Discriminative Cluster Refinement: Improving Object Category Recognition Given Limited Training Data

Liu Yang[1] yangliu1@cse.msu.edu
Rong Jin[1] rongjin@cse.msu.edu
Caroline Pantofaru[2] crp@cs.cmu.edu
Rahul Sukthankar[3,2] rahuls@cs.cmu.edu

[1]Department of Computer Science and Engineering, Michigan State University
[2]The Robotics Institute, Carnegie Mellon University
[3]Intel Research Pittsburgh

Abstract

A popular approach to problems in image classification is to represent the image as a bag of visual words and then employ a classifier to categorize the image. Unfortunately, a significant shortcoming of this approach is that the clustering and classification are disconnected. Since the clustering into visual words is unsupervised, the representation does not necessarily capture the aspects of the data that are most useful for classification. More seriously, the semantic relationship between clusters is lost, causing the overall classification performance to suffer.

*We introduce "**discriminative cluster refinement**" (DCR), a method that explicitly models the pairwise relationships between different visual words by exploiting their co-occurrence information. The assigned class labels are used to identify the co-occurrence patterns that are most informative for object classification. DCR employs a maximum-margin approach to generate an optimal kernel matrix for classification. One important benefit of DCR is that it integrates smoothly into existing bag-of-words information retrieval systems by employing the set of visual words generated by any clustering method. While DCR could improve a broad class of information retrieval systems, this paper focuses on object category recognition. We present a direct comparison with a state-of-the art method on the PASCAL 2006 database and show that cluster refinement results in a significant improvement in classification accuracy given a small number of training examples.*

1. Introduction

The success of vector space models for text information retrieval [2, 24] has motivated considerable interest in analogous techniques for computer vision applications, such as content-based image retrieval and object category recognition. The basic idea behind all of these approaches is to represent a document by the histogram of its words (typically termed a *bag-of-words*), which is treated as a high-dimensional vector. Given such a representation, classification and retrieval can be accomplished using a large variety of techniques including k-nearest-neighbor, naive Bayes and support vector machines [7]. For text applications, the mapping between a document and its vector space representation is straightforward since the concept of a "word" is universal. However, the notion of a "word" for computer vision documents is less obvious since images are not intrinsically composed of a concatenation of discrete word-like elements. Consequently, there has been significant interest in identifying low-level features in images that could serve an analogous role.

An attractive approach to defining word-like objects for images is to employ unsupervised clustering over the low-level features extracted from a large corpus of natural images. Popular candidates for these low-level features include small patches [1, 23] and local descriptors [17] that are obtained either at specific interest points or densely sampled over the image. Clustering is typically performed using an algorithm such as k-means [12], which identifies a good set of k cluster centers to represent the features observed in the corpus. Subsequently, each of the features in a new image is mapped to a cluster (corresponding to its nearest cluster center in feature space), enabling any image to be represented as a histogram over the clusters. Such ideas have shown promise in several computer vision applications [6, 10, 22, 25, 27].

Unfortunately, since the clustering into visual words is unsupervised, the representation does not necessarily capture those aspects of the data that are most important for classification. In particular, the semantic relationships between related clusters is ignored, which is detrimental to

1-4244-1180-7/07/$25.00 ©2007 IEEE

overall classification performance, as shown in Section 4. This problem becomes much more serious when the number of training images is small and is insufficient for reliably estimating the association between classes and a large number of clusters. To see the problem more clearly, consider two features f_a and f_b corresponding to the same concept. Due to the winner-takes-all nature of vector quantization, each feature is assigned to its closest cluster; thus these two synonym features are mapped to two distinct dimensions, say clusters C_a and C_b, in the vector space representation. Now, when the number of training images is small, it is likely that only one of these clusters appears in the training set. [1] Consider the scenario where C_a appears in the training images while C_b does not. In this case, the information related to cluster C_b will be wasted, and not used as a part of the classification scheme. However, if one could learn that the clusters C_a and C_b were strongly correlated, then the problem would be avoided. This paper introduces a method, termed "**Discriminative Cluster Refinement**" (DCR) to automatically learn the important relationships between clusters despite limited availability of training data.

Before describing discriminative cluster refinement in greater detail, we briefly discuss two intuitive approaches to the problem, which are unfortunately inadequate. The first idea is to employ a soft assignment of features to clusters: rather than clustering using k-means, one could use an expectation-maximization (EM) to associate each feature with a distribution over cluster centers. The hope would be that two related features would generate similar weights over nearby cluster centers even if k-means would assign them to different clusters. However, since the soft assignment is unsupervised, it still ignores class labels during clustering and fails to capture the relationship between clusters that are related to the same object category (as specified by class label).

The second idea is to incorporate the label information into the clustering procedure. The simplest approach is to augment the feature space with additional dimensions representing the class label and attempt to generate clusters that respect both the similarity according to the feature descriptor and also the class label. A more complicated variant is to learn a distance metric from the labeled images (e.g., [28]), and to cluster the feature descriptors using the learned distance metric. The problem with this approach is that due to the winner-takes-all nature, the resulting clusters may still separate two closely-related feature descriptors into two different clusters.

