ECCV*, 2012.

Lazebnik, S. and Raginsky, M. Supervised learning of quantizer codebooks by information loss minimization. *IEEE Tran. PAMI*, 31:1294–1309, 2009.

Lazebnik, S., Schmid, C., and Ponce, J. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, volume 2, pp. 2169–2178, 2006.

LeCun, Y., Huang, F., and Bottou, L. Learning methods for generic object recognition with invariance to pose and lighting. In *Proc. of CVPR*, June 2004.

Li, L.J. and Fei-Fei, L. What, where and who? classifying events by scene and object recognition. In *ICCV*, 2007.

Li, L.J., Su, H., Xing, E.P., and Fei-Fei, L. Object bank: A high-level image representation for scene classification and semantic feature sparsification. *NIPS*, 24, 2010.

Lowe, D. G. Distinctive image features from scale-invariant keypoints. *Int'l J. of Comp. Vis.*, 60(2):91–110, 2004.

Mairal, J., Bach, F., and Ponce, J. Task-driven dictionary learning. *arXiv preprint arXiv:1009.5358*, 2010.

Moosmann, F., Triggs, B., and Jurie, F. Fast discriminative visual codebooks using randomized clustering forests. In *NIPS*, pp. 985–992, 2006.

Moosmann, Frank, Nowak, Eric, and Jurie, Frédéric. Randomized clustering forests for image classification. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(9):1632–1646, 2008.

Ojala, T., Pietikäinen, M., and Harwood, D. A comparative study of texture measures with classification based on feature distributions. *Pattern Recognition*, 29:51–59, 1996.

Oliva, Aude and Torralba, Antonio. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3): 145–175, 2001.

Pandey, M. and Lazebnik, S. Scene recognition and weakly supervised object localization with deformable part-based models. In *ICCV*, pp. 1307–1314. IEEE, 2011.

Parikh, D. and Grauman, K. Relative attributes. In *ICCV*, pp. 503–510, 2011.

Parizi, S.N., Oberlin, J.G., and Felzenszwalb, P.F. Reconfigurable models for scene recognition. In *CVPR*, pp. 2775–2782. IEEE, 2012.

Pechyony, D. and Vapnik, V. On the theory of learning with privileged information. In *NIPS*, 2010.

Quattoni, A. and A.Torralba. Recognizing indoor scenes. In *CVPR*, 2009.

Sadeghi, F. and Tappen, M.F. Latent pyramidal regions for recognizing scenes. In *ECCV*, 2012.

Serre, Thomas and Poggio, Tomaso. A neuromorphic approach to computer vision. *Commun. ACM*, 53(10):54–61, 2010.

Singh, Saurabh, Gupta, Abhinav, and Efros, Alexei A. Unsupervised discovery of mid-level discriminative patches. In *ECCV*, 2012.

Torresani, L., Szummer, M., and Fitzgibbon, A. Efficient object category recognition using classemes. *ECCV*, pp. 776–789, 2010.

Vedaldi, A. and Fulkerson, B. VLFeat: An open and portable library of computer vision algorithms, 2008.

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

Winn, J., Criminisi, A., and Minka, T. Object categorization by learned universal visual dictionary. In *ICCV*, volume 2, pp. 1800–1807, 2005.

Wu, J. and Rehg, J.M. Beyond the euclidean distance: Creating effective visual codebooks using the histogram intersection kernel. In *ICCV*, pp. 630–637, 2009.

Wu, J. and Rehg, J.M. Centrist: A visual descriptor for scene categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1489–1501, 2011.

Xu, Y., Zhu, J.Y., Chang, E., and Tu, Z. Multiple clustered instance learning for histopathology cancer image classification, segmentation and clustering. In *CVPR*, pp. 964–971, 2012.

Yang, J., Yu, K., Gong, Y., and Huang, T. Linear spatial pyramid matching using sparse coding for image classification. In *CVPR*, pp. 1794–1801, 2009.

Yang, J., Yu, K., and Huang, T. Supervised translation-invariant sparse coding. In *CVPR*, pp. 3517–3524. IEEE, 2010.

Yang, L., Jin, R., Sukthankar, R., and Jurie, F. Unifying discriminative visual codebook generation with classifier training for object category recognition. In *Proc. of CVPR*, 2008.

Zhang, Dan, Wang, Fei, Si, Luo, and Li, Tao. M3ic: maximum margin multiple instance clustering. In *IJCAI*, pp. 1339–1344, 2009.

Zhang, Min-Ling and Zhou, Zhi-Hua. M3miml: A maximum margin method for multi-instance multi-label learning. In *ICDM*, pp. 688–697, 2008.

Zhu, J., Li, L.J., Fei-Fei, L., and Xing, E.P. Large margin learning of upstream scene understanding models. *NIPS*, 24, 2010.

Zhu, J., Zou, W., Yang, X., Zhang, R., Zhou, Q., and Zhang, W. Image Classification by Hierarchical Spatial Pooling with Partial Least Squares Analysis. In *British Machine Vision Conference*, 2012.