The failure of the above approaches motivates the central problem addressed by this paper: how should one exploit the class-label information from the training data in order to discriminatively refine the clusters and achieve better accuracy for object categorization? To address this fundamental question, we propose a framework that automatically augments cluster membership with the pairwise correlation between clusters. The key idea is to exploit the co-occurrence data of clusters. The underlying assumption is that two different clusters are likely to be related to the same concept if they co-occur frequently in the same images. However, directly using the co-occurrence information for cluster correlation estimation may be undesirable since unrelated clusters can also co-occur frequently in the same images. To resolve this problem, we exploit the label information to identify the informative co-occurrence patterns. More specifically, the co-occurrence patterns between two clusters are deemed to be informative when a support vector machine (SVM) classifier is able to maintain a large classification margin by collapsing these two clusters. To this end, DCR extends the theory of the SVM [4] to incorporate the cluster co-occurrence patterns into the maximum-margin classification model. An efficient algorithm based on the Second Order Cone Programming (SOCP) [3] technique is presented to improve the computational efficiency.

The primary contribution of our work is that it can improve a variety of existing bag-of-words approaches that are popular in object recognition, and that the general idea of unifying clustering with classification could significantly improve a broad class of algorithms in computer vision. It is important to note that our work is orthogonal to the work in data clustering. Indeed, it can be used to improve the classification results based on any clustering results. It is also important to emphasize that our work is particularly useful when the size of the training set is limited. This is because:

- The cluster co-occurrence information can be collected from both the labeled and unlabeled images. This information becomes particularly useful when the number of training examples is small.
- When the number of training examples is small, we will expect that many clusters will only appear in a few training images. As a result, the association between these clusters and class labels may not be learned reliably from the training examples. The proposed algorithm is able to improve the estimation for the cluster-class association by exploiting the estimated cluster correlation.

2. Related Work

There have been a small number of attempts to improve descriptor vocabularies and to integrate their construction into the classification model learning process [9, 15, 21, 22, 27]. Winn *et al*. [27] compress an initial large vocabulary by pair-wise word merging. Larlus and Jurie [15] take a different approach to the problem by enhancing the latent aspects concept. In previous approaches to latent aspect modeling,

[1] Note that the clustering procedure should not limit itself to the key points of training examples; instead, the visual vocabulary is constructed based on key points of all images in the entire dataset.

the problem was broken down such that images were a mixture of topics and topics were a mixture of words. Larlus and Jurie extend this so that words are a Gaussian mixture of descriptors, thereby learning the descriptor clusters as part of their model. Both of these approaches show that smaller dictionaries and allowing word dependency can lead to better classification. Farquhar *et al*. [9] propose to construct class-specific visual vocabularies using the Maximum A Posterior (MAP) approach. Moosmann *et al*. [21] propose to build discriminative visual word vocabularies using randomized clustering forests. Perronnin *et al*. [22] characterize images using a set of category-specific histograms, where each histogram describes whether the content can best be modeled by the universal vocabulary or by its corresponding category vocabulary. Our work differs from these approaches in the sense that our algorithm post-processes the clustering, and enhances the performance of classification by incorporating clustering information.

There has also been some work on defining a kernel for the similarity between two sets of image descriptors. Grauman and Darrell introduce the pyramid match kernel [11] for this task, however they do not use the training class labels to improve the discriminative power of the kernel. Lyu [18] introduces a new type of Mercer kernel, but once again it does not take advantage of training labels.

3. Discriminative Cluster Refinement

In this section, we introduce Discriminative Cluster Refinement (DCR) and present an efficient algorithm for its computation.

Let $\mathcal{D} = \{\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_{n_a}\}$ denote the collection of labeled and unlabeled images. Assume that the first n images are labeled by $\mathbf{y} = (y_1, y_2, \ldots, y_n)$, where y_i is $+1$ when image $\mathcal{I}_i$ contains a given object category and -1 when it does not. To represent the content of each image, we first extract keypoints and describe them using the SIFT descriptor [17]. The descriptors from all of the images (including both labeled and unlabeled data) are grouped into m clusters. Each image can now be represented by a histogram of the clusters corresponding to its descriptors. Let $\mathbf{b}_i \in \mathbb{N}^m$ be the histogram for image $\mathcal{I}_i$, and $B = (\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n)$ denote the histograms of all of the training images. We further denote the cluster histogram of all of the images (including both labeled and unlabeled ones) by $F = (\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_m)$, where $\mathbf{f}_i = (f_{i,1}, f_{i,2}, \ldots, f_{i,n_a})$ represents the occurrence of the ith cluster in all n_a images. This F matrix will be used in this paper to explicitly capture the cluster co-occurrence information. As discussed in the introduction, one major drawback with using the cluster histograms directly for object categorization is that the clustering and classification are disconnected. The goal of DCR is to estimate the cluster correlation that exploits the cluster co-occurrence information.

3.1. Discriminative Cluster Refinement Framework

Since our framework is an extension of SVM theory, we briefly review the dual formalism for SVM. An SVM solves the optimization:

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \quad & \boldsymbol{\alpha}^\top \mathbf{e} - \frac{1}{2}(\boldsymbol{\alpha} \circ \mathbf{y})^\top K (\boldsymbol{\alpha} \circ \mathbf{y}) \\ \text{s. t.} \quad & \boldsymbol{\alpha}^\top \mathbf{y} = 0 \\ & 0 \le \alpha_i \le C,\ i = 1, 2, \ldots, n, \end{aligned} \tag{1}$$

where $\boldsymbol{\alpha} = (\alpha_i, \alpha_2, \ldots, \alpha_n)$ are the weights assigned to the training images, $\mathbf{e}$ is a vector with all elements being 1, and the symbol $\circ$ denotes an element-wise product between two vectors. $K \in \mathbb{R}^{n \times n}$ is the kernel matrix whose elements $K_{i,j}$ represent the similarity between image $\mathcal{I}_i$ and $\mathcal{I}_j$. Furthermore, we denote the value of the objective function in Equation 1 by $\omega(K)$. It is well known that $\omega(K)$ is inversely-related to the classification margin [4]. So, to improve classification performance, we need to maximize the classification margin, which is equivalent to minimizing $\omega(K)$.

With the cluster histogram representation for images, we can compute the kernel matrix K as $K = B^\top B$. One major problem with such a similarity measurement is that two images will have zero similarity if they don't share any common clusters. This is problematic if clustering is not perfect and mistakenly divides a group of closely-related key points into two separate clusters. To address this problem, we introduce the cluster correlation matrix $M \in \mathbb{R}^{m \times m}$ where each element $M_{i,j}$ represents the correlation between the ith and the jth clusters. Then, the goal of discriminative cluster refinement is to estimate this cluster correlation matrix. To this end, we define the kernel matrix as $K = B^\top M B$, and search for the optimal cluster correlation matrix B by maximizing the classification margin, which is equivalent to minimizing the quantity $\omega(K)$. We thus have the following optimization problem M:

$$\arg\max_{M \succeq 0} \quad \omega(B^\top M B). \tag{2}$$

Note that the above restricts the cluster correlation matrix M to be positive semi-definite. This is a necessary condition for the kernel matrix K to be positive semi-definite.

The main problem with the formalism in Equation 2 is that it completely ignores the co-occurrence information in F when computing the cluster correlation M. In particular, one could assign a large value to the correlation between any two clusters that were not observed to co-occur in a training image — resulting in a serious overfitting problem. Thus, it is important to regularize the choice of cluster correlation matrix M according to the cluster co-occurrence matrix F. To this end, we consider an internal representation of clusters $Z = (\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_m)$ where each vector

$\mathbf{z}_i$ is the internal representation of the ith cluster. Then, the cluster correlation matrix can be computed as $MZ^\top Z$. Now, if the internal representation Z carries an equivalent amount of information as the co-occurrence matrix F, we would expect that the matrix F can be recovered from Z by a linear transformation. In other words, if F and Z are roughly equivalent representations of clusters, then there exists a matrix H such as $F = HZ$. Note that this is similar to the idea of non-negative matrix factorization (NMF) [14], which has been successfully applied to data clustering. Thus, we reformulate the problem in Equation 2 as

$$\begin{aligned} \underset{M,H,Z}{\arg\max} \quad & \omega(B^\top MB) \\ \text{s. t.} \quad & M = Z^\top Z,\ F = HZ. \end{aligned} \tag{3}$$

The key challenge in solving the optimization in Equation 3 arises in two aspects. First, $\omega(K)$ is not an analytic function. Rather, it is a function that results from the optimization problem in Equation 1. Second, the regularization of M does not come directly from the feature matrix F. Instead, the regularization comes indirectly through the internal representation Z. To overcome the first challenge, we rewrite the maximization problem in Equation 1 into a minimization problem by computing its dual, which leads to the following problem for $\omega(K)$:

$$\begin{aligned} \min_{t,\eta,\delta,\rho} \quad & t + 2C\delta^\top \mathbf{e} \\ \text{s. t.} \quad & \begin{pmatrix} K & \rho \circ \mathbf{y} + \lambda \mathbf{e} \\ (\rho \circ \mathbf{y} + \lambda \mathbf{e})^\top & t \end{pmatrix} \succeq 0 \\ & \rho = \mathbf{e} + \eta - \delta \\ & \delta_i \geq 0,\ \eta_i \geq 0,\ i = 1, 2, \ldots, n. \end{aligned}$$

By merging the above optimization problem with the problem in Equation 3, we obtain the following problem:

$$\begin{aligned} \min_{t,\eta,\delta,\rho,M} \quad & t + 2C\delta^\top \mathbf{e} \\ \text{s. t.} \quad & \begin{pmatrix} B^\top MB & \rho \circ \mathbf{y} + \lambda \mathbf{e} \\ (\rho \circ \mathbf{y} + \lambda \mathbf{e})^\top & t \end{pmatrix} \succeq 0 \\ & \rho = \mathbf{e} + \eta - \delta \\ & \delta_i \geq 0,\ \eta_i \geq 0,\ i = 1, 2, \ldots, n \\ & F = HZ,\ M = ZZ^\top. \end{aligned} \tag{4}$$

To address the second challenge, we combine the constraint $F = HZ$ and $M = ZZ^\top$ into the following positive semi-definite constraint

$$\begin{pmatrix} M & F^\top \\ F & T \end{pmatrix} \succeq 0, \tag{5}$$

where $T = HH^\top$. The proof that the above condition is equivalent to the constraints $F = HZ$ and $M = ZZ^\top$ is given in the Appendix. Using the constraint in Equation 5, we pose the optimization problem in Equation 4 as follows:

$$\begin{aligned} \min_{t,\eta,\delta,\rho,M} \quad & t + 2C\delta^\top \mathbf{e} + C_m \text{tr}(M) + C_t \text{tr}(T) \\ \text{s. t.} \quad & \begin{pmatrix} B^\top MB & \rho \circ \mathbf{y} + \lambda \mathbf{e} \\ (\rho \circ \mathbf{y} + \lambda \mathbf{e})^\top & t \end{pmatrix} \succeq 0 \\ & \delta_i \geq 0,\ \eta_i \geq 0,\ i = 1, 2, \ldots, n \\ & \rho = \mathbf{e} + \eta - \delta,\ \begin{pmatrix} M & F^\top \\ F & T \end{pmatrix} \succeq 0. \end{aligned} \tag{6}$$

Note that in the above, we introduce two regularization terms, *i.e.*, $C_m \text{tr}(M)$ and $C_t \text{tr}(T)$, into the objective function. These are useful in improving the stability of the optimal solution. The parameters C_m and C_t are used to weight the importance of the two regularization terms, respectively. They are set to be small values (*i.e.*, 0.01) in our experiments. Since the problem in Equation 6 is a Semi-Definite Programming (SDP) problem [3], in general it can be solved effectively using packages such as SeDuMi [26]. We do not impose uniqueness constraints on Z and H, because the goal is to compute M and T, thus any valid (Z, H) is sufficient.

Given the estimated cluster correlation matrix M, we can compute the kernel matrix K as $K = B^\top MB$, and solve the SVM classification model with kernel matrix K using a standard package such as LIBSVM [5].

3.2. An Efficient Algorithm for DCR

Although discriminative cluster refinement, as expressed in Equation 6, can be solved using SDP packages, this is typically very computationally expensive and does not easily scale to a large number of training examples. This subsection presents a computationally-efficient and scalable algorithm for DCR.

Let $\{\mathbf{v}_i, i = 1, 2, \ldots, n\}$ denote the right eigenvectors of matrix F, sorted in descending order of their eigenvalues θ_i. We then assume that the cluster correlation M can be constructed from the top s right eigenvectors of F, *i.e.*,

$$M \quad = \quad \gamma I_m + \sum_{i=1}^{s} (\alpha_i - \gamma) \mathbf{v}_i \mathbf{v}_i^\top, \tag{7}$$

where I_m is the $m \times m$ identity matrix, and $\alpha_i \geq 0, i = 1, \ldots, s$ and $\gamma \geq 0$ are non-negative combination weights. The introduction of term γI_m ensures that the matrix M is non-singular; this property is important when computing the expression for matrix T. By using Equation 7 for M, we convert the positive semi-definite constraint $M \succeq 0$ into simple non-negative constraints, *i.e.*, $\gamma \geq 0$ and $\{\alpha_i \geq 0\}_{i=1}^s$. Furthermore, the number of variables in M, which was originally $O(n^2)$, is now reduced to $s + 1$. Finally, we highlight two special cases of Equation 7:

1. When $\{\alpha_i = 0\}_{i=1}^{s}$, we have $M = \gamma I_m$. Thus, M is proportional to the identity matrix I_m, which indicates that the clusters are treated independently.
2. When $\gamma = 0$ and $\{\alpha_i = \theta_i^2\}_{i=1}^{s}$, we have $M = \sum_{i=1}^{s} \theta_i^2 \mathbf{v}_i \mathbf{v}_i^\top \approx F^\top F$. Thus, the solution M is basically $F^\top F$, which computes the correlation between any two clusters based on their visual features.

Given M as specified by Equation 7, we can obtain an expression from T. We use the Schur complement to convert the constraint in Equation 5 into the following inequality: $T \succeq FM^{-1}F^\top$.

Since there is only one term in the objective function related to T, *i.e.*, $C_t \text{tr}(T)$, we can show that the optimal solution for T is

$$T = FM^{-1}F^\top.$$

To efficiently compute M^{-1}, we note that M in Equation 7 can also be written as

$$M = \sum_{i=1}^{s} \alpha_i \mathbf{v}_i \mathbf{v}_i + \sum_{i=s+1}^{n} \gamma \mathbf{v}_i \mathbf{v}_i^\top.$$

Thus, M^{-1} can be computed as

$$\begin{aligned} M^{-1} &= \sum_{i=1}^{s} \frac{1}{\alpha_i} \mathbf{v}_i \mathbf{v}_i + \sum_{i=s+1}^{n} \frac{1}{\gamma} \mathbf{v}_i \mathbf{v}_i^\top \\ &= \frac{1}{\gamma} I_m + \sum_{i=1}^{s} \left(\frac{1}{\alpha_i} - \frac{1}{\gamma} \right) \mathbf{v}_i \mathbf{v}_i^\top. \end{aligned}$$

From this, we can obtain the following expressions for T and $\text{tr}(T)$:

$$T = \frac{1}{\gamma} FF^\top + \sum_{i=1}^{s} \left(\frac{\theta_i^2}{\alpha_i} - \frac{\theta_i^2}{\gamma} \right) \mathbf{v}_i \mathbf{v}_i^\top \quad (8)$$

$$\text{tr}(T) = \frac{1}{\gamma} \left(\text{tr}(FF^\top) - \sum_{i=1}^{s} \theta_i^2 \right) + \sum_{i=1}^{s} \frac{\theta_i^2}{\alpha_i}. \quad (9)$$

The next step is to simplify the constraint

$$\begin{pmatrix} B^\top M B & \rho \circ \mathbf{y} + \lambda \mathbf{e} \\ (\rho \circ \mathbf{y} + \lambda \mathbf{e})^\top & t \end{pmatrix} \succeq 0.$$

Using the Schur complement, we can rewrite the above constraint into the following form:

$$t \geq (\rho \circ \mathbf{y} + \lambda \mathbf{e})^\top (B^\top M B)^{-1} (\rho \circ \mathbf{y} + \lambda \mathbf{e}). \quad (10)$$

We can compute the quantity $(B^\top M B)^{-1}$ as

$$(B^\top M B)^{-1} = B^\top (BB^\top)^\dagger M^{-1} (BB^\top)^\dagger B,$$

where † denotes the pseudo-inverse of a matrix. By defining

$$\mathbf{u} = (BB^\top)^\dagger B (\rho \circ \mathbf{y} + \lambda \mathbf{e}),$$

we have

$$\begin{aligned} t &\geq (\rho \circ \mathbf{y} + \lambda \mathbf{e})^\top (B^\top M_z B)^{-1} (\rho \circ \mathbf{y} + \lambda \mathbf{e}) \\ &= \frac{1}{\gamma} \mathbf{u}^\top \mathbf{u} + \sum_{i=1}^{s} \left(\frac{1}{\alpha_i} - \frac{1}{\gamma} \right) (\mathbf{u}^\top \mathbf{v}_i)^2. \end{aligned} \quad (11)$$

Finally, the term $\text{tr}(M)$ in the objective function of Equation 6 can be computed as:

$$\begin{aligned} \text{tr}(M) &= \text{tr} \left(\sum_{i=1}^{s} \alpha_i \mathbf{v}_i \mathbf{v}_i^\top + \sum_{i=s+1}^{n} \gamma \mathbf{v}_i \mathbf{v}_i^\top \right) \\ &= (m-s)\gamma + \sum_{i=1}^{s} \alpha_i. \end{aligned} \quad (12)$$

By combining Equations 9, 11, and 12, we obtain

$$\begin{aligned} \min_{\eta, \delta, \alpha, \gamma, \mathbf{u}} \quad & \frac{1}{\gamma} \mathbf{u}^\top \mathbf{u} + \sum_{i=1}^{s} \left(\frac{1}{\alpha_i} - \frac{1}{\gamma} \right) (\mathbf{u}^\top \mathbf{v}_i)^2 + 2C\delta^\top \mathbf{e} \\ & + C_m \left((m-s)\gamma + \sum_{i=1}^{s} \alpha_i \right) \\ & + \frac{C_t}{\gamma} \text{tr}(FF^\top) + C_t \sum_{i=1}^{s} \left(\frac{\theta_i^2}{\alpha_i} - \frac{\theta_i^2}{\gamma} \right) \\ \text{s. t.} \quad & \eta_i \geq 0, \delta_i \geq 0, \ i = 1, 2, \ldots, n \\ & \alpha_i \geq 0, i = 1, 2, \ldots, s, \ \gamma \geq 0 \\ & \mathbf{u} = (BB^\top)^{-1} B \left(\lambda \mathbf{e} + \mathbf{y} + \mathbf{y} \circ (\eta - \delta) \right). \end{aligned}$$

Furthermore, we can convert the above problem into a Second Order Cone Programming (SOCP) problem [3] as follows:

$$\begin{aligned} \min_{a, \mathbf{b}, \mathbf{d}, w, \mathbf{u}, \alpha, \delta, \eta} \quad & a + \sum_{i=1}^{s} b_i + 2C\delta^\top \mathbf{e} + C_t \left(w + \sum_{i=1}^{s} d_i \right) \\ & + C_m \left((m-s)\gamma + \sum_{i=1}^{s} \alpha_i \right) \\ \text{s. t.} \quad & \delta_i \geq 0, \eta_i \geq 0, \ i = 1, 2, \ldots, n \\ & \alpha_i \geq 0, i = 1, 2, \ldots, s, \ \gamma \geq 0 \\ & \mathbf{u} = (BB^\top)^{-1} B \left(\lambda \mathbf{e} + \mathbf{y} + \mathbf{y} \circ (\eta - \delta) \right) \\ & \mathbf{g} = \left(I_m - \sum_{i=1}^{s} \mathbf{v}_i \mathbf{v}_i^\top \right) \mathbf{u} \\ & f_i = \mathbf{u}^\top \mathbf{v}_i, \\ & (\gamma, a, \mathbf{g}) \in \text{Rcone}, (\alpha_i, b_i, f_i) \in \text{Rcone} \\ & \left(\gamma, w, \sqrt{\text{tr}(FF^\top) - \sum_{i=1}^{s} \theta_i^2} \right) \in \text{Rcone} \\ & (\alpha_i, d_i, \theta_i) \in \text{Rcone}, \ i = 1, 2, \ldots, s \end{aligned}$$

where "Rcone" refers to the rotation of quadratic cone [26]. It is well known that a SOCP problem can be solved significantly more efficiently than a SDP problem.

4. Evaluation

4.1. Evaluation Dataset and Metric

We evaluate our approach on the PASCAL VOC Challenge 2006 data set [8]. The challenging dataset contains 5304 images with 9507 annotated objects. Ten annotated object classes are provided: bicycle, bus, car, motorbike, cat, cow, dog, horse, sheep and person. The populations of training/validation and test sets are well balanced across the distributions of images and objects by class. As a multi-object classification task, for each of the ten object classes, the goal is to predict the presence/absence of at least one object of that class in a test image. The binary classification performance for each object class, is measured quantitatively by the area under the ROC curve (AUR).

4.2. Local Detectors and Features

Our experiments closely follow the methodology described in [19]. We employ two separate channels of image representation, formed by two sets of SIFT [17] descriptor features extracted at two complementary sets of interest points: the Harris-Laplace detector [20], which captures corner-like regions and the Laplacian detector [16], which extracts blob-like regions. Following the notation in [19], we denote these two channels as HS-SIFT and LS-SIFT, respectively. The k-means clustering algorithm is applied to the local patch descriptors to produce 1000 clusters for each channel. Using the clusters of local feature descriptors, each image is represented as a histogram of clusters. Each image is represented by a concatenation of the histograms from these two channels, denoted as (HS+LS)-SIFT.

4.3. The Baseline Method

We compare our algorithm against a state-of-the-art method for object categorization [29], whose performance within $1\% - 2\%$ of the best in the object classification competition of the PASCAL VOC Challenge 2006 [8]. The basic idea of this baseline method is to classify objects by an SVM using the χ^2 kernel [13] that is computed based on the bag-of-features representation. To ensure that our implementation of this baseline method is accurate, we first reproduce the performance of object classification in the PASCAL VOC Challenge 2006 as reported in [8]. For this initial study, the χ^2 kernel is first computed for the entire dataset (*i.e.*, 2618 training and 2686 testing images) based on the dual channels of local descriptors. Then, the LIBSVM software package [5] is used for object classification. Finally, the posterior probability output of LIBSVM is used to to compute the AUR. Table 1 compares our baseline implementation against the results reported for INRIA Marszalek in [8]. Clearly, our baseline implementation achieves the reported accuracy. The minor differences can be attributed to potentially-different settings for the hinge loss parameter C in the SVM. We find that the χ^2 kernel is somewhat sensitive to the choice of C; our experiments use $C = 5$.

4.4. Object Category Recognition with Limited Training Data

This experiment focuses on the challenging problem of object category recognition given a limited number of labeled images. We randomly select 100 images for training. Both the baseline model and the proposed DCR algorithm learn a classification model from this small training set. In the implementation of the DCR algorithm, the top 200 right eigenvectors of the F matrix are used for computing the M matrix. The AUR is computed based on the prediction for the 2686 PASCAL testing images. Each experiment is repeated eighty times, and the AUR averaged over these trials is reported. Table 2 summarizes the AUR results of both the baseline model and the DCR algorithm for the HS-SIFT features, the LS-SIFT features, and the combined features.

First, we examine the classification results using the LS-SIFT features. As one should expect, the classification accuracy using the limited training set of 100 images is significantly worse than the results obtained from 2618 labeled training images. However, we note that for a number of categories, such as "bus", "car", and "bicycle", one can achieve a respectable classification accuracy even with this limited training data. Second, we observe that the DCR algorithm consistently improves over the baseline classification accuracy. The most noticeable case is the "dog" object category, whose area under the ROC curve is improved from 0.624 to 0.722 as a result of DCR. These results demonstrate that the DCR algorithm is effective at improving the accuracy of object classification when the training data is limited.

A more careful examination of the classification results indicates that DCR not only improves the classification accuracy but noticeably reduces the *standard deviation* in the classification accuracy. The standard deviations of DCR are mostly less than 0.010, however, those of the baseline algorithm are mostly between 0.010 and 0.020. The most significant case is category "sheep", whose standard deviation in AUR is reduced from 0.017 to 0.004. We hypothesize that the large standard deviation in the classification accuracy of the baseline model is mainly due to the small number of training images. Given a small number of training images, many feature clusters should only appear in a few training images. As a result, the association between the feature clusters and the class labels can not be reliably established. In extreme cases, when the feature clusters do not appear in any of the training images, no association can be established between these clusters and the class labels. We estimate that, on average, approximately 4 out of the 1000 LS-SIFT feature clusters are not used by any of randomly-selected

Table 1. Validation of baseline implementation: comparison against the INRIA_Marszalek entry in the PASCAL VOC Challenge 2006

Channel	bicycle	bus	car	cat	cow	dog	horse	motorbike	person	sheep
Baseline [HS-SIFT]	0.914	0.974	0.957	0.890	0.924	0.797	0.870	0.936	0.784	0.925
Baseline [LS-SIFT]	0.924	0.976	0.955	0.884	0.880	0.795	0.879	0.947	0.812	0.914
Baseline [(HS+LS)-SIFT]	0.934	0.980	0.964	0.913	0.926	0.834	0.901	0.964	0.833	0.934
(HS+LS)-SIFT [8]	0.929	0.984	0.971	0.922	0.938	0.856	0.908	0.964	0.845	0.944

Table 2. The AUR results on the PASCAL challenge 2006 dataset with 100 training examples.

	LS-SIFT		HS-SIFT		(HS+LS)-SIFT	
Object Class	Baseline	DCR	Baseline	DCR	Baseline	DCR
bicycle	0.784 ± 0.012	0.800 ± 0.006	0.764 ± 0.012	0.781 ± 0.004	0.793 ± 0.014	0.824 ± 0.003
bus	0.808 ± 0.021	0.842 ± 0.008	0.864 ± 0.016	0.888 ± 0.006	0.874 ± 0.016	0.881 ± 0.003
car	0.859 ± 0.005	0.863 ± 0.005	0.887 ± 0.003	0.897 ± 0.003	0.883 ± 0.003	0.891 ± 0.002
cat	0.725 ± 0.007	0.782 ± 0.001	0.778 ± 0.004	0.781 ± 0.003	0.776 ± 0.007	0.799 ± 0.001
cow	0.710 ± 0.010	0.735 ± 0.005	0.747 ± 0.011	0.767 ± 0.004	0.777 ± 0.010	0.779 ± 0.004
sheep	0.725 ± 0.017	0.796 ± 0.004	0.740 ± 0.012	0.765 ± 0.005	0.812 ± 0.008	0.842 ± 0.003
dog	0.624 ± 0.009	0.722 ± 0.002	0.654 ± 0.010	0.690 ± 0.003	0.674 ± 0.009	0.740 ± 0.002
horse	0.542 ± 0.014	0.601 ± 0.006	0.579 ± 0.016	0.659 ± 0.004	0.612 ± 0.016	0.655 ± 0.006
motorbike	0.737 ± 0.017	0.756 ± 0.010	0.717 ± 0.015	0.754 ± 0.004	0.750 ± 0.019	0.792 ± 0.007
person	0.596 ± 0.009	0.620 ± 0.003	0.595 ± 0.007	0.615 ± 0.002	0.622 ± 0.009	0.654 ± 0.002

100 images. Evidently, test images related to these missing feature clusters are likely to be classified incorrectly. By contrast, the DCR algorithm can resolve the problem of missing feature clusters by estimating the cluster correlation. For a missing feature cluster, its association with the class label can be reliably estimated through the correlation with other clusters that appear frequently in the training examples.

We then examine the classification results using the HS-SIFT features. Unlike the LS-SIFT features, for most categories the DCR algorithm only provides a slight improvement over the baseline. An exceptional case is the "horse" category, whose AUR improves from 0.579 to 0.659 with DCR. For the standard deviation in classification accuracy, we observe a similar result for the HS-SIFT as for the LS-SIFT features, namely, for a number of categories, the DCR algorithm significantly reduces the classification standard deviation. Another interesting observation is that the baseline model is able to improve the classification accuracy significantly by switching from the LS-SIFT features to the HS-SIFT features. However, the difference in classification accuracy between the LS-SIFT and the HS-SIFT features is marginal when using the DCR algorithm. This result indicates that the DCR algorithm is less sensitive to the quality of the local features.[2] In particular, the DCR algorithm is able to compensate for the weakness of a particular local feature by exploiting the co-occurrence information of feature clusters.

[2]Here, feature quality refers not to the choice of descriptor (SIFT in both cases) but to the value of an individual feature for classification.

Finally, we examine the classification results using the dual channels of feature descriptors (*i.e.*, (HS+LS)-SIFT). Again, we observe that the proposed DCR algorithm consistently improves both the accuracy and the reliability of object classification for a number of categories. Based on these results, we conclude that the proposed DCR algorithm is effective both at improving the classification accuracy and at reducing its standard deviation.

5. Conclusion

In this paper, we address an important problem for the bag-of-feature image representation, namely that the local features are clustered independently from the task of object classification. In order to connect the feature clusters with the class labels, we propose the discriminative cluster refinement (DCR) algorithm, which refines the cluster memberships by automatically estimating the correlation among clusters. To estimate cluster correlation, the DCR algorithm effectively exploits the cluster co-occurrence data, which can be collected from both the labeled and the unlabeled images. Furthermore, the DCR algorithm extends the theory of support vector machines to effectively identify those co-occurrence patterns that are most informative to the classification margin and ignore those that are irrelevant to object classification. The most important feature of the DCR algorithm is that it is orthogonal to the choice of the clustering algorithm and thus can be used to improve the classification performance for any clustering method. Empirical studies show that the proposed algorithm significantly improves both the accuracy and the reliability of object classification

when the number of training images is small. Given the practical difficulties involved in collecting large numbers of labeled training images for object recognition, we believe that DCR will enable researchers to better exploit the information contained in the limited training data. While this paper presents DCR in the context of object category recognition, the algorithm could be applied to a broad set of classification problems. In future work, we plan to conduct experiment of DCR with more training samples, and to evaluate the benefits of DCR in other domains.

Acknowledgments

This work is supported by the NSF grant IIS-0643494 and the NIH grant 1R01GM079688-01. We thank Marcin Marszalek and Cordelia Schmid for making direct comparisons possible by providing us with their interest point detections and descriptors on the PASCAL dataset.

Appendix

In this appendix, we show that the condition in Equation 5 is equivalent to the constraint $F = HZ$ and $M = Z^\top Z$. First, it can easily be shown that, if $F = HZ$ and $M = Z^\top Z$ hold, then Equation 5 will hold. This is because

$$\begin{pmatrix} M & F^\top \\ F & T \end{pmatrix} = \begin{pmatrix} Z^\top \\ H \end{pmatrix} (Z\ H^\top) \succeq 0.$$

Second, we show that if Equation 5 holds, there exists H such that $F = HZ$ and $M = Z^\top Z$. This is because any symmetric positive semi-definite matrix can also be written as $AA^\top$. We can further write $A = (Z^\top; H)$. By comparing the product $AA^\top = (Z^\top; H)(Z, H^\top)$ to the matrix in Equation 5, we obtain the constraint $F = HZ$ and $M = Z^\top Z$ hold.

References

[1] S. Agarwal, A. Awan, and D. Roth. Learning to detect objects in images via a sparse, part-based representation. *Trans. PAMI*, 2004.

[2] R. A. Baeza-Yates and B. A. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press / Addison-Wesley, 1999.

[3] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[4] C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2), 1998.

[5] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[6] G. Csurka, C. Bray, C. Dance, and L. Fan. Visual categorization with bags of keypoints. In *ECCV Workshop on Statistical Learning in Computer Vision*, 2004.

[7] R. Duda, P. Hart, and D. Stork. *Pattern Classification*. Wiley, 2001.

[8] M. Everingham, A. Zisserman, C. Williams, and L. Van Gool. The 2006 PASCAL visual object classes challenge.

[9] J. Farquhar, S. Szedmak, H. Meng, and J. Shawe-Taylor. Improving "bag-of-keypoints" image categorisation. Technical report, University of Southampton, 2005.

[10] L. Fei-Fei and P. Perona. A Bayesian hierarchical model for learning natural scene categories. In *Proc. CVPR*, 2005.

[11] K. Grauman and T. Darrell. The pyramid match kernel: Discriminative classification with sets of image features. In *Proc. ICCV*, 2005.

[12] J. Hartigan and M. Wang. A k-means clustering algorithm. *Applied Statistics*, 28, 1979.

[13] E. Hayman, B. Caputo, M. Fritz, and J.-O. Eklundh. On the significance of real-world conditions for material classification. In *Proc. ECCV*, 2004.

[14] P. O. Hoyer. Non-negative matrix factorization with sparseness constraints. *J. Mach. Learn. Res.*, 5, 2004.

[15] D. Larlus and F. Jurie. Latent mixture vocabularies for object categorization. In *Proc. BMVC*, 2006.

[16] T. Lindeberg. Feature detection with automatic scale selection. In *IJCV*, 1998.

[17] D. Lowe. Distinctive image features form scale-invariant keypoints. In *IJCV*, 2004.

[18] S. Lyu. Mercer kernels for object recognition with local features. In *Proc. CVPR*, 2005.

[19] M. Marszalek and C. Schmid. Spatial weighting for bag-of-features. In *Proc. CVPR*, 2006.

[20] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. In *IJCV*, 2004.

[21] F. Moosmann, B. Triggs, and F. Jurie. Randomized clustering forests for building fast and discriminative visual vocabularies. In *Proc. NIPS*, 2007.

[22] F. Perronnin, C. Dance, G. Csurka, and M. Bressian. Adopted vocabularies for generic visual categorization. In *Proc. ECCV*, 2006.

[23] E. Sali and S. Ullman. Combining class-specific fragments for object classification. In *Proc. BMVC*, 1999.

[24] G. Salton and M. J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., 1986.

[25] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proc. ICCV*, 2003.

[26] J. F. Sturm. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods Software*, 11/12(1–4), 1999.

[27] J. Winn, A. Criminisi, and T. Minka. Object categorization by learned universal visual dictionary. In *Proc. ICCV*, 2005.

[28] E. Xing, A. Ng, M. Jordan, and S. Russell. Distance metric learning with application to clustering with side-information. In *Proc. NIPS*, 2003.

[29] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local features and kernels for classifcation of texture and object categories: An in-depth study. Technical Report RR-5737, INRIA Rhône-Alpes, 2005.